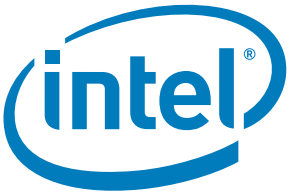


Intel® C620 Series Chipset Platform Controller Hub

Datasheet

October 2019

Document Number: 336067-009US

Notice: The information here is subject to change without notice.

Intel technologies features and benefits depend on system configuration and may require enabled hardware, software, or service activation. Learn more at intel.com, or from the OEM or retailer.

No computer system can be absolutely secure. Intel does not assume any liability for lost or stolen data or systems or any damages resulting from such losses.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

You may not use or facilitate the use of this document in connection with any infringement or other legal analysis concerning Intel products described herein. You agree to grant Intel a non-exclusive, royalty-free license to any patent claim thereafter drafted which includes subject matter disclosed herein.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest Intel product specifications and roadmaps.

Intel® AMT should be used by a knowledgeable IT administrator and requires enabled systems, software, activation, and connection to a corporate network. Intel AMT functionality on mobile systems may be limited in some situations. Your results will depend on your specific implementation. Learn more by visiting Intel® Active Management Technology.

Copies of documents which have an order number and are referenced in this document may be obtained by calling 1-800-548-4725 or by visiting www.intel.com/design/literature.htm.

Intel, Intel HD Audio, and the Intel logo are trademarks of Intel Corporation in the U.S. and other countries.

*Other names and brands may be claimed as the property of others.

Contents

1 Introduction 34
- 1.1 About this Manual 34
- 1.2 Overview 34
- 1.3 PCH I/O Features and Capabilities 35
- 1.4 PCH Software and Firmware 35
- 1.5 PCH SKUs 36
- 1.6 PCH Markings 36

2 PCH Controller Device IDs 38
- 2.1 Device and Revision ID Table 38

3 Pin Straps 42

4 Electrical Characteristics 45
- 4.1 PCH Power Supply Range 45
- 4.2 General DC Characteristics 46
- 4.3 AC Characteristics 59
- 4.4 PCIe* 4.0 Clock RMS Jitter Measurement 76
- 4.5 Overshoot/Undershoot Guidelines 76
- 4.6 Single Ended I/O Initial Power Up 77
- 4.7 Power Sequencing and Reset Signal Timings 80

5 Power and Ground Signals 87

6 Flexible I/O 89
- 6.1 Acronyms 89
- 6.2 Overview 89
- 6.3 Description 89

7 Memory Mapping 91
- 7.1 Overview 91
- 7.2 Functional Description 91
- 7.3 Memory Map 95

8 PCH and System Clocks 98
- 8.1 Acronyms 98
- 8.2 Overview 98
- 8.3 Signal Descriptions 98
- 8.4 Clock Signal Planes and States 101
- 8.5 Clocking Overview 101
- 8.6 Clock Interfaces 101
- 8.7 Integrated Clock Controller FW Accessible Registers Summary 103

9 Power Management 117
- 9.1 Acronyms 117
- 9.2 References 117
- 9.3 Overview 117
- 9.4 Signal Description 117
- 9.5 Integrated Pull-ups and Pull-downs 120
- 9.6 I/O Signal Planes and States 120
- 9.7 Functional Description 121
- 9.8 Power Management Configuration Registers Summary 144
- 9.9 PMC I/O Based Registers Summary 159
- 9.10 PMC Memory Mapped Registers Summary 178

10 Direct Media Interface 211
- 10.1 Acronyms 211

10.2 References 211
10.3 Overview 211
10.4 Signal Description 211
10.5 Integrated Pull-Ups and Pull-Downs 211
10.6 I/O Signal Planes and States 211
10.7 Functional Description 212

11 System Management 213
11.1 Acronyms 213
11.2 Overview 213
11.3 Feature 213
11.4 SMBus TCO I/O Registers Summary 218

12 High Precision Event Timer 227
12.1 References 227
12.2 Overview 227
12.3 HPET Memory Mapped Registers Summary 231

13 Thermal Management 240
13.1 Acronyms 240
13.2 PCH Thermal Sensor 240
13.3 Thermal Reporting Configuration Registers Summary 242
13.4 Thermal Reporting Memory Mapped Registers Summary 259

14 8254 Timers 273
14.1 Overview 273
14.2 8254 Timer Registers Summary 275

15 Integrated High Definition Audio 281
15.1 Acronyms 281
15.2 References 281
15.3 Overview 281
15.4 Signal Description 281
15.5 Integrated Pull-Ups and Pull-Downs 282
15.6 I/O Signal Planes and States 282
15.7 Features 282
15.8 High Definition Audio (D31:F3) PCI Configuration Registers Summary 283

16 Processor Interface 320
16.1 Acronyms 320
16.2 Overview 320
16.3 Signal Description 320
16.4 Integrated Pull-Ups and Pull-Downs 320
16.5 I/O Signal Planes and States 320
16.6 Functional Description 321
16.7 Processor Interface Memory Registers Summary 322

17 Enhanced Serial Peripheral Interface 326
17.1 Acronyms 326
17.2 References 326
17.3 Overview 326
17.4 Signal Description 327
17.5 Integrated Pull-Ups and Pull-Downs 327
17.6 I/O Signal Planes and States 327
17.7 Functional Description 328
17.8 Enhanced SPI (eSPI) Configuration Registers Summary 335
17.9 eSPI PCR Registers Summary 353

18 General Purpose Input and Output 362
18.1 Acronyms 362

18.2 References 362
18.3 Overview Integrated Pull-Ups and Pull-Downs 362
18.4 Overview 362
18.5 Signal Description 363
18.6 GPIO Community 0 Registers Summary 372
18.7 GPIO Community 1 Registers Summary 647
18.8 GPIO Community 2 Registers Summary 897
18.9 GPIO Community 3 Registers Summary 947
18.10 GPIO Community 4 Registers Summary 997
18.11 GPIO Community 5 Registers Summary 1131

19 Gigabit Ethernet Controller 1386
19.1 Acronyms 1386
19.2 References 1386
19.3 Overview 1386
19.4 Signal Description 1386
19.5 Integrated Pull-Ups and Pull-Downs 1387
19.6 I/O Signal Planes and States 1387
19.7 Functional Description 1388
19.8 GbE Configuration Registers Summary 1392
19.9 GbE Memory Mapped I/O Registers Summary 1405

20 Interrupt Interface 1412
20.1 Acronyms 1412
20.2 References 1412
20.3 Overview 1412
20.4 Signal Description 1412
20.5 Integrated Pull-Ups and Pull-Downs 1412
20.6 I/O Signal Planes and States 1413
20.7 Functional Description 1413
20.8 Advanced Programmable Interrupt Controller (APIC) (D31:F0) 1422
20.9 Serial Interrupt 1424
20.10 Interrupt Registers Summary 1427
20.11 Interrupt PCR Registers Summary 1438
20.12 APIC Indirect Registers Summary 1450
20.13 Advanced Programmable Interrupt Controller (APIC) Registers Summary 1466

21 Low Pin Count 1468
21.1 Acronyms 1468
21.2 References 1468
21.3 Overview 1468
21.4 Signal Description 1469
21.5 Integrated Pull-ups and Pull-downs 1469
21.6 I/O Signal Planes and States 1469
21.7 Functional Description 1469
21.8 LPC Configuration Registers Summary 1472
21.9 LPC PCR Registers Summary 1488

22 PCI Express* 1490
22.1 Acronyms and Register Attributes/Modifiers 1490
22.2 References 1490
22.3 Overview 1490
22.4 PCIe Root Port 1491
22.5 PCIe Upstream Port 1499
22.6 PCIe Virtual Switch Ports 1505
22.7 PCI Express Root Port Configuration Registers Register Summary 1506
22.8 PCI Express NP16 Uplink Port Configuration Registers Register Summary 1564
22.9 PCI Express NP8 Uplink Port Configuration Registers Register Summary 1606

22.10 PCI Express Virtual Switch Port (VSP)[5:0] Configuration Registers Summary 1646

23 Real Time Clock 1686
23.1 Acronyms 1686
23.2 References 1686
23.3 Overview 1686
23.4 Signal Description 1686
23.5 Integrated Pull-Ups and Pull-Downs 1687
23.6 I/O Signal Planes and States 1687
23.7 Functional Description 1687
23.8 RTC Indexed Registers Summary 1690
23.9 RTC Configuration Registers Summary 1695

24 Serial ATA 1697
24.1 Acronyms 1697
24.2 References 1697
24.3 Overview 1697
24.4 Signal Description 1698
24.5 Integrated Pull-Ups and Pull-Downs 1705
24.6 I/O Signal Planes and States 1705
24.7 Functional Description 1706
24.8 SATA Port Registers 1714
24.9 SSATA Port Registers 1785

25 System Management Interface and SMLink 1852
25.1 Acronyms 1852
25.2 References 1852
25.3 Overview 1852
25.4 Signal Description 1852
25.5 Integrated Pull-Ups and Pull-Downs 1853
25.6 I/O Signal Planes and States 1853
25.7 Functional Description 1853
25.8 SMBus TCO I/O Registers Summary 1854

26 Host System Management Bus Controller 1863
26.1 Acronyms 1863
26.2 References 1863
26.3 Overview 1863
26.4 Signal Description 1863
26.5 Integrated Pull-ups and Pull-downs 1863
26.6 I/O Signal Planes and States 1864
26.7 Functional Description 1864
26.8 SMBus Configuration Registers Summary 1877
26.9 SMBus I/O and Memory Mapped I/O Registers Summary 1888
26.10 SMBus PCR Registers Summary 1902

27 Serial Peripheral Interface 1904
27.1 Acronyms 1904
27.2 References 1904
27.3 Overview 1904
27.4 Signal Description 1904
27.5 Integrated Pull-Ups and Pull-Downs 1905
27.6 I/O Signal Planes and States 1905
27.7 Functional Description 1905
27.8 SPI Configuration Registers Summary 1911
27.9 SPI Memory Mapped Registers Summary 1919
27.10 BIOS Flash Program Registers Summary 1960

28 Testability 1963

28.1 Acronyms....1963
28.2 References....1963
28.3 Overview....1963
28.4 Signal Description....1963
28.5 I/O Signal Planes and States....1964
28.6 Intel® Trace Hub (Intel® TH)....1964
28.7 Direct Connect Interface (DCI)....1965
28.8 Intel Trace Huh Configuration Registers Summary....1967

29 Universal Serial Bus....1978
29.1 Acronyms....1978
29.2 References....1978
29.3 Overview....1978
29.4 Signal Description....1978
29.5 Integrated Pull-Ups and Pull-Downs....1981
29.6 I/O Signal Planes and States....1981
29.7 Functional Description....1981
29.8 xHCI Configuration Registers Summary....1981
29.9 xHCI Memory Mapped Registers Summary....2009

30 Intel® Management Engine Interface (Intel® MEI) (D22:F0, D22:F1, and D22:F4)....2058
30.1 Intel Management Engine Interface Configuration Registers Summary....2058
30.2 Intel MEI MMIO Registers Summary....2077

31 IDE Redirect (IDE-R) (D22:F2)....2078
31.1 IDE Redirect PCI Configuration (D22:F2) Registers Summary....2078

32 Keyboard and Text (KT) (D22:F3)....2093
32.1 Serial Port for Remote Keyboard and Text (KT) (D22:F3) Registers Summary....2093
32.2 Keyboard and Text (KT) Additional Configuration Registers Summary....2106

33 Innovation Engine (IE)....2108
33.1 Acronyms....2108
33.2 References....2108
33.3 Overview....2108
33.4 Signal Description....2108
33.5 Integrated Pull-Ups and Pull-Downs....2109
33.6 I/O Signal Planes and States....2110
33.7 Functional Description....2111
33.8 Registers....2114

34 Fan Control....2116
34.1 Acronyms....2116
34.2 References....2116
34.3 Overview....2116
34.4 Signal Description....2116
34.5 Integrated Pull-ups and Pull-downs....2117
34.6 I/O Signal Planes and States....2117
34.7 Functional Description....2117

35 Enterprise Value Add....2118
35.1 Acronyms....2118
35.2 References....2118
35.3 Overview....2118
35.4 Functional Description....2118
35.5 MROM0 Configuration Registers Summary....2119
35.6 MROM1 Configuration Registers Summary....2135

36 Additional Configuration Registers....2150

36.1 FIA Configuration PCR Registers Summary 2150
36.2 DMI Configuration Registers Summary 2158
36.3 DCI Configuration Registers Summary 2185
36.4 PSF1 PCR Registers Summary 2187
36.5 PSF2 PCR Registers Summary 2198
36.6 PSF3 PCR Registers Summary 2201
36.7 PSF4 PCR Registers Summary 2222
36.8 I/O Trap Registers Summary 2223
36.9 P2SB Bridge 2233
36.10 PHY Tuning Registers 2258

37 Debug Registers 2261
37.1 Introduction 2261
37.2 Register Summary 2261

38 10 GbE Controller 2283
38.1 10 GbE Controller Overview 2283
38.2 Typical System Configurations 2285
38.3 High-Speed Signal Interface 2287
38.4 Sideband Signals 2288
38.5 Low-speed Sideband Signal Interface 2288
38.6 Host Interface 2291
38.7 Various Interfaces 2295
38.8 Features 2296
38.9 Overview Of Standards 2306
38.10 PCI-Express (PCIe) 2311
38.11 Link Management Topologies 2350
38.12 General Purpose I/O (GPIO) 2409
38.13 Initialization 2414
38.14 Device/Port/Function Configuration 2451
38.15 Shared Resource Management 2456
38.16 Power Management 2460
38.17 Network Interfaces Power Management 2466
38.18 Protocol Offload (Proxying) 2470
38.19 Non-Volatile Memory Map 2481
38.20 NVM General Summary Table 2503
38.21 Shared Resources 2709
38.22 Interrupts 2908
38.23 Virtualization 2917
38.24 Data Center Bridging (DCB) 2927
38.25 Transmit Scheduling 2963
38.26 Host Memory Cache (HMC) 3016
38.27 AQ 3036
38.28 Statistics 3056
38.29 LLDP Protocol 3062
38.30 LAN Engine 3084
38.31 LAN Transmit Data Path 3112
38.32 TimeSync (IEEE 1588 and 802.1AS) 3132
38.33 Software Timer 3142
38.34 Protocol Engine (PE) 3142
38.35 Features 3143
38.36 Functional Description 3145
38.37 System Manageability 3280
38.38 Programming Interface 3424
38.39 Device Registers - PF 3432
38.40 Device Registers - VF 3683
38.41 PCIe Programming Interface 3700

38.42 PCI Configuration Space 3701
38.43 Capabilities in PCI Configuration Space 3708
38.44 PCIe Extended Configuration Space 3730
38.45 Virtual Functions 3748
38.46 Reliability, Diagnostics and Testability 3757
38.47 Acronyms 3759

39 Intel® QuickAssist Technology (Intel® QAT) 3761
39.1 Acronyms 3761
39.2 References 3762
39.3 Overview 3762
39.4 Intel QuickAssist Technology Features 3762
39.5 Intel QuickAssist Integrated Accelerator 3763
39.6 Intel QuickAssist Technology and PCH Upstream Port Requirements 3766
39.7 Intel QuickAssist Technology Endpoint Only (EPO) Mode 3767
39.8 Intel QuickAssist Technology PCIe Endpoint Function 3768
39.9 Interrupts 3771
39.10 Endpoint Errors 3772
39.11 Error Handling with SR-IOV 3777
39.12 PCIe Endpoint Function 0 Registers (Intel QuickAssist Technology) - EndPoint (EP) Physical Function (PF) PCI Configuration Space 3778
39.13 PCIe Endpoint Function 0 Registers (Intel QuickAssist Technology) - EndPoint (EP) Virtual Function (VF) PCI Configuration Space 3837

40 PCH EndPoint Only (EPO) Mode 3870
40.1 Acronyms 3870
40.2 References 3870
40.3 PCH EndPoint Only (EPO) Mode Overview 3870
40.4 PCH Endpoint Only (EPO) Mode Features 3870
40.5 PCH EPO Mode Considerations 3871
40.6 EPO PCH - PCIe Uplink 3874

Figures

4-1 PCI Express Transmitter Eye 60
4-2 PCI Express Receiver Eye 61
4-3 Clock Timing 63
4-4 SMBus/SMLink Transaction 64
4-5 PCH Test Load 64
4-6 USB Rise and Fall Times 65
4-7 USB Jitter 66
4-8 USB EOP Width 66
4-9 SMBus/SMLink Timeout 68
4-10 Intel® High Definition Audio (Intel® HD Audio) Input and Output Timings 68
4-11 Valid Delay from Rising Clock Edge 69
4-12 Setup and Hold Times 69
4-13 Float Delay 69
4-14 Output Enable Delay 70
4-15 Valid Delay from Rising Clock Edge 70
4-16 Setup and Hold Times 70
4-17 Pulse Width 71
4-18 SPI Timings 73
4-19 eSPI Clock Timings 73
4-20 eSPI Command timings 73
4-21 eSPI DATA Timings 74
4-22 PM_SYNC Timings 74
4-23 NCSI Timings 75

4-24 Maximum Acceptable Overshoot/Undershoot Waveform 77
4-25 Intel® C620 Series Chipset Power Supply Ramp Order 78
4-26 SEIO Pulses on Power Up 79
4-27 Power-up Glitches on GPIOs with Pull-up Resistors 80
4-28 Power Ramp with DSX Support 80
4-29 Power Ramp Without DSX Support 81
4-30 SLP_S5# High to PLTRST# High 83
4-31 S0 to S5 84
4-32 Powerdown to DeepSX 85
4-33 Surprise Powerdown to G3 85
4-34 eSPI Reset 86
6-1 HSIO Muxing on the Intel® C620 Series Chipset 89
9-1 Conceptual Diagram of SLP_GbE# 138
9-2 ADR Flow 143
11-1 TCO Compatible Mode SMBus Configuration 215
11-2 Advanced TCO Mode 217
16-1 PM_SYNC protocol 321
17-1 Basic eSPI Protocol 328
17-2 eSPI Slave Request to PCH for PCH Temperature 333
17-3 PCH Response to eSPI Slave with PCH Temperature 333
17-4 eSPI Slave Request to PCH for PCH RTC Time 334
17-5 PCH Response to eSPI Slave with RTC Time 334
22-1 Generation of SERR# to Platform 1498
22-2 PCIe Uplink Configuration 1 1503
22-3 NP8 PCIe Uplink Configuration 2 1504
22-4 Virtual Switch Ports Block Diagram 1506
24-1 Flow for Port Enable/Device Present Bits 1709
24-2 Serial Data Transmitted Over the SGPIO Interface 1713
27-1 Flash Descriptor Regions 1908
28-1 Platform Setup with Intel Trace Hub 1965
28-2 Platform Setup with DCI Connection 1967
33-1 IE Block Diagram 2111
38-1 Typical Rack / Pedestal System Configuration 2285
38-2 Typical Blade System Dual-Redundant Star Configuration 2286
38-3 10 GbE Controller Interfaces 2287
38-4 Traffic Class (TC) and Virtual Channels (VC) 2316
38-5 Error Reporting Mechanism 2320
38-6 10 GbE Controller MAC Configuration 2332
38-7 Basic PHY/Module Connectivity 2343
38-8 MDIO Timing Sourced By the MAC 2344
38-9 MDIO Timing Sourced by the PHY 2344
38-10 Behavior of MDIO During TA Field of a Read Transaction 2347
38-11 Basic Example of Module Connectivity (1 of 2) 2351
38-12 Basic Example Of Module Connectivity 2351
38-13 Quad 10 GbE SFP+ 2353
38-14 Quad 10 GbE SFP+ with Re-Driver PHY 2354
38-15 Quad 10 GbE QSFP+ 2354
38-16 Quad 10GBASE-T/1000BASE-T 2355
38-17 PHY Auto-Detect Flow 2357
38-18 Sample Shared External Flash Illustration 2379
38-19 NVM Shadow RAM 2381
38-20 Flash High Level Map Diagram 2383
38-21 Initial State Before the NVM Update Command 2389
38-22 Final State After Completion of the NVM Update Command 2389
38-23 Sign and Verify Procedures For Authenticated NVM Modules 2396
38-24 Hierarchical Reset Tree 2415

38-25 PCI Reset Flow 2418
38-26 PFR Flow....2420
38-27 VF Reset (VFR) Flow....2423
38-28 Global Resets Flow....2426
38-29 Virtualized Device Driver Initialization Flow....2449
38-30 Power Management State Diagram....2461
38-31 Structure of a Flexible Functionality Module 2488
38-32 Structure of a Type 1 Module....2489
38-33 Structure of a Type 2 Module....2490
38-34 Structure of a Type 3 Module....2491
38-35 Access Through a Type 3 Module of 32-Bits Wide Data Elements 2492
38-36 Access Through a Type 3 Module of 64-Bits Wide Per Port Data Elements 2493
38-37 Structure of a Type 4 Module....2494
38-38 Access Through a Type 4 Module of 32-Bits Wide Data Elements 2495
38-39 Access Through a Type 4 Module of 64-Bits Wide Per Port Data Elements 2496
38-40 Shadow RAM Word Write Command 2502
38-41 CSR Write Command....2502
38-1 Input Set Example for an FD Filter 2718
38-2 RSS Block Diagram 2728
38-3 Flow Director Filter Programming 2730
38-4 Flow Director Table Allocation (Example for 4 x PFs: PF0 ... PF3)....2732
38-5 Buckets and Collided Filter Entries Space 2736
38-6 Quad Hash Filter....2737
38-7 L2 Tags Order 2749
38-8 Transmit Tag Handling....2749
38-9 Tag Insertion....2752
38-10 Example: Non-VLAN Aware, Non-DCB Aware Operating System 2762
38-11 Tx Inner-to-Outer UP Translation 2763
38-12 Standard L2 Frame Format....2764
38-13 Order of L2 EtherType Headers....2765
38-14 MPLS Header....2766
38-15 IPv4 Datagram 2768
38-16 IPv6 Datagram 2769
38-17 IPv6 Hop-by-hop Extension Header 2770
38-18 IPv6 Routing Header 2770
38-19 IPv6 Fragment Header....2770
38-20 IPv6 Destination Options....2770
38-21 IPv6 Mobility Header 2771
38-22 ARP Packet Format....2774
38-23 UDP Packet Format 2775
38-24 IPv4 Pseudo Header for TCP/UDP Checksum....2775
38-25 IPv6 Pseudo Header for TCP/UDP Checksum....2775
38-26 TCP Packet Format....2776
38-27 SCTP Packet Format....2777
38-28 Tunneled Packet Formats....2777
38-29 IP / Teredo Tunneling....2778
38-30 VXLAN Header Structure....2779
38-31 VXLAN-GPE Header Structure 2782
38-32 NSH Header Structure 2783
38-33 MPLS Header Structure....2784
38-34 Internal Switching Configuration — Software Virtual Embedded Bridging 2786
38-35 Internal Switching Configuration — Hardware VEB....2787
38-36 External Switching Configuration — Virtual Port Aggregator 2788
38-37 External Switching Configuration — Port Virtualizer 2790
38-38 Cascaded VEB 2791
38-39 Cascaded....2792

38-40 Cascaded Virtual Ethernet Port Aggregator (VEPA) 2793
38-41 Cascaded VEB and Port Virtualizer 2794
38-42 Switching Elements Per Port 2798
38-43 Port Virtualizer Flow 2803
38-44 Cloud Forwarding Filters Priority 2807
38-45 Cloud VEB Flow Diagram 2809
38-46 VEPA Flow Diagram 2810
38-47 Advanced Switching Capabilities 2819
38-48 Private VLAN Configuration 2822
38-49 Example: Non-VLAN Aware, Non-DCB Aware Operating System 2825
38-50 Tx Inner-to-outer UP Translation 2826
38-51 Switching Structure Example 2847
38-52 Switch Configuration (Basic L2) 2848
38-53 VEB/VEPA 2849
38-54 Port Virtualizer 2850
38-55 Port Virtualizer + VEB/VEPA 2851
38-56 LAN Initial Configuration 2851
38-57 Conceptual Interrupt Logic 2911
38-58 Mapping Interrupt Causes to Interrupt Signaling 2915
38-59 VMDq System 2920
38-60 SR-IOV-Based System 2921
38-61 Operation of the Rx Low Latency Prioritization Arbiter 2931
38-62 Normal Partitioning of the Per-port Rx Packet Buffer 2934
38-63 Normal Configuration at Step 1 2940
38-64 Normal Configuration at Step 2 2941
38-65 Normal Configuration at Step 3 2942
38-66 ETS Recommendation FLV Format 2951
38-67 ETS Configuration FLV Format 2951
38-68 Priority-based Flow Control TLV Format 2953
38-69 Application Priority Table TLV Format 2954
38-70 Application Priority Table 2954
38-71 IEEE 802.3 Interface Delays 2957
38-72 Hierarchy of Switching Elements 2963
38-73 10 GbE Ethernet Controller Bandwidth Management Attributes 2964
38-74 Example of Bandwidth Distribution 2966
38-75 Example of Limiting Maximum Bandwidth 2968
38-76 Queue Qet Assignment Diagram 2970
38-77 ETS-based Scheduler Configuration Scheme 2971
38-78 ETS-based Bandwidth Configuration Example 2972
38-79 Default VSI Configuration 3003
38-80 Adding VEB/PA 3004
38-81 Add VSI 3005
38-82 Software Control Zone in SFP Mode 3006
38-83 Enable Physical Port ETS 3007
38-84 Change Physical Port ETS 3007
38-85 initial Setting in SFP (One Initial VSI/Port) 3009
38-86 ETS-Based Scheme: Relative Bandwidth Calculation Example 3013
38-87 Host Memory Cache Private Memory Address Space 3019
38-88 HMC FPM Space 3021
38-89 HMC FPM Object Access 3022
38-90 HMC Direct Segment 3023
38-91 FPM Address Decomposition 3031
38-92 PF LAN FPM 3084
38-93 LAN Queue Pairs Allocation Example of 4 PFs 3085
38-94 PF Queues (Example) 3086
38-95 Tunneled Packet Formats 3091

38-96 Receive Packet in System Memory 3093
38-97 Receive Descriptor Ring Structure 3103
38-98 Header Split Option by Packet Formats 3109
38-99 Transmit Descriptor Ring Structure....3120
38-100 Transmit L3 and L4 Header Lengths (in Host Memory) for Integrity Offload....3128
38-101 TSO Functionality (Example) 3129
38-102 Sync Flow and Offset Calculation....3135
38-103 PDelay Flow 3135
38-104 Time Stamp Point 3136
38-105 1588 Packet Format 3141
38-106 PE Packet Classification....3146
38-107 AEQ 3148
38-108 CEQ 3149
38-109 CQ....3150
38-110 CQ Resize Operation....3152
38-111 STag Decomposition to Physical Address 3153
38-112 Direct Page Translation....3154
38-113 Two-Level Page Translation 3155
38-114 STag Format 3155
38-115 Memory Region to Memory Window Relationship 3156
38-116 PE Page Location in BARs....3157
38-117 PE Pages 3158
38-118 PE SQ....3159
38-119 PE Operation: SQ....3160
38-120 PE Operation: Virtual SQ....3162
38-121 Combined Push Page Detail 3164
38-122 Split Push Page Detail....3165
38-123 PE Operation: SQ Push Mode 3166
38-124 PE RQ 3168
38-125 PE Operation: RQ....3169
38-126 Partial FPDU Software Algorithm 3174
38-127 10 GbE Controller and Host Memory Data Structures 3184
38-128 Sideband Interface....3281
38-129 Manageability Decision Filters 3294
38-130 OS-to-BMC Flow 3301
38-131 OS-to-BMC and VM-to-VM Filtering....3302
38-132 SMBus ARP Flow 3307
38-133 Single NC Package, Two NC Channels 3347
38-134 Two NC Packages (Left, With Two NC Channels and Right, With One NC Channel)....3347
38-135 NC-SI Frames Receive Flow for the NC....3349
38-136 Multi-NC Environment....3408
38-137 10 BGBE Controller MCTP Connections 3412
38-138 MCTP Endpoints Topology 3414
38-139 MCTP bus transition state machine 3415
38-140 VF stride....3743
39-1 Intel QuickAssist Technology Usage Block Diagram....3763
39-2 Direct I/O Assignment Model with IOV Support 3765
39-3 Intel QuickAssist Technology and PCH Upstream Ports 3767
39-4 Endpoint Functional Block Diagram....3768
39-5 PF EP PCI Configuration Registers 3778
40-1 PCH in EndPoint Only (EPO) Mode Block Diagram 3873
40-2 Intel QuickAssist Technology and PCH Upstream Ports 3874
40-3 EPO PCH with Uplink Configuration 1 3875
40-4 EPO PCH with Uplink Configuration 2 3876

Tables

1-1 PCH I/O Capabilities 35
1-2 Intel® C620 Series Chipset SKU Differentiation 36
1-3 PCH Markings 36
2-1 Intel® C620 Series ChipsetDevice and Revision ID Table 38
3-1 Functional Strap Definitions 42
3-2 SPI Flash and BIOS Location 44
4-1 Voltage Ranges 45
4-2 ICC Sx 46
4-3 Absolute Maximum Voltage Levels for Power Planes 46
4-4 Intel® C620 Series Chipset PCH Estimated I_{cc} (Maximum Configuration C627 and C629) 46
4-5 Intel® C620 Series Chipset PCH Estimated I_{cc} (C621) 47
4-6 Intel® C620 Series Chipset PCH Estimated I_{cc} (C622) 48
4-7 Intel® C620 Series Chipset PCH Estimated I_{cc} (C624) 49
4-8 Intel® C620 Series Chipset PCH Estimated I_{cc} (C625) 49
4-9 Intel® C620 Series Chipset PCH Estimated I_{cc} (C626) 50
4-10 Intel® C620 Series Chipset PCH Estimated I_{cc} (C628) 51
4-11 Intel® C620 Series Chipset Real Time Clock Vcc and Icc 52
4-12 DC Characteristics of Single Ended Signals 52
4-13 Differential Signals Characteristics 56
4-14 PCI Express* Interface Timings 59
4-15 Intel® C620 Series Chipset Clock Timings 61
4-16 USB 2.0 Timing 64
4-17 USB 3 Interface Transmit and Receiver Timings 65
4-18 SATA Interface Timings 66
4-19 SMBus and SMLink Timing 67
4-20 Intel® High Definition Audio (Intel® HD Audio) Timing 68
4-21 LPC Timing 68
4-22 Miscellaneous Timings 70
4-23 SPI Timings (17 MHz) 71
4-24 SPI Timings (30 MHz) 71
4-25 SPI Timings (40 MHz) 72
4-26 eSPI Timing 74
4-27 PM_SYNC Timing 75
4-28 NCSI Timing 75
4-29 Overshoot/Undershoot Specifications 76
4-30 Power Ramp with DSX Support 80
4-31 Power Ramp Without DSX Support 82
4-32 SLP_S5# High to PLTRST# High 83
4-33 S0 to S5 84
4-34 Powerdown to DeepSX 85
4-35 Surprise Powerdown to G3 86
4-36 eSPI Reset from G3 86
7-1 PCI Devices and Functions 91
7-2 Fixed I/O Ranges Decoded by PCH 92
7-3 Variable I/O Decode Ranges 94
7-4 PCH Memory Decode Ranges (from CPU perspective) 95
7-5 SPI Mode Address Swapping 97
8-1 Clock Signal Planes and States 101
8-2 Summary of Integrated Clock Controller FW Accessible Registers 103
9-1 General Power States for Systems Using the PCH 122
9-2 State Transition Rules for the PCH 122
9-3 System Power Plane 123
9-4 Causes of SMI and SCI 124
9-5 Sleep Types 128
9-6 Causes of Wake Events 128

9-7 Transitions Due to Power Failure 130
9-8 Supported Deep Sx Policy Configurations..... 130
9-9 Deep Sx Wake Events..... 131
9-10 Transitions Due to Power Button 132
9-11 Write Only Registers with Read Paths in ALT Access Mode..... 135
9-12 PIC Reserved Bits Return Values 135
9-13 Register Write Accesses in ALT Access Mode 136
9-14 SUSPWRDNACK/SUSWARN#/GPP_A13 Pin Behavior 139
9-15 SUSPWRDNACK During Reset 139
9-16 Causes of Host and Global Resets..... 140
9-17 Summary of Power Management Configuration Registers..... 144
9-18 Summary of PMC I/O Based Registers..... 159
9-19 Summary of PMC Memory Mapped Registers..... 178
11-1 Event Transitions that Cause Messages 215
11-2 Summary of SMBus TCO I/O Registers..... 218
12-1 Legacy Replacement Routing 229
12-2 Summary of HPET Memory Mapped Registers..... 232
13-1 Summary of Thermal Reporting Configuration Registers 242
13-2 Summary of Thermal Reporting Memory Mapped Registers 259
14-1 Counter Operating Modes..... 274
14-2 Summary of 8254 Timer Registers..... 275
15-1 Integrated Pull-Ups and Pull-Downs 282
15-2 I/O Signal Planes and States..... 282
15-3 Summary of High Definition Audio (D31:F3) PCI Configuration Registers 283
16-1 Summary of Processor Interface Memory Registers..... 322
17-1 eSPI Channels and Supported Transactions..... 329
17-2 eSPI Virtual Wires (VW) 330
17-3 Summary of Enhanced SPI (eSPI) Configuration Registers 335
17-4 Summary of eSPI Additional Registers 353
18-1 GPIO Group Summary 363
18-2 General Purpose I/O Signals 363
18-3 Summary of GPIO Community 0 Registers..... 372
18-4 Summary of GPIO Community 1 Registers..... 647
18-5 Summary of GPIO Community 2 Registers..... 897
18-6 Summary of GPIO Community 3 Registers..... 947
18-7 Summary of GPIO Community 4 Registers..... 997
18-8 Summary of GPIO Community 5 Registers.....1131
19-1 GbE LAN Signals 1386
19-2 Integrated Pull-Ups and Pull-Downs 1387
19-3 Power Plane and States for Output Signals 1387
19-4 Power Plane and States for Input Signals.....1388
19-5 LAN Mode Support 1391
19-6 Summary of GbE Configuration Registers 1392
19-7 Summary of GbE Memory Mapped I/O Registers.....1405
20-1 Interrupt Options - 8259 Mode.....1414
20-2 Interrupt Options - APIC Mode 1414
20-3 Interrupt Logic Signals.....1415
20-4 Interrupt Controllers PIC 1416
20-5 Interrupt Status Registers 1417
20-6 Content of Interrupt Vector Byte 1417
20-7 APIC Interrupt Mapping1.....1422
20-8 Stop Frame Explanation 1425
20-9 Data Frame Format 1426
20-10 Summary of Interrupt Registers 1427
20-11 Summary of Interrupt PCR Registers.....1438
20-12 Summary of APIC Indirect Registers 1450

20-13 Summary of Advanced Programmable Interrupt Controller (APIC) Registers 1466
21-1 LPC Interface Diagram 1468
21-2 LPC Cycle Types Supported 1469
21-3 Start Field Bit Definitions 1470
21-4 Cycle Type Bit Definitions 1470
21-5 Transfer Size Bit Definition 1470
21-6 SYNC Bit Definition 1471
21-7 Summary of LPC Configuration Registers 1472
21-8 Summary of LPC PCR Registers 1488
22-1 Power Plane and States for PCI Express* Signals 1493
22-2 PCI Express Port Feature Details 1494
22-3 PCI Express Link Configurations Supported 1494
22-4 MSI Versus PCI IRQ Actions 1495
22-5 NPx16 Port 1500
22-6 NPx8 1501
22-7 PCI Express Link Configurations Supported 1502
22-8 Summary of PCI Express Port Configuration Registers 1506
22-9 Summary of PCI Express Port Configuration Registers 1564
22-10 Summary of PCI Express Port Configuration Registers 1606
22-11 Summary of PCI Express Port Configuration Registers 1646
23-1 RTC Crystal Requirements 1689
23-2 External Crystal Oscillator Requirements 1690
23-3 Summary of RTC Indexed Registers 1690
23-4 Summary of RTC Configuration Registers 1695
24-1 Summary of SATA Configuration Registers 1714
24-2 Summary of SATA ABAR Registers 1742
24-3 Summary of SATA AIDP Registers 1781
24-4 Summary of SATA MXTBA Registers 1782
24-5 Summary of SATA MXPBA Registers 1784
24-6 Summary of SATA Configuration Registers 1785
24-7 Summary of SSATA ABAR Registers 1813
24-8 Summary of SATA AIDP Registers 1846
24-9 Summary of SATA MXTBA Registers 1848
24-10 Summary of SATA MXPBA Registers 1851
25-1 Summary of SMBus TCO I/O Registers 1854
26-1 I^2C* Block Read 1867
26-2 Enable for SMBALERT# 1869
26-3 Enables for SMBus Slave Write and SMBus Host Events 1870
26-4 Enables for the Host Notify Command 1870
26-5 Slave Write Registers 1871
26-6 Command Types 1871
26-7 Slave Read Cycle Format 1872
26-8 Data Values for Slave Read Registers 1873
26-9 Host Notify Format 1875
26-10 Slave Read Cycle Format 1875
26-11 Data Values for Slave Read Registers 1876
26-12 Enables for SMBus Slave Write and SMBus Host Events 1877
26-13 Summary of SMBus Configuration Registers 1877
26-14 Summary of SMBus I/O and Memory Mapped I/O Registers 1888
26-15 Summary of SMBus PCR Registers 1902
27-1 SPI Flash Regions 1906
27-2 Region Size Versus Erase Granularity of Flash Components 1907
27-3 Region Access Control Table 1909
27-4 Summary of SPI Configuration Registers 1911
27-5 Summary of SPI Memory Mapped Registers 1919
27-6 Summary of BIOS Flash Program Registers 1960

28-1 Testability Signals 1963
28-2 Power Planes and States for PCH Testability Signals 1964
28-3 Summary of Intel Trace Huh Configuration Registers....1967
29-1 Summary of xHCI Configuration Registers 1981
29-2 Summary of xHCI Memory Mapped Registers 2009
30-1 Summary of Intel Management Engine Interface Configuration Registers 2058
30-2 Summary of Intel MEI MMIO Registers 2077
31-1 Summary of IDE Redirect PCI Configuration (D22:F2) Registers 2078
32-1 Summary of Serial Port for Remote Keyboard and Text (KT) (D22:F3) Registers 2093
32-2 Summary of Keyboard and Text (KT) Additional Configuration Registers 2106
33-1 IE Host Root Space Devices 2112
33-2 IE Root Space Devices 2112
33-3 IE Features 2113
35-1 Summary of xHCI Configuration Registers 2119
35-2 Summary of MROM1 Configuration Registers 2135
36-1 Summary of FIA Configuration Registers 2150
36-2 Summary of DMI Configuration Registers 2158
36-3 Summary of DCI Configuration Registers 2185
36-4 Summary of DCI PCR Registers 2186
36-5 Summary of PSF1 PCR Registers 2187
36-6 Summary of PSF2 PCR registers 2198
36-7 Summary of PSF3 PCR Registers 2201
36-8 Summary of PSF4 PCR registers 2222
36-9 Summary of I/O Trap Registers 2223
36-10 Private Configuration Space Register Target Port IDs 2233
36-11 Summary of Internal Configuration Registers 2235
36-12 Intel® C620 Series Chipset PCH SBREG_BAR and SBI Address Offset Details 2259
37-1 Summary of Debug Registers....2261
38-1 10 GbE LAN Configurations 2284
38-2 PCIe* Host Interface Features....2296
38-3 Link Layer Ethernet Port Features 2297
38-4 Performance On The Network vs. Packet Size 2297
38-5 Transmit And Receive Scheduling 2298
38-6 LAN Engine Features....2299
38-7 Protocol Engine iWARP RDMA Features 2300
38-8 Protocol Engine User Space Direct Access Features 2302
38-9 Internal Switching Features 2302
38-10 System Manageability Features 2303
38-11 Power Management Features 2303
38-12 General Features....2304
38-13 Standards Supported By the 10 GbE controller 2306
38-14 Messages In the 10 GbE controller (As a Receiver) 2315
38-15 Messages In the 10 GbE controller (As a Transmitter) 2315
38-16 Response and Reporting Of PCIe Error Events 2321
38-17 PCIe* Statistic Events Encoding 2325
38-18 Link And Physical Layer Performance Events 2327
38-19 XP PM Compare Low Bits Register (XPPMCL[1:0]) (0x3288, 0x328c) 2328
38-20 XP PM Compare Low Bits Register (XPPMDL[1:0]) (0x32e8, 0x32f0) 2328
38-21 XP PM Data High Bits Register (XPPMDH[1:0]) (0x32ec, 0x32f4) 2328
38-22 XP PM Response Control Register (XPPMR[1:0]) (0x3294, 0x3298) 2328
38-23 XP PM Resource Events Register (XPPMER[1:0]) (0x32AC, 0x32B0) 2329
38-24 Performance Monitor Local Control Register (XPPERFCON) (0x32C4) 2329
38-25 XP PM Data Low Bits Register (XPPMDL[1:0]) (0x3280, 0x3284) 2329
38-26 NPQ Control Register - NPQC (0x00000) 2330
38-27 NPQ Round-trip Delay 1 Register - NPQRTDLY1 (0x00030; RO) 2331
38-28 NPQ Round-trip Delay 2 Register - NPQRTDLY2 (0x00034; RO) 2331

38-29 Resolution Of The Latency Counters 2331
38-30 Mac Port And Possible Speed Configurations 2334
38-31 Mac Control Frame Format 2336
38-32 802.3x Link Flow Control Frame Formats 2336
38-33 PFC Packet Format 2337
38-34 Format Of Priority Enable Vector 2337
38-35 Transfer of PAUSE Packet to Host (DPF bit) in 10 Gb/s MAC 2339
38-36 Pacing Speeds At 10 Gb/s Link Speed 2341
38-37 MAC Error Counters 2342
38-38 Clause 22 Basic MDIO Frame Format 2345
38-39 Clause 45 Indirect Addressing MDIO Frame Format 2345
38-40 I^2C Read Transaction - Dummy Write 2349
38-41 I^2C Read Transaction - Word Read 2349
38-42 I^2C Read Transaction - Dummy Write 2349
38-43 I^2C Read Transaction - Byte Read 2350
38-44 I^2C Write Transaction - Word Write 2350
38-45 I^2C Write Transaction - Byte Write 2350
38-46 Example for SDP, MDIO and I^2C Ports Usage for SFP+ and QSFP+ Modules 2352
38-47 OCP Use Cases 2356
38-48 OCP NVM Content 2358
38-49 OCP NVM content 2358
38-50 OCP Cards IDs 2359
38-51 Link Configuration Admin Commands (0x06xx) 2362
38-52 Set PHY Config Command (opcode: 0x0601) 2362
38-53 Set PHY Config Command Data Structure 2363
38-54 Set PHY Config Command Response (Opcode: 0x0601) 2364
38-55 Set MAC Config Command (Opcode: 0x0603) 2365
38-56 Set MAC Config Command Data Structure 2365
38-57 Set MAC Config Command Response (Opcode: 0x0603) 2366
38-58 Restart AN Command (Opcode: 0x0605) 2366
38-59 Restart Auto-negotiation Command Response (Opcode: 0x0605) 2367
38-60 Get PHY Abilities Command (Opcode: 0x0600) 2367
38-61 Get PHY Abilities Command Response (Opcode: 0x0600) 2368
38-62 Get PHY Abilities Command Response Data Structure 2368
38-63 Get Link Status Command (Opcode: 0x0607) 2370
38-64 Get Link Status Response (Opcode: 0x0607) 2371
38-65 Get Link Status Response Data Structure 2372
38-66 Reported Link Events 2374
38-67 Set Event Mask Command (Opcode: 0x0613) 2375
38-68 Set Event Mask Command Response (Opcode: 0x0613) 2375
38-69 PF-to-physical Lane Mappings (Supported Configurations) 2376
38-70 Clients And Access Types to the NVM 2392
38-71 NVM Access Admin Commands And Events 2398
38-72 NVM Read Admin Command 2399
38-73 NVM Read Response 2399
38-74 NVM Erase Admin Command 2400
38-75 NVM Erase Response 2401
38-76 NVM Erase Completion (on ARQ) 2401
38-77 NVM Update Admin Command 2402
38-78 NVM Update Response 2403
38-79 NVM Update Completion (on ARQ) 2404
38-80 NVM Config Read Admin Command 2405
38-81 NVM Config Read Response 2406
38-82 NVM Config Write Admin Command (0x0705) 2406
38-83 NVM Config Write Response (0x0705) 2407
38-84 Feature Buffer For NVM Config Read /Write 2407

38-85 Immediate Buffer for NVM Config Read /Write....2408
38-86 Small Resource Structure....2408
38-87 Large Resource Structure....2408
38-88 VPD Structure....2408
38-89 GPIO Pin Configuration Registers....2410
38-90 LED Source Modes....2411
38-91 NVM Section Loaded by Reset Source Covered by GLNVM_ULD Register....2418
38-92 Per PF Alt RAM Content....2437
38-93 List of AQ Commands for the Alternate Structure....2439
38-94 Write Alternate - Direct Command....2439
38-95 Completion For The Write Alternate - Direct Command....2440
38-96 Write Alternate - Indirect Command....2440
38-97 Completion For The Write Alternate - Indirect Command....2441
38-98 Read Alternate - Direct Command....2441
38-99 Completion For The Read Alternate - Direct Command....2441
38-100 Supported Mechanism....2456
38-101 Used Mechanism....2457
38-102 Magic Packet Structure....2468
38-103 Set WoL Filter Command (Opcode: 0x0120)....2475
38-104 Buffer Content (for ACPI Filters)....2475
38-105 Set WoL Filter Response (Opcode: 0x0120)....2476
38-106 Get Wake Up Reason Command (Opcode: 0x0121)....2476
38-107 Get Wake Up Reason Response (Opcode: 0x0121)....2476
38-108 Set Proxying Configuration Command Table Entry....2477
38-109 Set Proxying Configuration Command Descriptor Fields....2477
38-110 Set Proxying Configuration Command Descriptor Fields....2478
38-111 Set Proxying Configuration Response Descriptor Fields....2478
38-112 Set NS Proxy Table Entry Command Descriptor Fields....2478
38-113 IPv6 NS Table Entry Command Structure....2479
38-114 Set NS Proxy Table Entry Response Descriptor Fields....2480
38-115 IPv6 NS Table Entry Response Structure....2480
38-116 Hierarchical NVM map....2481
38-117 NVM Header Map....2484
38-118 NVM Integrity Summary Table....2499
38-119 NVM Header of Modules Mapped to Shadow RAM....2500
38-120 Header of Authenticated NVM Modules....2500
38-121 RO Commands Section Format....2501
38-122 RO Commands Types....2502
38-123 NVM General Summary Table....2503
38-124 Classification filters per PF/VF....2710
38-125 Filter actions by filter type....2711
38-126 Static Classification Filters Registers Initialization....2713
38-127 Dynamic Classification Filters Registers Initialization....2713
38-128 Packet Classifier Types and its Input Sets....2713
38-129 Flow Director PCTYPE Translation....2714
38-130 Flexible Payload (GLOQF_ORT[33:35] Registers)....2716
38-131 PRTQF_FLX_PIT Registers' Field....2717
38-132 GLQF_ORT Register Settings....2718
38-133 GLQF_PIT Registers Settings from NVM....2719
38-134 Protocol Layers Table (Type Rule Bus)....2720
38-135 Default Field Vector Table....2722
38-136 Detailed L2 Tags in the Field Vector (Assuming all L2 Tags are VLANs)....2723
38-137 Second or Fifth Priority Filters for Receive Queue Classification....2723
38-138 Tunneling UDP Admin Commands....2725
38-139 Supported Tunneling Types....2725
38-140 Network Key Structure of Tunneling Header....2726

38-141 PE Quad Hash Input Set 2734
38-142 Packet Types that Miss the PE Quad Hash Filter 2735
38-143 Examples of Input Fields for the Hash Function 2739
38-144 IPv4 2739
38-145 IPv6 2740
38-146 Add Tunneling UDP Command (Opcode: 0x0B00) 2741
38-147 Completion for the Add Tunneling UDP Command (Opcode: 0x0B00) 2742
38-148 Remove Tunneling UDP Port Command (Opcode: 0x0B01) 2743
38-149 Completion for Remove Tunneling UDP Command (Opcode: 0x0B01) 2743
38-150 Set RSS Key Command (Opcode: 0x0B02) 2743
38-151 Set RSS Key Response (Opcode: 0x0B02) 2744
38-152 Set RSS LUT Command (Opcode: 0x0B03) 2744
38-153 Set RSS LUT Response (Opcode: 0x0B03) 2745
38-154 Get RSS Key Command (Opcode: 0x0B04) 2745
38-155 Get RSS Key Response (Opcode: 0x0B04) 2746
38-156 Get RSS LUT Command (Opcode: 0x0B05) 2746
38-157 Get RSS LUT Response (Opcode: 0x0B05) 2747
38-158 Single Tag Handling 2750
38-159 Port-Based Tag Handling (Transmit) 2751
38-160 Tag Extraction Example 2753
38-161 L2 Tag Control Registers - Global 2754
38-162 L2 Tag Control Registers - Per VSI 2755
38-163 L2 Tags 2756
38-164 L2 Headers Support 2756
38-165 Port VLAN List and VSI Parameters 2759
38-166 List of Serial EtherType Headers 2765
38-167 List of Ethernet Types 2765
38-168 MPLS Header Structure 2766
38-169 Structure of LLDP Frame 2767
38-170 Structure of LLDP PDU 2767
38-171 Structure of LLDP TLV 2767
38-172 Structure of LLDP Organizationally Specific TLV 2767
38-173 List of Organizationally Specific TLV 2767
38-174 IPv6 Extension Headers and their Recommended Ordering 2769
38-175 Header Type Encoding and Lengths 2771
38-176 ICMP Packet Format 2771
38-177 IPv4 Packet Structure and Processing 2771
38-178 ICMPv6 Packet Format 2772
38-179 ICMPv6 Packet Structure and Processing 2773
38-180 ARP Packet Structure and Processing 2774
38-181 VXLAN Header Structure 2780
38-182 Geneve Header Structure 2780
38-183 Geneve Header Structure 2780
38-184 VXLAN-GPE Header Structure (Additive Fields on Top of Original VXLAN) 2781
38-185 NSH Header Structure 2782
38-186 MPLS Header Structure 2783
38-187 S-tag Format 2800
38-188 VSI Context 2817
38-189 Port VLAN List and VSI Parameters 2820
38-190 Private VLAN Membership 2823
38-191 External Elements 2845
38-192 Internal Elements 2846
38-193 Switching Structure Representation 2846
38-194 Switching Structure Representation - Examples 2847
38-195 Switching Elements Resources 2852
38-196 Switching Rules Resources 2852

38-197 Initial Elements....2856
38-198 Switch Configuration Admin Commands (0x02xx)....2858
38-199 Get Switch Configuration Command (Opcode: 0x0200)....2861
38-200 Get Switch Configuration Response (Opcode: 0x0200)....2861
38-201 Get Switch Configuration Command Response Buffer....2862
38-202 Get Switch Configuration Command Response Header....2862
38-203 Get Switch Configuration (Switch Element)....2862
38-204 Add Statistics Command (Opcode: 0x0201)....2863
38-205 Add Statistics Response (Opcode: 0x0201)....2863
38-206 Remove Statistics Command (Opcode: 0x0202)....2864
38-207 Remove Statistics Response (Opcode: 0x0202)....2864
38-208 Set Port Parameters Command (Opcode: 0x0203)....2864
38-209 Set Port Parameters Response (Opcode: 0x0203)....2865
38-210 Get Switch Resources Allocation Command (Opcode: 0x0204)....2865
38-211 Get Switch Resources Response (Opcode: 0x0204)....2866
38-212 Return Buffer Structures for Resource Type Format....2866
38-213 Set Switch Configuration command (Opcode: 0x0205)....2867
38-214 Set Switch Configuration response (Opcode: 0x0205)....2867
38-215 Add VSI Command (Opcode: 0x0210)....2868
38-216 Add VSI Command Buffer....2869
38-217 Add VSI Response Buffer....2874
38-218 Add VSI Response (Opcode: 0x0210)....2874
38-219 Add VSI Recommended Settings....2875
38-220 Update VSI Command (Opcode: 0x0211)....2877
38-221 Update VSI Response (Opcode: 0x0211)....2877
38-222 Get VSI Parameters Response (Opcode: 0x0212)....2878
38-223 Get VSI Parameters Command (Opcode: 0x0212)....2878
38-224 Get VSI Parameters Response Buffer....2878
38-225 Add Port Virtualizer Command (Opcode: 0x0220)....2879
38-226 Add Port Virtualizer Response (Opcode: 0x0220)....2880
38-227 Update Port Virtualizer Command (Opcode: 0x0221)....2880
38-228 Update Port Virtualizer Response (Opcode: 0x0221)....2881
38-229 Get Port Virtualizer Command (Opcode: 0x0222)....2881
38-230 Get Port Virtualizer Response (Opcode: 0x0222)....2881
38-231 Add VEB Command (Opcode: 0x0230)....2882
38-232 Add VEB Response (Opcode: 0x0230)....2883
38-233 Get VEB Parameters Command (Opcode: 0x0232)....2883
38-234 Get VEB Parameters Response (Opcode: 0x0232)....2884
38-235 Delete Element Command (Opcode: 0x0243)....2884
38-236 Delete Element Response (Opcode: 0x0243)....2885
38-237 Add MAC, VLAN Pair Command (Opcode: 0x0250)....2885
38-238 MAC, VLAN Pairs to Add Detail....2886
38-239 Add MAC, VLAN Pair Response (Opcode: 0x0250)....2886
38-240 MAC, VLAN Pairs Allocation Detail....2887
38-241 Remove MAC, VLAN Pair Command (Opcode: 0x0251)....2887
38-242 Remove MAC, VLAN Pair Response (Opcode: 0x0251)....2888
38-243 MAC, VLAN Pairs to Remove Detail....2889
38-244 Response Buffer Results (MAC, VLAN Pairs Removal)....2889
38-245 Add VLAN Command (Opcode: 0x0252)....2889
38-246 Add VLAN Response (Opcode: 0x0252)....2890
38-247 Command Buffer - VLAN IDs to Add Detail....2891
38-248 Response Buffer - VLANs Allocation Detail....2891
38-249 Remove VLAN Command (Opcode: 0x0253)....2891
38-250 Command Buffer - VLAN Pairs to Remove Detail....2892
38-251 Remove VLAN Response (Opcode: 0x0253)....2892
38-252 Response Buffer - MAC, VLAN Pairs Removal Detail....2893

38-253 Set VSI Promiscuous Modes Command (Opcode: 0x0254) 2893
38-254 Set VSI Promiscuous Modes Response (Opcode: 0x0254) 2894
38-255 Add S-tag Command (Opcode: 0x0255) 2894
38-256 Add S-tag Response (Opcode: 0x0255) 2895
38-257 Remove S-tag Response (Opcode: 0x0256) 2896
38-258 Remove S-tag Command (Opcode: 0x0256) 2896
38-259 Update S-tag Command (Opcode: 0x0259) 2897
38-260 Update S-tag Response (Opcode: 0x0259) 2897
38-261 Add Control Packet Filter Command (Opcode: 0x025A) 2898
38-262 Add Control Packet Filter Response (Opcode: 0x025A) 2899
38-263 Remove Control Packet Filter Command (Opcode: 0x025B) 2899
38-264 Remove Control Packet Filter Response (Opcode: 0x025B) 2900
38-265 Add Cloud Filters Command (Opcode: 0x025C) 2900
38-266 Add Cloud Filters Response (Opcode: 0x025C) 2901
38-267 Add Cloud Filters - Command Buffer 2901
38-268 Add Cloud Filters - Response Buffer 2903
38-269 Remove Cloud Filters Command (Opcode: 0x025D) 2903
38-270 Remove Cloud Filters Response (Opcode: 0x025D) 2903
38-271 Remove Cloud Filters - Response Buffer 2904
38-272 Clear All WoL Switch Filters Command (Opcode: 0x0025E) 2904
38-273 Clear All WoL Switch Filters Response (Opcode: 0x025E) 2904
38-274 Add Mirror Rule Command (Opcode: 0x0260) 2905
38-275 Add Mirror Rule Command Buffer 2906
38-276 Add Mirror Rule Response (Opcode: 0x0260) 2906
38-277 Delete Mirror Rule Command (Opcode: 0x0261) 2906
38-278 Delete Mirror Rule - Command Buffer 2907
38-279 Delete Mirror Rule Response (Opcode: 0x0261) 2907
38-280 MSI-X Vector Allocation per Function 2908
38-281 Other Interrupt Causes of the PFs 2913
38-282 Other interrupt causes of the VFs 2913
38-283 Product Name Versus the 82599 Virtualization Support 2922
38-284 Virtualization Features Enablement 2923
38-285 Data Checks on Fetches for Tx Data 2924
38-286 Tx Descriptor Validity Checks 2924
38-287 Rx Data Fetch Checks 2925
38-288 VF Resource Allocation 2926
38-289 Example #1: 1 No-drop TC + 3 Drop TCs 2943
38-290 Example #2: 2 No-drop TCs + 3 Drop TCs 2943
38-291 Example #3: 8 No-drop TCs 2944
38-292 Example #4: 4 No-drop TCs + 2 Drop TCs 2944
38-293 Local DCBX Managed Objects 2955
38-294 PFC Managed Objects 2956
38-295 Changes in DCB Registers Made via NVM Settings When Port Operated at Different Link Speed 2960
38-296 PFC Ignore Response 2960
38-297 PFC Ignore Command 2961
38-298 DCB Updated Response 2962
38-299 DCB Updated Command 2962
38-300 LAN Queue Overflow Event 2962
38-301 10 GbE Ethernet Controller bandwidth Management Attributes 2965
38-302 Scheduler Configuration Admin Queue Commands 2976
38-303 Transmit Scheduler Admin Queue Completion Errors 2980
38-304 Configure VSI Bandwidth Limit Command Fields 2981
38-305 Configure VSI Bandwidth Limits per Traffic Type Command Fields 2982
38-306 Configure VSI Bandwidth Limit per Traffic Type Command Buffer 2983
38-307 Configure VSI Bandwidth Limit per Traffic Type Completion Buffer 2984

38-308 Configure VSI Bandwidth Allocation per Traffic Type Command Fields....2985
38-309 Configure VSI Bandwidth Allocation per Traffic Type Command Buffer....2985
38-310 Configure VSI Bandwidth Allocation per Traffic Type Completion Buffer....2986
38-311 Configure Switching Component Bandwidth Limit Command Fields....2987
38-312 Configure Physical Port ETS Command Fields....2988
38-313 Configure Physical Port ETS Command Buffer....2988
38-314 Suspend Port's Tx Traffic....2989
38-315 Resume PF Traffic....2990
38-316 Configure Switching Component Bandwidth Limits per Traffic Type Command Fields....2991
38-317 Configure Switching Component Bandwidth Limit per Traffic Type Command Buffer....2991
38-318 Configure Switching Component Bandwidth Allocation per Traffic Type Command Fields....2993
38-319 Configure Switching Component Bandwidth Allocation per Traffic Type Command Buffer....2993
38-320 Query VSI Bandwidth Configuration Command Fields....2994
38-321 Query VSI Configuration Completion Buffer....2994
38-322 Query VSI Bandwidth Allocation per Traffic Type Command Fields....2995
38-323 Query VSI Bandwidth Allocation per Traffic Type Completion Buffer....2996
38-324 Query Switching Component Configuration Command Fields....2997
38-325 Query Switching Component Configuration Completion Buffer....2997
38-326 Query Physical Port ETS Configuration Command Fields....2998
38-327 Query Physical Port ETS/SLA Completion Buffer....2998
38-328 Query Switching Component Bandwidth Allocation per Traffic Type Command Fields....3000
38-329 Query Switching Component Bandwidth Allocation per Traffic Type Completion Buffer....3000
38-330 Scheduler Configuration Summary....3002
38-331 ETS-Based Scheme-relative Bandwidth Calculation Example....3014
38-332 HMC Objects....3016
38-333 HMC page descriptor format....3020
38-334 10 GbE Controller HMC Resource Profiles....3024
38-335 Default HMC Profile Examples....3025
38-336 Initial 10 GbE Controller Host Memory Cache Partitioning per PF....3025
38-337 Example SR-IOV VF Primary Profile Resource Partitioning per FPM....3026
38-338 SR-IOV Even Distribution Profile Resource Partitioning Per FPM....3027
38-339 FPM Object Registers....3028
38-340 FPM Object Order and Alignment....3029
38-341 HMC Errors....3034
38-342 Admin Queue Descriptor Structure (in LE 32 Order)....3037
38-343 Admin Descriptor Field descriptions....3038
38-344 AQ Registers....3038
38-345 Receive Queue Element - Initial Values....3040
38-346 Direct Admin Command Structure....3041
38-347 Direct Command Completion Event Template....3042
38-348 Indirect Command Template....3042
38-349 Indirect Command Completion....3043
38-350 Commands Allowed for VF Firmware Admin Queues....3044
38-351 Non-Completion Events....3044
38-352 Admin Queue Return Value Fields....3045
38-353 Admin Queue Return Values and Error Codes....3045
38-354 opcode groups....3046
38-355 Generic Commands....3046
38-356 Get Version Command (Opcode: 0x0001)....3047
38-357 Driver Version Command (Opcode: 0x0002)....3047
38-358 Driver Version Buffer....3048
38-359 Queue Shutdown Command (Opcode: 0x0003)....3048
38-360 Set PF Context Command (Opcode: 0x0004)....3049
38-361 Shared Resources....3049

38-362 Request Resource Ownership Admin Command (Opcode: 0x0008) 3049
38-363 Release Resource Ownership Admin Command (Opcode: 0x0009) 3050
38-364 Discover Device/function Capabilities (Opcode: 0x000A, 0x000B) 3050
38-365 Capability Structure 3051
38-366 Resources Recognized by This Version of the Command 3051
38-367 Configure No-drop Policy (Opcode: 0x0112) 3054
38-368 Virtualization Admin Commands 3054
38-369 Send to PF Command (Opcode: 0x0801) 3054
38-370 Message From VF Event (Opcode: 0x0801) 3055
38-371 Send to VF Command (Opcode: 0x0802) 3055
38-372 Message From PF Event (Opcode: 0x0802) 3056
38-373 Counter Name Prefixes 3056
38-374 Ethernet Interface Group Statistics Counters 3058
38-375 Error Counters Not Implemented in Current Design 3059
38-376 Additional Per Port Counters 3060
38-377 VEB Per VLAN Counters 3061
38-378 VEB Per UP Counters 3061
38-379 VEB Per VSI Counters 3061
38-380 Other Statistics 3062
38-381 Forwarding Rules for Rx LLDP Packets 3064
38-382 LLDP Admin Queue Commands 3068
38-383 Get LLDP MIB Command 3069
38-384 Get LLDP MIB Response 3070
38-385 Configure LLDP MIB Change Event Command 3071
38-386 Configure LLDP MIB Change Event Response 3071
38-387 LLDP MIB Change Event 3072
38-388 Add LLDP TLV Command 3072
38-389 Add LLDP TLV Response 3073
38-390 Update LLDP TLV Command 3074
38-391 Update LLDP TLV Response 3074
38-392 Delete LLDP TLV Command 3075
38-393 Delete LLDP TLV Response 3076
38-394 Stop LLDP Agent Command 3077
38-395 Stop LLDP Agent Response 3077
38-396 Start LLDP Agent Command 3078
38-397 Start LLDP Agent Response 3078
38-398 Restore LLDP Agent Factory Settings Command 3078
38-399 Restore LLDP Agent Factory Settings Response 3079
38-400 Get CEE DCBX OPER CFG Command 3079
38-401 Get CEE DCBX OPER CFG Response 3080
38-402 Get CEE DCBX OPER CFG Response Buffer Format 3080
38-403 Set local LLDP MIB Command 3082
38-404 Set Local LLDP MIB Response 3082
38-405 Stop/Start a Specific LLDP Agent Command 3083
38-406 Stop/Start a Specific LLDP Agent Command Response 3083
38-407 5 VSIs While the PF Owns 128 LQPs Example 3088
38-408 Clear PXE Mode Admin Command (Opcode: 0x0110) 3089
38-409 Clear PXE Mode Admin Command Completion (Opcode: 0x0110) 3090
38-410 Tunneled Packet Processing 3091
38-411 Steering Tag and Processing Hint Programming by the LAN Engine 3092
38-412 16-Byte Receive Descriptors Read Format 3094
38-413 32-Byte Receive Descriptors Read Format 3094
38-414 Status Bits 3095
38-415 Error Bits 3096
38-416 Mirroring Functionality 3097
38-417 Length Bits 3097

38-418 Packet Types 3097
38-419 Ext_Status Bits 3100
38-420 L2 Tag Bits 3101
38-421 Flexible Bytes Low Bits 3101
38-422 FD Filter Programming Indications 3102
38-423 Status Field Bits 3102
38-424 Receive Queue Enable / Disable Flags 3103
38-425 LAN Rx Queue Context in the Private Host Memory (174 Bits = 25 Bytes) 3106
38-426 Header Split Modes Defined by the DTYPE Field 3109
38-427 Split Headers Enabled by the HSPLIT_0 Field 3110
38-428 Split Headers Enabled by the HSPLIT_1 Field 3110
38-429 Header Split vs. Packet and Buffer Sizes 3110
38-430 Integrity Offload Check for Receive Packet Types 3112
38-431 LAN Descriptor Types 3113
38-432 Transmit Buffer 3115
38-433 Transmit Queue Enable / Disable Flags 3120
38-434 LAN Tx Queue Context in the Private Host Memory 3123
38-435 Pseudo Header Pre-Loaded Values 3127
38-436 Transmit Integrity Offload for Packet Types 3128
38-437 Packet Time Stamp Latency 3136
38-438 PRTTSYN_INC as a Function of MAC Clock Frequency 3137
38-439 Receive Time Stamp Control Parameters 3140
38-440 PTP Header Format (V1 and V2 Formats) 3141
38-441 PTP V2 and V1 Message Types 3141
38-442 PE features 3143
38-443 Doorbell Page Register Summary 3158
38-444 QP WR Size Requirements 3159
38-445 iWARP QP State Behavior 3170
38-446 iWARP QP State Transition 3171
38-447 iWARP QP State Transition Driven from the Wire 3173
38-448 PE CEQ Entry Format 3175
38-449 PE AEQ Entry Format 3175
38-450 PE AE codes (AE_ID) 3178
38-451 STag and Processing Hint Programming 3183
38-452 PE Objects Dependencies 3185
38-453 PE Resource Initialization 3186
38-454 RDMA Resource Initialization 3187
38-455 UDA Resource Initialization 3188
38-456 CQP Context Format 3190
38-457 CQP Error Codes 3192
38-458 CQP CQ Entry Format 3194
38-459 CQP Operations 3195
38-460 CQP Common WQE Fields 3196
38-461 CQP Create/Modify/Discard QP WQE Format 3196
38-462 Product Name iWARP QP State Transitions 3200
38-463 QP Doorbell Shadow Area 3200
38-464 CQP Create/Modify/Discard CQ WQE Format 3200
38-465 CQ Doorbell Shadow Area 3203
38-466 CQP Allocate/Register/Register Shared/De-Allocate STag WQE Format 3203
38-467 WQE Fields Valid for Allocate/Register/Register Shared/De-Allocate STAG 3206
38-468 CQP Query STag WQE Format 3207
38-469 Allocate Local MAC and IP Address Entry WQE Format 3207
38-470 Manage Local MAC WQE Format 3208
38-471 Manage ARP Table WQE Format 3209
38-472 CQP Manage VF PBLE Backing Pages WQE Format 3210
38-473 CQP Manage Push Page WQE Format 3211

38-474 CQP Upload Context WQE Format 3212
38-475 Uploaded Context Structure Format 3213
38-476 CQP Manage HMC PM Function Table WQE Format..... 3214
38-477 CQP Create/Discard CEQ WQE Format 3214
38-478 CQP Create/Discard AEQ WQE Format 3216
38-479 CQP Update PE SDs WQE Format 3217
38-480 Update SD Host Memory Structure..... 3218
38-481 CQP Query FPM Values WQE Format..... 3219
38-482 FPM Query Configuration Structure Format..... 3219
38-483 CQP commit FPM Values WQE Format..... 3221
38-484 FPM Commit Configuration Structure Format 3221
38-485 CQP Flush WQEs WQE Format 3222
38-486 CQP Manage Accelerated Port Table WQE Format 3224
38-487 CQP NOP WQE Format..... 3224
38-488 CQP Manage Quad Hash Table WQE Format 3226
38-489 CQP Suspend QP WQE Format 3228
38-490 CQP Resume QP WQE Format..... 3228
38-491 Static HMC Resources Allocated WQE Format..... 3229
38-492 iWARP Q2 Structure Format 3230
38-493 iWARP QP Context Structure Format..... 3230
38-494 iWARP QP Error Codes..... 3242
38-495 iWARP CQ Entry Format Without Immediate Data 3242
38-496 iWARP QP Operations 3244
38-497 iWARP Common WQE Fields 3245
38-498 iWARP Additional Fragment Descriptor Format 3246
38-499 iWARP SQ NOP WQE Format 3247
38-500 iWARP SQ Send WQE Format 3247
38-501 iWARP SQ Send With Inline Data WQE Format 3248
38-502 iWARP SQ RDMA write WQE Format 3249
38-503 iWARP SQ RDMA Write With Inline Data WQE Format 3250
38-504 iWARP RDMA Read WQE Format 3251
38-505 iWARP SQ Bind Memory Window WQE Format 3251
38-506 iWARP Local Invalidate WQE Format..... 3252
38-507 iWARP SQ Fast Register WQE Format 3254
38-508 iWARP SQ Established WQE Format..... 3256
38-509 iWARP RQ WQE Format 3256
38-510 UDA QP Context Structure Format 3264
38-511 UDA CQ Entry Format 3269
38-512 UDA QP Error Codes..... 3271
38-513 UDA SQ Send WQE Format 3272
38-514 UDA SQ Send With Inline Data WQE Format 3274
38-515 UDA SQ Send WQE Format 3275
38-516 UDA RQ WQE Format 3276
38-517 Summary of MIBs Supported With Product Name Hardware Statistics Counters..... 3276
38-518 PE Statistics Instantiated Per-PCI Function 3277
38-519 PE Intel-Specific Statistics Instantiated Once 3278
38-520 SMBus PEC Modes 3283
38-521 Filters Accessible to MC 3286
38-522 Exclusive Traffic Behavior 3295
38-523 PRT_MNG_MNGONLY Register Description and Usage 3295
38-524 PRT_MNG_MANC Bits Impact 3296
38-525 Filtering Programming Interfaces 3297
38-526 Typical SMBus Transaction..... 3305
38-527 Shorthand Field Names 3305
38-528 UDID 3308
38-529 Flag Options During Transmit Packet Transactions..... 3313

38-530 Management control command parameters/content 3321
38-531 Management Receive Filter Parameters 3321
38-532 Filter enable parameters 3323
38-533 Set Common Filters Receive Control Bytes 3324
38-534 SMBus Read Transactions 3326
38-535 CO LAN Packet Status Data 3327
38-536 Packet Status Information 3328
38-537 MNG Status 3328
38-538 Status Data Byte 1 3329
38-539 Status Data Byte 2 3329
38-540 Get Controller Information Data 3333
38-541 Example 1 MDEF Results 3336
38-542 Example 2 MDEF Results 3338
38-543 Example 3 MDEF Results 3340
38-544 Example 4 MDEF Results 3342
38-545 NC-SI Terminology 3345
38-546 Supported NC-SI Commands 3349
38-547 Optional NC-SI features support 3350
38-548 Get NC-SI PT Statistics Generation 3351
38-549 OEM Specific Command Response Reason Codes 3354
38-550 OEM Commands Summary 3355
38-551 Filter data group 3359
38-552 Filter Values 3363
38-553 Extended Filter 1 Values 3363
38-554 Filter Data Group 3367
38-555 Packet Reduction Field Description 3371
38-556 Extended Packet Reduction Field Description 3371
38-557 Set IP Address Command Packet Format 3381
38-558 IP filters Pools 3381
38-559 Set IP Flag Field 3382
38-560 Set IP Address Response Packet Format 3382
38-561 Get IP Address Command Packet Format 3382
38-562 Get IP Address Response Packet Format 3383
38-563 Get IP Flag Field 3383
38-564 Set Port Command Packet Format 3384
38-565 Set Port Flags Field Descriptions 3385
38-566 Set Port Response Packet Format 3385
38-567 Get Port Command Packet Format 3385
38-568 Get Port Command Field Descriptions 3385
38-569 Get Port Response Packet Format 3386
38-570 Get Port Flags Field Descriptions 3386
38-571 Enable Unicast Infrastructure Filter Command 3387
38-572 Unicast Infrastructure Packet Filter Settings Field 3387
38-573 Enable Unicast Infrastructure Filter Response Packet Format 3389
38-574 Get Shared IP Capabilities Command Packet Format 3390
38-575 Get Shared IP Capabilities Response Packet Format 3390
38-576 Filtering Capabilities Field 3391
38-577 Shared IP Enable Broadcast Filtering Command 3391
38-578 Shared IP Broadcast Packet Filter Settings Field 3392
38-579 Shared IP Enable Broadcast Filtering Packet Format 3392
38-580 Shared IP Enable Global Multicast Filtering Command 3393
38-581 Shared IP Multicast Packet Filter Settings Field 3393
38-582 Shared IP Enable Global Multicast Filtering Packet Format 3393
38-583 Get Shared IP Parameters Command Packet Format 3394
38-584 Get Shared IP Parameters Response Packet Format 3394
38-585 IP Address Flags Field 3394

38-586 Port Flags Field 3395
38-587 Set Binding Command Packet Format 3396
38-588 Set Binding Flags Field Descriptions 3396
38-589 Set binding response packet format 3397
38-590 Get Binding Command Packet Format 3397
38-591 Get Binding Response Packet Format 3398
38-592 Set Shared Mode Command Packet Format 3398
38-593 Set Shared Mode Response Packet Format 3399
38-594 Status Byte Description 3402
38-595 Controller Information Items 3403
38-596 NC-SI/Ethernet Over MCTP Over PCIe Message Format 3417
38-597 NC-SI/Ethernet Over MCTP Over SMBus Message Format 3418
38-598 NC-SI Over MCTP Over PCIe/SMBus Message Format 3420
38-599 Ethernet Over MCTP Over PCIe/SMBus Message Format 3421
38-600 MCTP Commands Support 3421
38-601 Address Space Regions 3424
38-602 IOADDR and IODATA in I/O Address Space 3425
38-603 IOADDR and IODATA in Configuration Address Space 3426
38-604 Structure of the PF memory BAR 3428
38-605 Structure of the PF Memory BAR 3428
38-606 Structure of the VF Memory BAR 3429
38-607 Structure of the VF memory BAR 3429
38-608 Scope Mapping 3430
38-609 BAR0 Registers Summary 3432
38-610 BAR3 Registers Summary 3680
38-611 VF Registers Summary 3683
38-612 BAR0 Registers Summary 3685
38-613 BAR3 Registers Summary 3698
38-614 RID Per PF - ARI Mode 3700
38-615 RID Per PF - non-ARI Mode 3700
38-616 PCI Capabilities Supported by Function 3701
38-617 PCI Configuration Space - PF 3703
38-618 Configuration Sharing of PCI Configuration Space 3703
38-619 Product Name BAR Description 3707
38-620 Product Name base Address Setting in 64-bit BARs Mode 3707
38-621 Base Address Registers' Fields 3708
38-622 PCI Capabilities List 3708
38-623 Sharing the Power Management Capability Registers 3709
38-624 Configuration Sharing of the MSI Capability 3711
38-625 Configuration Sharing of the MSI-X Capability 3714
38-626 Configuration Sharing of the PCIe Capability 3718
38-627 Extended Capabilities List 3730
38-628 Configuration Sharing of the AER Capability 3731
38-629 Configuration Sharing of the Serial Number Capability 3736
38-630 Configuration Sharing of the ARI Capability 3738
38-631 Configuration Sharing of the SR-IOV Capability 3740
38-632 Configuration Sharing of the TPH Requester Capability 3746
38-633 RID Per VF - ARI Mode 3749
38-634 RID Per VF - Non ARI Mode 3749
38-635 VF PCIe Configuration Space 3750
38-636 VF BARs in Product Name 3752
38-637 Extended Capabilities List 3753
39-1 Ring Assignment to Intel QuickAssist Technology Virtual Functions 3766
39-2 PCIe Commands Supported by the Receive Interface 3769
39-3 PCIe Endpoint Function Mapping 3770
39-4 Endpoint Mapping of MMIOs to BARs 3770

39-5 ARI: Bus, Device and Function Interpretation 3771
39-6 PCIe Error Sources 3774
39-7 Device Specific Errors 3777
39-8 Register Attributes / Modifiers 3778
39-9 Bus M, Function 0: Summary of PCIe Intel QuickAssist Technology Configuration Registers 3780
39-10 PVID - PF Vendor Identification Register 3783
39-11 PDID - PF Device Identification Register 3783
39-12 PPCICMD - PF Device Command Register 3784
39-13 PPCISTS - PF PCI Device Status Register 3785
39-14 PRID - PF Revision ID and Class Code Register 3786
39-15 PCLS - PF Cache Line Size Register 3786
39-16 PHDR - PF Header Type Register 3787
39-17 PMISCLBAR - PF Miscellaneous Lower Base Address Register 3787
39-18 PMISCUBAR - PF Miscellaneous Upper Base Address Register 3788
39-19 PETRINGCSRLBAR - PF Ring CSR Lower Base Address Register 3788
39-20 PETRINGCSRUBAR - PF Ring CSR Upper Base Address Register 3789
39-21 PSVID - PF Subsystem Vendor ID Register 3789
39-22 PSID - PF Subsystem ID Register 3790
39-23 PCP - PF Capabilities Pointer Register 3790
39-24 PIRQL - PF Interrupt Line Register 3791
39-25 PIRQP - PF Interrupt Pin Register 3791
39-26 PMSIX - PF Message Signaled Interrupt X Capability ID Register 3792
39-27 PMSIXNCP - PF MSIX Next Capability Pointer Register 3792
39-28 PMSIXCNTL - PF Message Signaled Interrupt X Control Register 3793
39-29 PMSIXTBIR - PF MSI-X Table Offset and Table BIR Register 3793
39-30 PMSIXPBABIR - PF MSI-X Pending Bit Array and BIR Offset Register 3794
39-31 PPMCAP - PF Power Management Capabilities ID Register 3794
39-32 PPMCP - PF Power Management Next Capability Pointer Register 3795
39-33 PPMC - PF Power Management Capabilities Register 3795
39-34 PPMCSR - PF Power Management Control and Status Register 3796
39-35 PPCID - PF PCI Express Capability Register 3798
39-36 PPCP - PF PCI Express Next Capability Pointer Register 3798
39-37 PPCR - PF PCI Express Capabilities Register 3799
39-38 PPDCAP - PF PCI Express Device Capabilities Register 3799
39-39 PPDCNTL - PF PCI Express Device Control Register 3801
39-40 PPDSTAT - PF PCI Express Device Status Register 3802
39-41 PLCAPR - PF Link Capabilities Register 3803
39-42 PLCNTLR - PF Link Control Register 3805
39-43 PLSR - PF Link Status Register 3807
39-44 PDCAPR2 - PF Device Capabilities 2 Register 3808
39-45 PDCNTR2 - PF Device Control 2 Register 3809
39-47 PLCNTLR2 - PF Link Control 2 Register 3810
39-46 PLCAPR2 - PF Link Capabilities 2 Register 3810
39-48 PLSR2 - PF Link Status 2 Register 3811
39-49 PMSICID - PF Message Signaled Interrupt Capability ID Register 3812
39-50 PMSINCP - PF Message Signaled Interrupt Next Capability Pointer Register 3812
39-51 PMSICTL - PF Message Signaled Interrupt Control Register 3813
39-52 PMSILADDR - PF Message Signaled Interrupt Lower Address Register 3813
39-53 PMSIUADDR - PF Message Signaled Interrupt Upper Address Register 3814
39-54 PMSIDATA - PF Message Signaled Interrupt Data Register 3814
39-55 PMSIMSK - PF Message Signaled Interrupt Mask Register 3815
39-56 PMSIPND - PF Message Signaled Interrupt Pending Register 3815
39-57 PPCIEAERCAPID - PF PCI Express AER Capability ID Register 3816
39-58 PPAERUCS - PF PCI Express AER Uncorrectable Error Status Register 3816
39-59 PPAERUCM - PF PCI Express AER Uncorrectable Error Mask Register 3818

39-60 PPAERUCSEV - PF PCI Express AER Uncorrectable Error Severity Register 3819
39-61 PPAERCS - PF PCI Express AER Correctable Error Register..... 3820
39-62 PPAERCM - PF PCI Express AER Correctable Error Mask Register 3821
39-63 PPAERCTLCAP - PF PCI Express AER Control and Capability Register 3822
39-64 PPAERHDRLOG0 - PF PCI Express AER Header Log 0 Register 3822
39-65 PPAERHDRLOG1 - PF PCI Express AER Header Log 1 Register 3823
39-66 PPAERHDRLOG2 - PF PCI Express AER Header Log 2 Register 3823
39-67 PPAERHDRLOG3 - PF PCI Express AER Header Log 3 Register 3824
39-68 PARIDHDR - PF Alternative Routing ID Capability Header 3824
39-69 PFARICAP - PF ARI Capabilities Register..... 3825
39-70 PARIDCTL - PF Alternative Routing ID Control Register 3825
39-71 PSRIOVCAPID - PF SR IOV Capability ID Register..... 3826
39-72 PSRIOVCVNC - PF SRIOV Capability Version and Next Capability Pointer Register 3826
39-73 PSRIOVCAP - PF SRIOV Capabilities Register 3827
39-74 PSRIOVCS - PF SRIOV Control and Status Register..... 3827
39-75 PSRIOVMTOTINI - PF SRIOV Initial and Total VF's Register..... 3828
39-76 PSRIOVNUMVF - PF SRIOV Number of VF's Register 3828
39-77 PSRIOVFVFO - PF SRIOV First VF Offset Register 3829
39-78 PSRIOVVFS - PF SRIOV VF Stride Register 3829
39-79 PSRIOVFDID - PF SRIOV VF Device ID Register..... 3830
39-80 PSRIOVPAGESIZE - PF SRIOV Supported Page Size Register..... 3830
39-81 PSRIOVSYSPS - PF SRIOV System Page Size Register..... 3831
39-82 PSRIOVLBAR0 - SRIOV Lower BAR0 Register..... 3832
39-83 PSRIOVUBAR0 - SRIOV Upper BAR0 Register 3833
39-84 PSRIOVLBAR1 - SRIOV Lower BAR1 Register..... 3833
39-85 PSRIOVUBAR1 - SRIOV Upper BAR1 Register 3834
39-86 PSRIOVVFMA - PF SRIOV VF Migration Array Register..... 3834
39-87 PACSCAPID - PF ACS Capability ID Register 3835
39-88 PACSCVNC - PF ACS Capability Version and Next Capability Pointer Register..... 3835
39-89 PACSCAP - PF ACS Capabilities Register..... 3836
39-90 VF PCIe Configuration Space 3837
39-91 Bus M, Function 8+Index 1: Summary of PCIe Intel QuickAssist Technology VF Configuration Registers..... 3839
39-92 VVID[0:15] - VF Vendor Identification Register..... 3841
39-93 VDID[0:15] - VF Device Identification Register 3841
39-94 VPCICMD[0:15] - VF Device Command Register..... 3842
39-95 VPCISTS[0:15] - VF PCI Device Status Register..... 3843
39-96 VRID[0:15] - VF Revision ID Register..... 3844
39-97 VHDR[0:15] - VF Header Type Register 3844
39-98 VSVID[0:15] - VF Subsystem Vendor ID Register..... 3844
39-99 VSID[0:15] - VF Subsystem ID Register 3845
39-100 VCP[0:15] - VF Capabilities Pointer Register..... 3845
39-101 VIRQL[0:15] - VF Interrupt Line Register 3845
39-102 VIRQP[0:15] - VF Interrupt Pin Register 3846
39-103 VPCID[0:15] - VF PCI Express Capability ID Register..... 3846
39-104 VPCP[0:15] - VF PCI Express Next Capability Pointer Register..... 3846
39-105 VPCR[0:15] - VF PCI Express Capabilities Register 3847
39-106 VPDCAP[0:15] - VF PCI Express Device Capabilities Register 3847
39-107 VPDC[0:15] - VF PCI Express Device Control Register..... 3848
39-108 VPDS[0:15] - VF PCI Express Device Status Register..... 3849
39-109 VLCR[0:15] - VF Link Capabilities Register 3850
39-110 VLCNTLR[0:15] - VF Link Control Register..... 3852
39-111 VLSR[0:15] - VF Link Status Register 3852
39-112 VDCAPR2[0:15] - VF Device Capabilities 2 Register 3853
39-113 VLCR2[0:15] - VF Link Capabilities 2 Register..... 3853
39-114 VLCNTLR2 [0:15] - VF Link Control 2 Register 3854

39-115 VLSR2[0:15] - VF Link Status Register 3854
39-116 VMSICID[0:15] - Message Signaled Interrupt Capability ID Register 3854
39-117 VMSINCP[0:15] - Message Signaled Interrupt Next Capability Pointer Register 3855
39-118 VMSICTL[0:15] - Message Signaled Interrupt Control Register 3855
39-119 VMSILADDR[0:15] - Message Signaled Interrupt Lower Address Register 3856
39-120 VMSIUADDR[0:15] - Message Signaled Interrupt Upper Address Register 3856
39-121 VMSIDATA[0:15] - Message Signaled Interrupt Data Register 3856
39-122 VMSIMSK[0:15] - VF Message Signaled Interrupt Mask Register 3857
39-123 VMSIPND[0:15] - VF Message Signaled Interrupt Pending Register 3857
39-124 VPCIEAERCAPID[0:15] - VF PCI Express AER Capability ID Register 3858
39-125 VPAERUCS[0:15] - VF PCI Express AER Uncorrectable Error Status Register 3858
39-126 VPAERUCM[0:15] - VF PCI Express AER Uncorrectable Error Mask Register 3859
39-127 VPAERUCSEV[0:15] - VF PCI Express AER Uncorrectable Error Severity Register 3861
39-128 VPAERCS[0:15] - VF PCI Express AER Correctable Error Status Register 3862
39-129 VPAERCM[0:15] - VF PCI Express AER Correctable Error Mask Register 3863
39-130 VPAERCTLCAP[0:15] - VF PCI Express AER Control and Capability Register 3864
39-131 VPAERHDRLOG0[0:15] - VF PCI Express AER Header Log 0 Register 3865
39-132 VPAERHDRLOG1[0:15] - VF PCI Express AER Header Log 1 Register 3865
39-133 VPAERHDRLOG2[0:15] - VF PCI Express AER Header Log 2 Register 3865
39-134 VPAERHDRLOG3[0:15] - VF PCI Express AER Header Log 3 Register 3866
39-135 VARIDHDR[0:15] - VF Alternative Routing ID Capability Header 3866
39-136 VFARICAP[0:15] - VF ARI Capabilities Register 3867
39-137 VARIDCTL[0:15] - VF Alternative Routing ID Control Register 3867
39-138 VACSCAPID[0:15] - VF ACS Capability ID Register 3868
39-139 VACSCVNC[0:15] - VF ACS Capability Version and Next Capability Pointer Register 3868
39-140 VACSCAP[0:15] - VF ACS Capabilities Register 3868
40-1 EPO PCH - Functions Not Supported 3872

Revision History

Document Number	Revision Number	Description	Date
336067	001	Initial release	July 2017
336067	002	Updated Section 24.7.10. Updated Section 27.8.5 bit 3 default from 0b to 1b. Updated Section 27.9.2. Added note 3 to 27.1 overview. Updated Chapter 27 to correctly state the frequencies supports (17, 30 and 40 MHz. Updated Chapter 18. Many of the defaults bit settings were incorrect for pins that defaulted to native functions. These were corrected. Updated Section 21.7 to call out no LPC DMA or bus mastering. Updated descriptions in "Surprise Powerdown to G3" in Chapter 4. Updated 9.10.7. Updated SEIO Glitch description in chapter 4 to include SEIOs with pullup resistors. Updated Chapter 20 to include description of the NMI and SMI signals on GPP_G18 and GPP_G19. Updated 10GbE DIDs in Chapter 2. Updated Chapter 19 and Chapter to 36 to reflect that BIOS can not disable the legacy 1 GbE LAN.	December 2017
336067	003	Updated Chapter 4 to include PCIe Gen4 jitter specs and notes. Corrected various sections where SLP_GbE# was still being referred to as SLP_LAN#. Updated DMI chapter to describe DMI packet sizes. Corrected PAD_CFG_DW0_GPD_6 default inconsistencies. Updated name of bits 0 and 1 in table of Section 9.10.7 (PM_CFG register). Updated sections 24.8.2.11 and 24.9.2.11 (changed defaults).	March 2018
336067	004	Replaced *Intel® QuickAssist Technology Programmer's Guide* with *Intel® QuickAssist Technology API Programmer's Guide* in Chapter 39. Renamed legacy PCHHOT# in the Thermal Management chapter with PHOT# to match pin name.	March 2018
336067	005	Replaced "Gen3" with "Gen4" in the signal descriptions in Chapter 8. Added C629 SKU information in Chapter 1. Updated SMI Status Register (SMI_STS, offset 34h) in Chapter 9. Added CDC_DUT_DETECT_N description in the power and ground signals chapter. Updated sections 4.4 and 4.6. Corrected MM# error in Table 1-3 for C624. Added C629 DID in Chapter 2. Added C629 to Table 4-4. Updated Hardstrap on GPP_B14, GPP_B18 and GPP_B22.	August 2018
336067	006	Updated Table 3-2. Updated Table 4-12: corrected GPP_G10_FANPWM2_FANPWM2IE name; added GPP_G14; removed duplicate entry of RSMRST_N, PCH_PWROK, DSW_PWROK, SRTCRST_N, RTCRST_N from the grouping of the DSW signals. Updated Chipset Initialization Register E0 (CIR_E0) in Section 9.10.45. Updated Chapter 16/Chapter 28 with native function on GPD0. Updated Proc_pwrgd description in Chapter 16. Updated Section 27.7.2 SPI Support for TPM. Chapter 38: updated Table 38-1, added electrical interface column, added NVM Recovery Mode information (.S security), added OCP Support, updated NVM Map to reflect Recovery Mode and OCP, updated Programming Information section to reflect Recovery Mode and OCP.	February 2019

Document Number	Revision Number	Description	Date
336067	007	Corrected Note 1 is SPI flash function description, Section 27.7.1.1. Made corrections to figures 17-2 and 17-3 (request and response for PCH temperature) in Chapter 17. Corrected ADR_Holdoff pin strap description in Chapter 3. Added P2SB Control register in Section 36.9. Updated HSIO PHY programming table in Section 36.10.4. Updated Section 38.20.22. Updated Chipset SKU Definition Table (Table 1-2). Updated diagram on diagrams for Power ramp with and without DSW to include SRTCRST# in Chapter 4. Updated measurement description T1 for timing tables on power ramp both with and without DSW.	May 2019
336067	008	In Chapter 2, PCH Controller Device IDs, updated the DID and virtual port name that the thermal sensor is attached to. In Chapter 22, corrected the PCIe X16 and x8 uplink Device IDs. Sections 22.8.1 and 22.9.1 Updated Chapter 38 in various sections. See change bars.	July 2019
336067	009	Chapter 29 (USB): Updated several sections. See change bars. Chapter 1: Updated PCH marking table. Chapter 38: Updated Table 38-546 (supported NC-SI Commands) footnote, Table 38-9 (Internal Switching Features), Table 38-56 (Set MAC Config Command Data Structure), Section 38.44.2.2 (Serial Number Registers). Chapter 17: Added note in Flash Access Channel Overview (Section 17.7.6.4) on SAFS support.	October 2019

§

1 Introduction

1.1 About this Manual

This document is intended for Original Equipment Manufacturers and BIOS vendors creating products based on the Intel® C620 Series Chipset Platform Controller Hub (PCH).

Note: Throughout this document the Platform Controller Hub is used as a general term and refers to all Intel® C620 Series Chipset PCH SKUs, unless specifically noted otherwise.

This manual assumes a working knowledge of the vocabulary and principles of interfaces and architectures such as PCI Express*, Universal Serial Bus, Advance Host Controller Interface, eXtensible Host Controller Interface, and so forth.

This manual abbreviates buses as B*n*, devices as D*n* and functions as F*n*. For example Device 31 Function 0 is abbreviated as D31:F0, Bus 1 Device 8 Function 0 is abbreviated as B1:D8:F0. Generally, the bus number will not be used, and can be considered to be Bus 0.

1.2 Overview

The Intel® C620 Series Chipset PCH provides extensive I/O support. Functions and capabilities include:

- ACPI Power Management Logic Support, Revision 4.0a
- *PCI Express* Base Specification*, Revision 3.0
- Integrated Serial ATA host controller, supports data transfer rates of up to 6 Gb/s on all ports.
- xHCI USB controller with SuperSpeed USB 3.0 ports
- Direct Media Interface
- Serial Peripheral Interface
- Enhanced Serial Peripheral Interface
- Flexible I/O—Allows some high speed I/O signals to be configured as PCIe* root ports, PCIe uplink for use with certain PCH SKUs, SATA (and sSATA), or USB 3.0.
- General Purpose Input Output (GPIO)
- Low Pin Count interface, interrupt controller, and timer functions
- *System Management Bus Specification*, Version 2.0
- Integrated Clock Controller/Real Time Clock Controller
- Intel® High Definition Audio (Intel® HD Audio) and Intel® Smart Sound Technology (Intel® SST)
- Integrated 10/1 Gb Ethernet
- Integrated 10/100/1000 Mbps Ethernet MAC
- Supports Intel® Rapid Storage Technology enterprise (Intel® RSTe)
- Supports Intel® Active Management Technology and Intel® Server Platform Services
- Supports Intel® Virtualization Technology (Intel® VT) for Directed I/O (Intel® VT-d)
- Supports Intel® Trusted Execution Technology (Intel® TXT)

- JTAG Boundary Scan support
- Intel® QuickAssist Technology (Intel® QAT)
- Intel® Trace Hub (Intel® TH) for debug
- Innovation Engine
- ADR Support

Note: Not all functions and capabilities may be available on all SKUs.

1.3 PCH I/O Features and Capabilities

The SKU information will be published in an upcoming revision.

The following table provides an overview of the PCHI/O capabilities.

1.4 PCH Software and Firmware

Table 1-1. PCH I/O Capabilities

Interface	Intel® C620 Series Chipset
CPU Interface	DMI 3.0 x4
PCIe	Up to 20 3.0 lanes (16 devices max) for root ports, and up to 24 3.0 lanes used as downstream ports
USB	Up to 10 superspeed, 14 highspeed
SATA	Up to 14 3.0
LAN Ports	One GbE, Four 10 GbE
Audio	HD-Audio, I2S, Direct attach Digital Microphone
LPC	24 MHz, No DMA
eSPI	Two CS#, Quad Mode
SMBus/SMLink	One host SMBus, and 6 SMLinks
SPI	Three devices, including TPM

Many of the features listed are dependent upon specific Firmware (FW) or Software (SW) to enable and operate them. Others can use industry standard drivers.

Intel RSTe is the driver available for use with the PCH. For information on the features available with Intel RSTe, check with your Intel Rep or SW PAE for full details.

There are two FW stacks provided by Intel that are used with Intel® C620 Series Chipset – Intel AMT FW (Intel® Management Engine [Intel® ME] 11) and Server Platform Software (SPS). These enable and use different features, such as the integrated 1 GbE MAC and the Innovation Engine. The Innovation Engine has its own FW stack that is developed by the customer for customer specific functionality.

Drivers and NVM setup for the 10 GbE are provided by Intel.

1.5 PCH SKUs

The chart below lists the differences between the various SKUs of the PCH. These SKU definitions are subject to change.

Table 1-2. Intel® C620 Series Chipset SKU Differentiation

SKU	C621	C622	C624	C625	C626	C627	C628	C629
Feature								
Legacy 10/100/1000 Mbps Ethernet	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
LAN Port 0	1G	10/1G	10/1G	10/1G	10/1G	10/1G	10/1G	10/1G
LAN Port 1	1G	10/1G	10/1G	10/1G	10/1G	10/1G	10/1G	10/1G
LAN Port 2	1G	1G	10/1G	10/1G	10/1G	10/1G	10/1G	10/1G
LAN Port 3	1G	1G	10/1G	10/1G	10/1G	10/1G	10/1G	10/1G
Dedicated PCIe Uplink	x1	x8	x16	x16	x16	X16	X16	X16
Muxed x8 PCIe Uplink	Disabled	Disabled	Disabled	Disabled	Enabled	Enabled	Enabled	Enabled
Intel QAT Engines	No	No	No	1	2	3	3	3
Intel QAT Clock Speed	N/A	N/A	N/A	533 MHz	533 MHz	685 MHz	685 MHz	800 MHz
AVID Enabled	No	No	No	Yes (0.85-1.0V)	Yes (0.85-1.0V)	Yes (0.85-1.0V)	Yes (0.85-1.0V)	Yes (0.85-1.0V)

1.6 PCH Markings

Table 1-3. PCH Markings (Sheet 1 of 2)

SKU	Phase	Stepping	S-Spec	QDF	MM#	ROHS Compliant	Notes
Intel® C621 Series Chipset (Intel® EY82C621 PCH)	Production	B1	R36S	N/A	954858	YES	
Intel® C621 Series Chipset (Intel® EY82C621 PCH)	Production	S0	R354	N/A	953999	YES	
Intel® C622 Series Chipset (Intel® EY82C622 PCH)	Production	S0	R36X	N/A	954863	YES	
Intel® C624 Series Chipset (Intel® EY82C624 PCH)	Production	S0	R36Y	N/A	954864	YES	
Intel® C625 Series Chipset (Intel® EY82C625 PCH)	Production	B1	R36W	N/A	954862	YES	
Intel® C626 Series Chipset (Intel® EY82C626 PCH)	Production	B1	R36V	N/A	954861	YES	
Intel® C627 Series Chipset (Intel® EY82C627 PCH)	Production	B1	R36U	N/A	954860	YES	
Intel® C628 Series Chipset (Intel® EY82C628 PCH)	Production	B1	R36T	N/A	954859	YES	
Intel® C621 Series Chipset (Intel® EY82C621 PCH)	Production	B2	R3HE	N/A	957625	YES	
Intel® C621 Series Chipset (Intel® EY82C621 PCH)	Production	S1	R3HL	N/A	957631	YES	

Table 1-3. PCH Markings (Sheet 2 of 2)

SKU	Phase	Stepping	S-Spec	QDF	MM#	ROHS Compliant	Notes
Intel® C622 Series Chipset (Intel® EY82C622 PCH)	Production	S1	R3HK	N/A	957630	YES	
Intel® C624 Series Chipset (Intel® EY82C624 PCH)	Production	S1	R3HM	N/A	957632	YES	
Intel® C625 Series Chipset (Intel® EY82C625 PCH)	Production	B2	R3HJ	N/A	957629	YES	
Intel® C626 Series Chipset (Intel® EY82C626 PCH)	Production	B2	R3HH	N/A	957628	YES	
Intel® C627 Series Chipset (Intel® EY82C627 PCH)	Production	B2	R3HG	N/A	957672	YES	
Intel® C628 Series Chipset (Intel® EY82C628 PCH)	Production	B2	R3HF	N/A	957626	YES	
Intel C629 Series Chipset (Intel® EY82C629 PCH)	Production	B2	RCWR	N/A	976392	Yes	

§

2 PCH Controller Device IDs

2.1 Device and Revision ID Table

The Device IDs (DID) listed below are the numbers that can be found in the PCI standard header registers at offset 02H for each the functions listed below. The Device IDs are unique to Intel® C620 Series Chipset with the exception of the Alternate IDs. They are common with previous generations so external SW can identify this as an Intel component. The LPC DID is used to identify the SKUs. There are two sets of DIDs for most components, one for super SKU parts and the other for production stepping steppings. The 10 GbE and Intel® QuickAssist Technology (Intel® QAT) engines only have one DID.

§

Table 2-1. Intel® C620 Series ChipsetDevice and Revision ID Table (Sheet 1 of 4)

Intel® C620 Series Chipset Production SKUs	Intel® C620 Series Chipset Super SKUs	Device Function - Device Description	A0 SRID	B0 SRID	B1 SRID	B2 SRID	S0 SRID	S1 SRID	Notes
A182	A202	D23:F0 - SATA Controller (AHCI)	0	2	3	4	8	9	AHCI
A186	A206	D23:F0 - SATA Controller (RAID 0/1/5/10)	0	2	3	4	8	9	Third Party RAID ID
2822	2822	D23:F0 - SATA Controller (RAID 0/1/5/10) Alternate ID	0	2	3	4	8	9	Alternate ID
2826	2826	D23:F0 - SATA Controller (RAID 0/1/5/10) Alternate ID	0	2	3	4	8	9	Alternate ID for Intel RSTe
A190	A210	D28:F0 - PCI Express* Root Port #0	0	2	3	4	8	9	
A191	A211	D28:F1 - PCI Express Root Port #1	0	2	3	4	8	9	
A192	A212	D28:F2 - PCI Express Root Port #2	0	2	3	4	8	9	
A193	A213	D28:F3 - PCI Express Root Port #3	0	2	3	4	8	9	
A194	A214	D28:F4 - PCI Express Root Port #4	0	2	3	4	8	9	
A195	A215	D28:F5 - PCI Express Root Port #5	0	2	3	4	8	9	
A196	A216	D28:F6 - PCI Express Root Port #6	0	2	3	4	8	9	
A197	A217	D28:F7 - PCI Express Root Port #7	0	2	3	4	8	9	
A198	A218	D29:F0 - PCI Express Root Port #8	0	2	3	4	8	9	
A199	A219	D29:F1 - PCI Express Root Port #9	0	2	3	4	8	9	
A19A	A21A	D29:F2 - PCI Express Root Port #10	0	2	3	4	8	9	
A19B	A21B	D29:F3 - PCI Express Root Port #11	0	2	3	4	8	9	

Table 2-1. Intel® C620 Series ChipsetDevice and Revision ID Table (Sheet 2 of 4)

Intel® C620 Series Chipset Production SKUs	Intel® C620 Series Chipset Super SKUs	Device Function - Device Description	A0 SRID	B0 SRID	B1 SRID	B2 SRID	S0 SRID	S1 SRID	Notes
A19C	A21C	D29:F4- PCI Express Root Port #12	0	2	3	4	8	9	
A19D	A21D	D29:F5- PCI Express Root Port #13	0	2	3	4	8	9	
A19E	A21E	D29:F6- PCI Express Root Port #14	0	2	3	4	8	9	
A19F	A21F	D29:F7 - PCI Express Root Port #15	0	2	3	4	8	9	
A1A0	A220	D31:F1 – P2SB	0	2	3	4	8	9	
A1A1	A221	D31:F2 - Power Management Controller	0	2	3	4	8	9	
A1A3	A223	D31:F4 - SMBus	0	2	3	4	8	9	
A1A4	A224	D31:F5 – SPI Controller	0	2	3	4	8	9	
15B9		D31:F6 – GbE Controller	0	2	3	4	8	9	GbE DID will be updated with NVM Load
A1A6	A226	D31:F7 - Intel® Trace Hub	0	2	3	4	8	9	
A1AF	A22F	D20:F0 – USB 3.0 xHCI Controller	0	2	3	4	8	9	
A1B1	A231	D20:F2 – Thermal Subsystem	0	2	3	4	8	9	
A1BA	A23A	D22:F0 - Intel ME HECI #1	0	2	3	4	8	9	
A1BB	A23B	D22:F1 – Intel ME HECI #2	0	2	3	4	8	9	
A1BC	A23C	D22:F2 – Intel ME: IDE Redirection	0	2	3	4	8	9	
A1BD	A23D	D22:F3 – Intel ME: Keyboard and Text (KT) Redirection	0	2	3	4	8	9	
A1BE	A23E	D22:F4 - Intel ME HECI #3	0	2	3	4	8	9	
	A242	D31:F0 - LPC or eSPI Controller	0	2	3	N/A	N/A	N/A	Intel® C624 Series Chipset Supersku
	A243	D31:F0 - LPC or eSPI Controller	0	2	3	N/A	N/A	N/A	Intel® C627 Series Chipset Supersku
	A244	D31:F0 - LPC or eSPI Controller	0	2	3	N/A	N/A	N/A	Intel® C621 Series Chipset Supersku
	A245	D31:F0 - LPC or eSPI Controller	0	2	3	N/A	N/A	N/A	Intel® C627 Series Chipset Supersku
	A246	D31:F0 - LPC or eSPI Controller	0	2	3	N/A	N/A	N/A	Intel® C628 Series Chipset Supersku
A1C1		D31:F0 - LPC or eSPI Controller	0	2	3	4	8	9	Intel® C621 Series Chipset (QS/PRQ)
A1C2		D31:F0 - LPC or eSPI Controller	0	2	3	N/A	8	9	Intel® C622 Series Chipset (QS/PRQ)
A1C3		D31:F0 - LPC or eSPI Controller	0	2	3	N/A	8	9	Intel® C624 Series Chipset (QS/PRQ)
A1C4		D31:F0 - LPC or eSPI Controller	0	2	3	4	N/A	N/A	Intel® C625 Series Chipset (QS/PRQ)
A1C5		D31:F0 - LPC or eSPI Controller	0	2	3	4	N/A	N/A	Intel® C626 Series Chipset (QS/PRQ)

Table 2-1. Intel® C620 Series ChipsetDevice and Revision ID Table (Sheet 3 of 4)

Intel® C620 Series Chipset Production SKUs	Intel® C620 Series Chipset Super SKUs	Device Function - Device Description	A0 SRID	B0 SRID	B1 SRID	B2 SRID	S0 SRID	S1 SRID	Notes
A1C6		D31:F0 - LPC or eSPI Controller	0	2	3	4	N/A	N/A	Intel® C627 Series Chipset (QS/PRQ)
A1C7		D31:F0 - LPC or eSPI Controller	0	2	3	4	N/A	N/A	Intel® C628 Series Chipset (QS/PRQ)
A1CA	N/A	D31:F0 - LPC or eSPI Controller	N/A	N/A	N/A	4	N/A	N/A	Intel® C629 Series Chipset(QS/PRQ)
A1D2	A252	D17:F5 - SSATA Controller (AHCI)	0	2	3	4	8	9	AHCI Mode
A1D6	A256	D17:F5 - SSATA Controller (RAID 0/1/5/10)	0	2	3	4	8	9	Third Party RAID ID
2823	2823	D17:F5 - SSATA Controller (RAID 0/1/5/10)	0	2	3	4	8	9	Alternate ID
2827	2827	D17:F5 - SSATA Controller (RAID 0/1/5/10)	0	2	3	4	8	9	Alternate ID for Intel RSTe
A1E7	A267	D27:F0 - PCI Express Root Port #16	0	2	3	4	8	9	
A1E8	A268	D27:F1 - PCI Express Root Port #17	0	2	3	4	8	9	
A1E9	A269	D27:F2 - PCI Express Root Port #18	0	2	3	4	8	9	
A1EA	A26A	D27:F3 - PCI Express Root Port #19	0	2	3	4	8	9	
A1EC	A26C	MROM 0	0	2	3	4	8	9	
A1ED	A26D	MROM 1	0	2	3	4	8	9	
A1F0	A270	D31:F3 Audio	0	2	3	4	8	9	
A1F8	A278	D16:F0 - IE:HECI #1	0	2	3	4	8	9	
A1F9	A279	D16:F1 - IE: HECI #2	0	2	3	4	8	9	
A1FA	A27A	D16:F2 - IE:IDE Redirection	0	2	3	4	8	9	
A1FB	A27B	D16:F3 - IE: Keyboard and Text Redirection	0	2	3	4	8	9	
A1FC	A27C	D16:F4 - IE: HECI #3	0	2	3	4	8	9	
37C0	37C0	PCIe x16 Uplink (NPX16)	0	2	3	4	8	9	
37C1	37C1	PCIe x8 Uplink (NPX8)	0	2	3	4	8	9	
37C2	37C2	Virtual Switch Port 0	0	2	3	4	8	9	For Intel QAT 0
37C3	37C3	Virtual Switch Port 1	0	2	3	4	8	9	For Intel QAT 1
37C4	37C4	Virtual Switch Port 2	0	2	3	4	8	9	For Intel QAT 2
37C5	37C5	Virtual Switch Port 3	0	2	3	4	8	9	For 10 GbE LAN
37C7	37C7	Virtual Switch Port 5	0	2	3	4	8	9	For Thermal Sensor in Endpoint (EP) mode
37C8	37C8	PF0 for Intel QuickAssist Technology 0-2	0	2	3	4	8	9	
37C9	37C9	VF for Intel QuickAssist Technology 0-2	0	2	3	4	8	9	
37B1	37B1	Thermal Sensor	0	2	3	4	8	9	For Thermal Sensor in EP mode

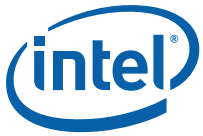


Table 2-1. Intel® C620 Series ChipsetDevice and Revision ID Table (Sheet 4 of 4)

Intel® C620 Series Chipset Production SKUs	Intel® C620 Series Chipset Super SKUs	Device Function - Device Description	A0 SRID	B0 SRID	B1 SRID	B2 SRID	S0 SRID	S1 SRID	Notes
10A6, 37CC, 37CE-37D3		10 Gb Ethernet	0	2	3	4	8	9	10A6 is for disable function or before NMV loaded. 37CC is with NVM loaded but PHY device not detected. 37CE-37D3 depends on NVM image loaded. See LEK for which PHY corresponds to which DID.

3 Pin Straps

The following signals are used for static configuration. They are sampled at the rising edge of RSMRST# or PCH_PWROK to select configuration and then revert later to their normal usage. To invoke the associated mode, the signal should be driven at least four PCI clocks prior to the time it is sampled.

The straps that are latched with RSMRST# also have a secondary qualifier. In order to the straps to be latched at this time, SLP_SUS# must be high. If DeepSX is supported then this is not an issue. If DeepSX is not supported, RSMRST# will need to wait until SLP_SUS# is high.

Table 3-1. Functional Strap Definitions (Sheet 1 of 3)

Signal	Usage	When Sampled	Comment
GPP_B14_SPKR	Top Swap Override	Rising edge of PCH_PWROK	The signal has a weak internal pull-down. 0 = **Disable** "Top Swap" mode. (Default) 1 = **Enable** "Top Swap" mode. This inverts an address bit (defined in FITC) at the top of system BIOS space. This provides an alternate physical address that the processor accesses at boot. ***Notes:*** 1. The internal pull-down is disabled after PCH_PWROK# de-asserts. 2. Software will not be able to clear the Top Swap bit until the system is rebooted. 3. The status of this strap is readable using the Top Swap bit (Bus0, Device31, Function0, offset DCh, bit4).
GPP_B18	No Reboot	Rising edge of PCH_PWROK	The signal has a weak internal pull-down. 0 = **Disable** "No Reboot" mode. 1 = **Enable** "No Reboot" mode (PCH will disable the TCO Timer system reboot feature). This function is useful when running Intel® In-Target Probe (Intel® ITP)/ XDP. ***Notes:*** 1. The internal pull-down is disabled after PCH_PWROK# de-asserts. 2. The status of this strap is readable using the NO REBOOT bit (Chipset Configuration Registers: RCBA + Offset 3410h:Bit 5).
GPP_C2_SM-BALERT#	TLS Confi-dentiality	Rising edge of RSMRST#	This signal has a weak internal pull-down. 0 = **Disable** Intel ME Crypto Transport Layer Security (TLS) cipher suite (no confidentiality). 1 = **Enable** Intel ME Crypto Transport Layer Security (TLS) cipher suite (with confidentiality). Must be pulled up to support Intel AMT with TLS and Intel® Small Business Advantage (Intel® SBA) with TLS. ***Notes:*** 1. The internal pull-down is disabled after RSMRST# de-asserts.

Table 3-1. Functional Strap Definitions (Sheet 2 of 3)

Signal	Usage	When Sampled	Comment
GPP_B22	Boot BIOS Strap Bit BBS	Rising edge of PCH_PWROK	This Signal has a weak internal pull-down. This field determines the destination of accesses to the BIOS memory range. Also controllable using Boot BIOS Destination bit (Chipset Configuration Registers: Offset 3410h:Bit 10). This strap is used in conjunction with Boot BIOS Destination Selection 0 strap. **Bit 10** — **Boot BIOS Destination** 0 — SPI 1 — LPC/eSPI ***Notes:*** 1. The internal pull-down is disabled after PCH_PWROK# de-asserts. 2. If option 1 (LPC/eSPI) is selected, BIOS may still be placed on LPC/eSPI but all platforms are required to have SPI flash connected directly to the PCH's SPI bus with a valid descriptor in order to boot. 3. Boot BIOS Destination Select to LPC by functional strap or using Boot BIOS Destination Bit will not affect SPI accesses initiated by Intel ME or LAN.
GPP_C5_SML0ALERT_IE#	eSPI_EN	Rising edge of RSMRST#	This signal has a weak internal pull-down. 0 = **LPC** functionality is selected. 1 = **eSPI** functionality is selected. ***Notes:*** 1. The internal pull-down is disabled after RSMRST# de-asserts.
HDA_SDO	Flash Descriptor Security Override	Rising edge of PCH_PWROK	This signal has a weak internal pull-down. 0 = **Enable** security measures defined in the Flash Descriptor. 1 = **Disable** Flash Descriptor Security (override). This strap should only be asserted high using external pull-up in manufacturing/debug environments ONLY. ***Notes:*** 1. The internal pull-down is disabled after PLTRST# de-asserts. 2. Asserting HDA_SDO high on the rising edge of PWROK will also halt Intel Management Engine after Chipset bring up and disable runtime Intel ME features. This is a debug mode and must not be asserted after manufacturing/debug.
GPP_H12_SML2ALART_IE# /	Master or Slave Attached Flash	Rising edge of RSMRST#	This signal has a weak internal pull-down. 0 = Master attached Flash. 1 = Slave Attached Flash. ***Notes:*** 1. The internal pull-down is disabled after RSMRST# de-asserts.
GPP_H15	ADR Timer Hold Off	RSMRST	This signal has a weak internal pull-up. 0 = ADR Timer Hold off is enabled. 1 = ADR Timer Hold off is not enabled. When sampled LOW, the ADR timer will not expire which prevents ADR_COMPLETE from asserting and keeps the Global Reset from occurring.
SPI_MISO_IO1	SPI Buffer voltage	rising edge of RSMRST#	If sampled low, then SPI buffer is set to 1.8V. If high, then the buffer is 3.3V.
GPP_H18_SML4ALERT_N_IE_N	RSVD	rising edge of RSMRST#	This pin has an internal pull-up. This strap needs to be sampled high during the rising edge of RSMRST#.
GPP_K7	RSVD	rising edge of RSMRST#	This pin has an internal pulldown. This strap needs to be sampled low during the rising edge of RSMRST#.
SPIO_MOSI_IO0	RSVD	rising edge of RSMRST#	This pin has an internal pull-up. This strap needs to be sampled high during the rising edge of RSMRST#.

Table 3-1. Functional Strap Definitions (Sheet 3 of 3)

Signal	Usage	When Sampled	Comment
GPP_B23_MEI-E_SML1AL-RT_N_PHOT_N	DCI Disable	rising edge of RSMRST#	This pin has an internal pulldown. This strap needs to be sampled low during the rising edge of RSMRST# in order to allow DCI to be enabled or disabled via soft straps or BIOS. If this strap is sampled high then DCI is disabled.
SPIO_IO_2	RSVD	rising edge of RSMRST#	This pin has an internal pull-up. This strap needs to be sampled high during the rising edge of RSMRST#.
SPIO_IO_3	RSVD	rising edge of RSMRST#	This pin has an internal pull-up. This strap needs to be sampled high during the rising edge of RSMRST#.

There are now 3 bits that encode where the Flash devices are located and where the BIOS is located. The table below can be used to understand how those encodings work.

Table 3-2. SPI Flash and BIOS Location

GPP_C5_SML0ALERT_IE# (eSPI or LPC Selection)	GPP_B22 (BIOS Location: SPI or eSPI/LPC)	GPP_H12_SML2ALART_IE# / (Master or Slave Attached Flash)	Configuration
0	**0**	**0**	BIOS and all SPI contents on SPI flash attached to PCH.
0	**0**	**1**	Invalid configuration ... not supported
0	**1**	**0**	BIOS is on LPC, the rest of the SPI contents are in a SPI Flash device attached to the PCH SPI bus.
0	**1**	**1**	Invalid configuration ... not supported
1	**0**	**0**	BIOS and all SPI contents on SPI flash attached to PCH.
1	**0**	**1**	BIOS and all SPI contents are on SPI flash attached to eSPI (same a 1,1,1).
1	**1**	**0**	BIOS is on eSPI, the rest of the SPI contents are in a SPI Flash device attached to the PCH SPI bus.
1	**1**	**1**	All content (BIOS, Intel ME, LAN, etc.) are on the eSPI bus, behind a BMC.

§

4 Electrical Characteristics

4.1 PCH Power Supply Range

Table 4-1. Voltage Ranges

Parameter	Nom	DC		AC		Total		Unit	Notes
		Min	Max	Min	Max	Min	Max		
VCCPGPPA, VCCPGPPB, VCCPGPPC, VCCPGPPD, VCCPGPPE, VCCPGPPF, VCCPGPPG, VCCPGPPH, VCCPGPPJ, VCCPGPPK; VCCPUSBDWS_3P3, VDDPDSW_3P3, VCCSPI VCCP_3P3, VCCPRTCPRIM_3P3	3.3	0.033	0.033	0.033	0.033	3.135	3.465	V	DC=± 1% AC= ±1% total = ±5%
VCCPGPPA, VCCPGPPB, VCCPGPPC, VCCPGPPD, VCCPGPPE, VCCPGPPF, VCCPGPPG, VCCPGPPH, VCCPGPPJ, VCCPGPPK; VCCP_1P8, VCCP10GBE_HV_1P8, VCCPGPP_1P8, VDDPHDA_1P8, VPSPI_1P8	1.8	0.018	0.018	0.018	0.018	1.71	1.89	V	DC=± 1% AC= ± 1% total = ± 5%
VCCA_CLKFILT_1P05, VCCA_CLKPLL_1P05, VCCA_CLKUNF_1P05, VCCA_CLKXTAL_1P05, VCCMPHY_1P05, VCCP_HSIOPLL_1P05, VCCP_UPPLL_1P05, VCCP10GBE_LV_1P05, VCCPRIM_1P05, VCCA_PLL0_1P05, VCCA_PLL1_1P05, VCCP_HSIOPLLUNF_1P05, VCCP_UPPLLUNF_1P05, VCCP10GBEPLL_1P05	1.05	0.00525	0.00525	0.00315	0.00315	1.0185	1.0815	V	DC=± 0.5% AC= ± 0.3% total = ± 3%
VCCPRIM_AVID	1.00	0.01	0.01	0.01	0.01	0.955	1.045	V	DC=± 1% AC= ± 1% total = ±4.5%
VCCPRIM_AVID	0.95	0.0095	0.0095	0.0095	0.0095	0.9072	0.99275	V	DC=± 1% AC= ± 1% total = ±4.5%
VCCPRIM_AVID	0.90	0.009	0.009	0.009	0.009	0.8595	0.9405	V	DC=± 1% AC= ± 1% total = ±4.5%
VCCPRIM_AVID	0.85	0.0085	0.0085	0.0085	0.0085	0.81175	0.88825	V	DC=± 1% AC= ± 1% total = ±4.5%
Notes: DC min/max is DC voltage offset from nominal. AC min/max is AC offset component to voltage. Total min/max is absolute maximum and minimum the voltage can be.									

Table 4-2. ICC Sx

Configuration 25C, Typical parts	Vcc_Avid	1.05V	1.8V	3.3V	3.3V_DSW
10GbE WOL enabled	1300 mA	400 mA	300 mA	10-200 mA	50 mA
10GbE WOL disabled	350 mA	60 mA	50 mA	10-200 mA	50 mA
Deep Sx	0 mA	0 mA	0 mA	0 mA	1 mA
G3	0 mA	0 mA	0 mA	0 mA	0 mA
Notes: Sx power is for a typical part at 25C. 3.3V variance is due to GPIO configuration.					

Table 4-3. Absolute Maximum Voltage Levels for Power Planes

Voltage Levels	Min Limit	Maximum Limit
Vccprim_Avid (0.85-1.0V) 1.05V	-0.5V	1.4V
1.8V	-0.5V	2.0V
3.3V	-0.7V	3.7V
Note: Max. overshoot and undershoot of I/O signals are listed in Section 4.5.		

4.2 General DC Characteristics

Table 4-4. Intel® C620 Series Chipset PCH Estimated I_{cc} (Maximum Configuration C627 and C629) (Sheet 1 of 2)

Voltage Rail	Voltage (V)	Published Performance Workload x16 Uplink Only		High Performance Workload, x24 Uplink Enabled	
		S0 Iccmax Current[3] (A)	S0 IccTDC (A)	S0 Iccmax Current[3] (A)	S0 IccTDC (A)
VCCDSW_3P3	3.3	0.5	0.5	0.5	0.5
VCCP_1P8	1.8	1.0	0.6	1.0	0.6
VCCP_3P3	3.3	0.6	0.6	0.6	0.6
VCCPRIM_AVID	0.85-1.0	23	19.5	23	20.5
VCCPRIM_1P05	1.05	15	12	15	12
VCCPGPPA	3.3	0.1	0.1	0.1	0.1
	1.8	0.1	0.1	0.1	0.1
VCCPGPPB	3.3	0.025	0.025	0.025	0.025
	1.8	0.025	0.025	0.025	0.025
VCCPGPPC	3.3	0.1	0.1	0.1	0.1
	1.8	0.1	0.1	0.1	0.1
VCCPGPPD	3.3	0.025	0.025	0.025	0.025
	1.8	0.025	0.025	0.025	0.025
VCCPGPPE	3.3	0.025	0.025	0.025	0.025
	1.8	0.025	0.025	0.025	0.025
VCCPGPPF	3.3	0.025	0.025	0.025	0.025
	1.8	0.025	0.025	0.025	0.025

Table 4-4. Intel® C620 Series Chipset PCH Estimated I_{cc} (Maximum Configuration C627 and C629) (Sheet 2 of 2)

VCCPGPPG	3.3	0.025	0.025	0.025	0.025
	1.8	0.025	0.025	0.025	0.025
VCCPGPPH	3.3	0.05	0.05	0.05	0.05
	1.8	0.05	0.05	0.05	0.05
VCCPGPPJ	3.3	0.05	0.05	0.05	0.05
	1.8	0.05	0.05	0.05	0.05
VCCPGPPK	3.3	0.025	0.025	0.025	0.025
	1.8	0.025	0.025	0.025	0.025

Note: Vccprim_1P05 is the current of all 1.05 voltages combined. Vccp_3P3 is the current for all 3.3 voltages (minus GPIO power pins) combined. Vccp_1P8 is the current of all 1.8 voltages combined (minus GPIO power pins).

Note: The *Intel*® C620 Series Chipset *PDG* has VCCP_3P3 as 0.6A and not 0.3A as it rolls all the other GPIOs up to the 3.3V rail, minus VCCPGPPA.

Table 4-5. Intel® C620 Series Chipset PCH Estimated I_{cc} (C621) (Sheet 1 of 2)

Voltage Rail	Voltage (V)	S0 Iccmax Current[3] (A)	S0 IccTDC
VCCDSW_3P3	3.3V	0.5	0.5
VCCP_1P8	1.8V	1	0.6
VCCPP_3P3	3.3V	0.6	0.6
VCCPRIM_AVID	1.0V	10.5	9
VCCPRIM_1P05	1.05V	10	8.5
VCCPGPPA	3.3	0.1	0.1
	1.8	0.1	0.1
VCCPGPPB	3.3	0.025	0.025
	1.8	0.025	0.025
VCCPGPPC	3.3	0.1	0.1
	1.8	0.1	0.1
VCCPGPPD	3.3	0.025	0.025
	1.8	0.025	0.025
VCCPGPPE	3.3	0.025	0.025
	1.8	0.025	0.025
VCCPGPPF	3.3	0.025	0.025
	1.8	0.025	0.025
VCCPGPPG	3.3	0.025	0.025
	1.8	0.025	0.025
VCCPGPPH	3.3	0.05	0.05
	1.8	0.05	0.05
VCCPGPPJ	3.3	0.05	0.05
	1.8	0.05	0.05
VCCPGPPK	3.3	0.025	0.025
	1.8	0.025	0.025

Table 4-5. Intel® C620 Series Chipset PCH Estimated I_{CC} (C621) (Sheet 2 of 2)

Voltage Rail	Voltage (V)	S0 Iccmax Current[3] (A)	S0 IccTDC
Notes: Vccprim_1P05 is the current of all 1.05 voltages combined. Vccp_3P3 is the current for all 3.3 Voltages (minus GPIO power pins) combined. Vccp_1P8 is the current of all 1.8 voltages combined (minus GPIO power pins). The *Intel*® C620 Series Chipset *PDG* has VCCP_3P3 as 0.6A and not 0.3A as it rolls all the other GPIOs up to the 3.3V rail, minus VCCPGPPA.			

Table 4-6. Intel® C620 Series Chipset PCH Estimated I_{CC} (C622)

Voltage Rail	Voltage (V)	S0 Iccmax Current[3] (A)	S0 IccTDC
VCCDSW_3P3	3.3	0.5	0.5
VCCP_1P8	1.8	1.0	0.6
VCCP_3P3	3.3	0.6	0.6
VCCPRIM_AVID	1.0	10.5	9.5
VCCPRIM_1P05	1.05	12.0	9.5
VCCPGPPA	3.3	0.1	0.1
	1.8	0.1	0.1
VCCPGPPB	3.3	0.025	0.025
	1.8	0.025	0.025
VCCPGPPC	3.3	0.1	0.1
	1.8	0.1	0.1
VCCPGPPD	3.3	0.025	0.025
	1.8	0.025	0.025
VCCPGPPE	3.3	0.025	0.025
	1.8	0.025	0.025
VCCPGPPF	3.3	0.025	0.025
	1.8	0.025	0.025
VCCPGPPG	3.3	0.025	0.025
	1.8	0.025	0.025
VCCPGPPH	3.3	0.05	0.05
	1.8	0.05	0.05
VCCPGPPJ	3.3	0.05	0.05
	1.8	0.05	0.05
VCCPGPPK	3.3	0.025	0.025
	1.8	0.025	0.025
Notes: Vccprim_1P05 is the current of all 1.05 voltages combined. Vccp_3P3 is the current for all 3.3 Voltages (minus GPIO power pins) combined. Vccp_1P8 is the current of all 1.8 voltages combined (minus GPIO power pins). The *Intel*® C620 Series Chipset *PDG* has VCCP_3P3 as 0.6A and not 0.3A as it rolls all the other GPIOs up to the 3.3V rail, minus VCCPGPPA.			

Table 4-7. Intel® C620 Series Chipset PCH Estimated I_{cc} (C624)

Voltage Rail	Voltage (V)	S0 Iccmax Current[3] (A)	S0 IccTDC
VCCDSW_3P3	3.3	0.5	0.5
VCCP_1P8	1.8	1	0.6
VCCP_3P3	3.3	0.6	0.6
VCCPRIM_AVID	1.0	10.5	9.5
VCCPRIM_1P05	1.05	14	11.0
VCCPGPPA	3.3	0.1	0.1
	1.8	0.1	0.1
VCCPGPPB	3.3	0.025	0.025
	1.8	0.025	0.025
VCCPGPPC	3.3	0.1	0.1
	1.8	0.1	0.1
VCCPGPPD	3.3	0.025	0.025
	1.8	0.025	0.025
VCCPGPPE	3.3	0.025	0.025
	1.8	0.025	0.025
VCCPGPPF	3.3	0.025	0.025
	1.8	0.025	0.025
VCCPGPPG	3.3	0.025	0.025
	1.8	0.025	0.025
VCCPGPPH	3.3	0.05	0.05
	1.8	0.05	0.05
VCCPGPPJ	3.3	0.05	0.05
	1.8	0.05	0.05
VCCPGPPK	3.3	0.025	0.025
	1.8	0.025	0.025

Notes: Vccprim_1P05 is the current of all 1.05 voltages combined. Vccp_3P3 is the current for all 3.3 Voltages (minus GPIO power pins) combined. Vccp_1P8 is the current of all 1.8 voltages combined (minus GPIO power pins).
The *Intel*® C620 Series Chipset *PDG* has VCCP_3P3 as 0.6A and not 0.3A as it rolls all the other GPIOs up to the 3.3V rail, minus VCCPGPPA.

Table 4-8. Intel® C620 Series Chipset PCH Estimated I_{cc} (C625) (Sheet 1 of 2)

Voltage Rail	Voltage (V)	S0 Iccmax Current[3] (A)	S0 IccTDC
VCCDSW_3P3	3.3	0.5	0.5
VCCP_1P8	1.8	1	0.6
VCCP_3P3	3.3	0.6	0.6
VCCPRIM_AVID	0.85-1.0	12	11
VCCPRIM_1P05	1.05	14	11
VCCPGPPA	3.3	0.1	0.1
	1.8	0.1	0.1

Table 4-8. Intel® C620 Series Chipset PCH Estimated I_{CC} (C625) (Sheet 2 of 2)

Voltage Rail	Voltage (V)	S0 Iccmax Current[3] (A)	S0 IccTDC
VCCPGPPB	3.3	0.025	0.025
	1.8	0.025	0.025
VCCPGPPC	3.3	0.1	0.1
	1.8	0.1	0.1
VCCPGPPD	3.3	0.025	0.025
	1.8	0.025	0.025
VCCPGPPE	3.3	0.025	0.025
	1.8	0.025	0.025
VCCPGPPF	3.3	0.025	0.025
	1.8	0.025	0.025
VCCPGPPG	3.3	0.025	0.025
	1.8	0.025	0.025
VCCPGPPH	3.3	0.05	0.05
	1.8	0.05	0.05
VCCPGPPJ	3.3	0.05	0.05
	1.8	0.05	0.05
VCCPGPPK	3.3	0.025	0.025
	1.8	0.025	0.025

Notes: Vccprim_1P05 is the current of all 1.05 voltages combined. Vccp_3P3 is the current for all 3.3 Voltages (minus GPIO power pins) combined. Vccp_1P8 is the current of all 1.8 voltages combined (minus GPIO power pins).
The *Intel*® C620 Series Chipset *PDG* has VCCP_3P3 as 0.6A and not 0.3A as it rolls all the other GPIOs up to the 3.3V rail, minus VCCPGPPA.

Table 4-9. Intel® C620 Series Chipset PCH Estimated I_{CC} (C626) (Sheet 1 of 2)

Voltage Rail	Voltage (V)	S0 Iccmax Current[3] (A)	S0 IccTDC
VCCDSW_3P3	3.3	0.5	0.5
VCCP_1P8	1.8	1.0	0.6
VCCP_3P3	3.3	0.6	0.6
VCCPRIM_AVID	0.85-1.0	16.5	15
VCCPRIM_1P05	1.05	14.5	11.5
VCCPGPPA	3.3	0.1	0.1
	1.8	0.1	0.1
VCCPGPPB	3.3	0.025	0.025
	1.8	0.025	0.025
VCCPGPPC	3.3	0.1	0.1
	1.8	0.1	0.1
VCCPGPPD	3.3	0.025	0.025
	1.8	0.025	0.025
VCCPGPPE	3.3	0.025	0.025
	1.8	0.025	0.025

Table 4-9. Intel® C620 Series Chipset PCH Estimated I_{cc} (C626) (Sheet 2 of 2)

Voltage Rail	Voltage (V)	S0 Iccmax Current[3] (A)	S0 IccTDC
VCCPGPPF	3.3	0.025	0.025
	1.8	0.025	0.025
VCCPGPPG	3.3	0.025	0.025
	1.8	0.025	0.025
VCCPGPPH	3.3	0.05	0.05
	1.8	0.05	0.05
VCCPGPPJ	3.3	0.05	0.05
	1.8	0.05	0.05
VCCPGPPK	3.3	0.025	0.025
	1.8	0.025	0.025

Notes: Vccprim_1P05 is the current of all 1.05 voltages combined. Vccp_3P3 is the current for all 3.3 voltages (minus GPIO power pins) combined. Vccp_1P8 is the current of all 1.8 voltages combined (minus GPIO power pins).
The *Intel*® C620 Series Chipset *PDG* has VCCP_3P3 as 0.6A and not 0.3A as it rolls all the other GPIOs up to the 3.3V rail, minus VCCPGPPA

Table 4-10. Intel® C620 Series Chipset PCH Estimated I_{cc} (C628) (Sheet 1 of 2)

Voltage Rail	Voltage (V)	Published Performance Workload x16 Uplink Only		High Performance Workload, x24 Uplink Enabled	
		S0 Iccmax Current[3] (A)	S0 IccTDC (A)	S0 Iccmax Current[3] (A)	S0 IccTDC (A)
VCCDSW_3P3	3.3	0.5	0.5	0.5	0.5
VCCP_1P8	1.8	1.0	0.6	1.0	0.6
VCCP_3P3	3.3	0.6	0.6	0.6	0.6
VCCPRIM_AVID	0.85-1.0	21.5	18	21.5	19.5
VCCPRIM_1P05	1.05	13.5	10.5	13.5	10.5
VCCPGPPA	3.3	0.1	0.1	0.1	0.1
	1.8	0.1	0.1	0.1	0.1
VCCPGPPB	3.3	0.025	0.025	0.025	0.025
	1.8	0.025	0.025	0.025	0.025
VCCPGPPC	3.3	0.1	0.1	0.1	0.1
	1.8	0.1	0.1	0.1	0.1
VCCPGPPD	3.3	0.025	0.025	0.025	0.025
	1.8	0.025	0.025	0.025	0.025
VCCPGPPE	3.3	0.025	0.025	0.025	0.025
	1.8	0.025	0.025	0.025	0.025
VCCPGPPF	3.3	0.025	0.025	0.025	0.025
	1.8	0.025	0.025	0.025	0.025
VCCPGPPG	3.3	0.025	0.025	0.025	0.025
	1.8	0.025	0.025	0.025	0.025

Table 4-10. Intel® C620 Series Chipset PCH Estimated I_{CC} (C628) (Sheet 2 of 2)

Voltage Rail	Voltage (V)	Published Performance Workload x16 Uplink Only		High Performance Workload, x24 Uplink Enabled	
		S0 Iccmax Current[3] (A)	S0 IccTDC (A)	S0 Iccmax Current[3] (A)	S0 IccTDC (A)
VCCPGPPH	3.3	0.05	0.05	0.05	0.05
	1.8	0.05	0.05	0.05	0.05
VCCPGPPJ	3.3	0.05	0.05	0.05	0.05
	1.8	0.05	0.05	0.05	0.05
VCCPGPPK	3.3	0.025	0.025	0.025	0.025
	1.8	0.025	0.025	0.025	0.025

Notes: Vccprim_1P05 is the current of all 1.05 voltages combined. Vccp_3P3 is the current for all 3.3 voltages (minus GPIO power pins) combined. Vccp_1P8 is the current of all 1.8 voltages combined (minus GPIO power pins).
The *Intel*® C620 Series Chipset *PDG* has VCCP_3P3 as 0.6A and not 0.3A as it rolls all the other GPIOs up to the 3.3V rail, minus VCCPGPPA.

Table 4-11. Intel® C620 Series Chipset Real Time Clock Vcc and Icc

	G3 State	Deep Sleep	S5-S0
VccpRTC	2.0 to 3.2V	2.0 to 3.2V	2.6-3.2V
Icc	6 uA max	36 uA max	500 uA max

Notes: Icc G3 and Icc Deep Sleep is taken at 25 C temperature.
Icc G3 is taken with VccpRTC at 3.0V.

Table 4-12. DC Characteristics of Single Ended Signals (Sheet 1 of 5)

Type	Symbol	Parameter	Min	Max	Unit	Condition	Notes
Associated Signals: GPP_A0_RCIN_N_ESPI_ALERT1_N; GPP_A1_LAD0_ESPI_IO0; GPP_A2_LAD1_ESPI_IO1; GPP_A3_LAD2_ESPI_IO2; GPP_A4_LAD3_ESPI_IO3;GPP_A5_LFRAME_N_ESIP_CS0_N; GPP_A6_SERIRQ_ESPI_CS1_N; GPP_A7_PIRQA_N_ESPI_ALERT0_N; GPP_A8_CLKRUN_N; GPP_A9_CLKOUT_LPC0_ESPI_CLK; GPP_A10_CLKOUT_LPC1; GPP_A11_PME_N; GPP_A12_BMBUSY_N_SXEXITHLDOFF_N; GPP_A13_SWSWARN_N_SUSPWRDNACK; GPP_A14_ESPI_RESET_N; GPP_A15_SUSACK_N; GPP_A16_CLKOUT_LPC2;							
3.3V Operation							
Input	V_{IH}	Input High Voltage	0.625 x V_{CC} 0.5 x V_{CC}		V		1
	V_{IL}	Input Low Voltage		0.25x V_{CC} 0.3 x V_{CC}	V		2
	I_{IL}	Input Leakage Current	-15	15	µA		
	C_{IN}	Input Pin Capacitance		7	pF		
Output	V_{OH}	Output High Voltage	V_{CC} -0.45V		V	I_{oh}=3mA	
	V_{OL}	Output Low Voltage	0	0.45	V	I_{ol}=-3mA	
	R_{pu}	WPU Resistance	5K-30% 20K-30%	5K+30% 20K+30%			
	R_{pd}	WPD Resistance	5K-30% 20K-30%	5K+30% 20K+30%			

Table 4-12. DC Characteristics of Single Ended Signals (Sheet 2 of 5)

Type	Symbol	Parameter	Min	Max	Unit	Condition	Notes
1.8V Operation							
Input	V_{IH}	Input High Voltage	1.27		V		
	V_{IL}	Input Low Voltage		0.58	V		
	I_{IL}	Input Leakage Current	-15	15	µA		
	C_{IN}	Input Pin Capacitance		7	pF		
Output	V_{OH}	Output High Voltage	VCC-0.45		V	I_{oh}=3mA	
	V_{OL}	Output Low Voltage	0	0.45	V	I_{ol}=-3mA	
	R_{pu}	WPU Resistance	5K-30% 20K-30%	5K+30% 20K+30%			
	R_{pd}	WPD Resistance	5K-30% 20K-30%	5K+30% 20K+30%			
Notes: 1. The 0.625 is when set as GPIO, 0.5 is when configured as LPC. 2. The 0.25 is when configured as GPIO, 0.3 when configured as LPC.							
Associated Signals: GPP_A17; GPP_A18; GPP_A19; GPP_A20; GPP_A21;GPP_A22; GPP_A23; GPP_B0_CORE_VID0; GPP_B1_CORE_VID1; GPP_B2; GPP_B3_CPU_GP2; GPP_B4_CPU_GP3; GPP_B5_SRCCLKREQ0_N; GPP_B6_SRCCLKREQ1_N; GPP_B7_SRCCLKEREQ2_N; GPP_B8_SRCCLKREQ3_N; GPP_B9_SRCCLKREQ4_N; BPP_B10_SRCCLKREQ5_N; GPP_B11; GPP_B12_GLB_RST_WARN_N; GPP_B13_PLTRST_N; GPP_B14_SPKR; GPP_B15; GPP_B16; GPP_B17; GPP_B18; GPP_B19; GPP_B20; GPP_B21; GPP_B22; GPP_B23_MEIE_SML1LART_N_PHOT_N; GPP_C0_SMBCLK; GPP_C1_SMBDATA; GPP_C2_SMBALERT_N; GPP_C3_SML0CLK_IE; GPP_C4_SML0DATA_IE; GPP_C5_SML0ALERT_IE_N; GPP_C6_SML1CLK_IE; GPP_C7_SML1DATA_IE; GPP_C8; CPP_C9; GPP_C10; GPP_C11; GPP_C12; GPP_C13; GPP_C14; GPP_C15; GPP_C16; GPP_C17; GPP_C18; GPP_C19; GPP_C20; GPP_C21; GPP_C22; GPP_C23; GPP_D0; GPP_D1; GPP_D2; GPP_D3; GPP_D4; GPP_D5; GPP_D6; GPP_D7; DPP_D8; GPP_D9_SSATA_DEVSLP3; GPP_D10_SSATA_DEVSLP4; GPP_D11_SSATA_DEVSLP5; GPP_D12_SSATA_SDATAOUT1; GPP_D13_SML0BCLK_IE; GPP_D14_SML0BDATA_IE; GPP_D15_SSATA_DATAOUT0; GPP_D16_SML0BALERT_IE_N; GPP_D17; GPP_D18; GPP_D19; GPP_D20; GPP_D21_IE_UART_RX; GPP_D22_IE_UART_TX; GPP_D23; GPP_E0_SATAXPCIE0_SATAGP0; GPP_E1_SATAXPCIE1_SATAGP1; GPP_E2_SATAXPCIE2_SATAGP2; GPP_E3_CPU_GP0; GPP_E4_SATA_DEVSLP0; GPP_E5_SATA_DEVSLP1; GPP_E6_SATA_DEVSLP2; GPP_E7_CPU_GP1; GPP_E8_SATA_LED_N; GPP_E9_USB2_OC0_N; GPP_E10_USB2_OC1_N; GPP_E11_USB2_OC2_N; GPP_E12_USB2_OC3_N;							
3.3V Operation							
Input	V_{IH}	Input High Voltage	0.625 x V_{CC}		V		
	V_{IL}	Input Low Voltage		0.25x V_{CC}	V		
	I_{IL}	Input Leakage Current	-15	15	µA		
	C_{IN}	Input Pin Capacitance		7	pF		
Output	V_{OH}	Output High Voltage	V_{CC} -0.45V		V	I_{oh}=3mA	
	V_{OL}	Output Low Voltage	0	0.45	V	I_{ol}=-3mA	
	R_{pu}	WPU Resistance	5K-30% 20K-30%	5K+30% 20K+30%			
	R_{pd}	WPD Resistance	5K-30% 20K-30%	5K+30% 20K+30%			
1.8V Operation							
Input	V_{IH}	Input High Voltage	1.27		V		
	V_{IL}	Input Low Voltage		0.58	V		
	I_{IL}	Input Leakage Current	-15	15	µA		
	C_{IN}	Input Pin Capacitance		7	pF		

Table 4-12. DC Characteristics of Single Ended Signals (Sheet 3 of 5)

Type	Symbol	Parameter	Min	Max	Unit	Condition	Notes
Output	V_{OH}	Output High Voltage	VCC-0.45		V	I_{oh}=3mA	1
	V_{OL}	Output Low Voltage	0	0.45	V	I_{ol}=-3mA	
	R_{pu}	WPU Resistance	5K-30% 20K-30%	5K+30% 20K+30%			
	R_{pd}	WPD Resistance	5K-30% 20K-30%	5K+30% 20K+30%			
Associated Signals: GPP_F0_SATAXPCIE3_GP3; GPP_F1_SATAXPCIE4_SATAGP4; GPP_F2_SATAXPCIE5_SATAGP5; GPP_F3_SATAXPCIE6_SATAGP6; GPP_F4_SATAXPCIE7_SATAGP7; GPP_F5_SATA_DEVSLP3; GPP_F6_SATA_DEVSLP4; GPP_F7_SATA_DEVSLP5; GPP_F8_SATA_DEVSLP6; GPP_F9_SATA_DEVSLP7; GPP_F10_SATA_SCLOCK_GPP_F11_SATA_SLOAD; GPP_F12_SATA_SDATAOUT1; GPP_F13_SATA_SDATAOUT0; GPP_F14_SSATA_LED_N; GPP_F15_USB2_OC4_N; GPP_F16_USB2_OC5_N; GPP_F17_USB2_OC6_N; GPP_F18_USB2_OC7_N; GPP_F19_LAN_SMBCLK; GPP_F20_LAN_SMBDATA; GPP_F21_LAN_SMBLART_N; GPP_F22_SSATA_CLOCK; GPP_F23_SSAT_LOAD; GPP_G0_FANTACH0_FANTACH0IE; GPP_G1_FANTACH1_FANTACH1IE; GPP_G2_FANTACH2_FANTACH2IE; GPP_G3_FANTACH3_FANTACH3IE; GPP_G4_FANTACH4_FANTACH4IE; GPP_G5_FANTACH5_FANTACH5IE; GPP_G6_FANTACH6_FANTACH6IE; GPP_G7_FANTACH7_FANTACH7IE; GPP_G8_FANPWM0_FAN_PWM0IE; GPP_G9_FANPWM1_FANPWM1IE; GPP_G10_FANPWM2_FANPWM2IE; GPP_G11_FANPWM3FAN_PWM3IE; GPP_G12; GPP_G13;GPP_G14; GPP_G15; GPP_G16; GPP_G17_ADR_COMPLETE; GPP_G18_NMI_N; GPP_G19_SMI_N; GPP_G20_SSATA_DEVSLP0; GPP_G21_SSATA_DEVSLP1; GPP_G22_SSATA_DEVSLP2; GPP_G23_SSATAXPCIE0_SSATAGP0; GPP_H0_SRCCLKREQ6_N; GPP_H1_SRCCLKREQ7_N; GPP_H2_SRCCLKREQ8_N; GPP_H3_SRCCLKREQ9_N; GPP_H4_SRCCLKREQ10_N; GPP_H5_SRCCLKREQ11_N; GPP_H6_SRCCLKREQ12_N; GPP_H7_SRCCLKREQ13_N; GPP_H8_SRCCLKREQ_14_N; GPP_H9_SRCCLKREQ15_N; GPP_H10_SML2CLK_IE; GPP_H11_SML2DAT_IE; GPP_H12_SML2ALERT_N_IE; GPP_H13_SML3_CLK_IE; GPP_H14_SLM3DATA_IE; GPP_H15_SLM3ALERT_N_IE_N; GPP_H16_SML4_CLK_IE; GPP_H17_SML4_DATA_IE; GPP_H18_SML4ALERT_N_IE_N; GPP_H19_SSATAXPCIE1_SSATAGP1; GPP_H20_SSATAXPCIE2_SSATAGP2; GPP_H21_SSATAXPCIE3_SSATAGP3; GPP_H22_SSATAPCIE4_SSATAGP4; GPP_H23_SSATAXPCIE5_SSATAGP5;							
3.3V Operation							
Input	V_{IH}	Input High Voltage	0.625 x V_{CC}		V		
	V_{IL}	Input Low Voltage		0.25x V_{CC}	V		
	I_{IL}	Input Leakage Current	-15	15	µA		
	C_{IN}	Input Pin Capacitance		7	pF		
Output	V_{OH}	Output High Voltage	V_{CC} -0.45V		V	I_{oh}=3mA	
	V_{OL}	Output Low Voltage	0	0.45	V	I_{ol}=-3mA	
	R_{pu}	WPU Resistance	5K-30% 20K-30%	5K+30% 20K+30%			
	R_{pd}	WPD Resistance	5K-30% 20K-30%	5K+30% 20K+30%			
1.8V Operation							
Input	V_{IH}	Input High Voltage	1.27		V		
	V_{IL}	Input Low Voltage		0.58	V		
	I_{IL}	Input Leakage Current	-15	15	µA		
	C_{IN}	Input Pin Capacitance		7	pF		
Output	V_{OH}	Output High Voltage	VCC-0.45		V	I_{oh}=3mA	
	V_{OL}	Output Low Voltage	0	0.45	V	I_{ol}=-3mA	
	R_{pu}	WPU Resistance	5K-30% 20K-30%	5K+30% 20K+30%			
	R_{pd}	WPD Resistance	5K-30% 20K-30%	5K+30% 20K+30%			

Table 4-12. DC Characteristics of Single Ended Signals (Sheet 4 of 5)

Type	Symbol	Parameter	Min	Max	Unit	Condition	Notes
Associated Signals: GPP_I0_LAN_TDO; GPP_I1_LAN_TCK; GPP_I2_LAN_TMS; GPP_I3_LAN_TDI; GPP_I4_DO_RESET_IN_N; GPP_I5_DO_RESET_OUT_N; GPP_I6_RESET_DONE; GPP_I7_LAN_TRST_N; GPP_I8_PCI_DIS_N; GPP_I9_LAN_DIS_N; GPP_I10; GPP_J0_LAN_LED_P0_0; GPP_J1_LAN_LED_P0_1; GPP_J2_LAN_LED_P1_0; GPP_J3_LAN_LED_P1_1; GPP_J4_LAN_LED_P2_0; GPP_J5_LAN_LED_P2_1; GPP_J6_LAN_LED_P3_0; GPP_J7_LAN_LED_P3_1; GPP_J8_LAN_I2C_SCL_MDC_P0; GPP_J9_LAN_I2C_SDA_MDPIO_P0; GPP_J10_LAN_I2C_SCL_MDC_P1; GPP_J11_LAN_I2C_SDA_MDPIO_P1; GPP_J12_LAN_I2C_SCL_MDC_P2; GPP_J13_LAN_I2C_SDA_MDPIO_P2; GPP_J14_LAN_I2C_SCL_MDC_P3; GPP_J15_LAN_I2C_SDA_MDPIO_P3; GPP_J16_LAN_SDP_P0_0; GPP_J17_LAN_SDP_P0_1; GPP_J18_LAN_SDP_P1_0; GPP_J19_LAN_SDP_P1_1; GPP_J20_LAN_SDP_P2_0; GPP_J21_LAN_SDP_P2_1; GPP_J22_LAN_SDP_P3_0; GPP_J23_LAN_SDP_P3_1; GPP_K0_LAN_NCSI_CLK_IN; GPP_K1_LAN_NCSI_TXD0; GPP_K2_LAN_NCSI_TXD1;GPP_K3_LAN_NCSI_TX_EN; GPP_K4_LAN_NCSI_CRS_DV; GPP_K5_LAN_NCSI_RXD0; GPP_K6_LAN_NCSI_RXD1; GPP_K7; GPP_K8_LAN_NCSI_ARB_IN; GPP_K9_LAN_NCSI_ARB_OUT; GPP_K10_PE_RST_N;HDA_BCLK; HDA_RST_N; HDA_SDI_0; HDA_SDI_1; HDA_SDO; HDA_SYNC;SPI0_CLK; SPI0_FLASH_CS0_N; SPIO_FLASH_CS1_N; SPI0_IO2; SPI0_IO3; SPI0_MISO_IO1; SPIO_MOSI_IO0; SPIO_TPM_CS_N							
3.3V Operation							
Input	V_{IH}	Input High Voltage	0.625 x V_{CC}		V		
	V_{IL}	Input Low Voltage		0.25x V_{CC}	V		
	I_{IL}	Input Leakage Current	-15	15	µA		
	C_{IN}	Input Pin Capacitance		7	pF		
Output	V_{OH}	Output High Voltage	V_{CC} -0.45V		V	I_{oh}=3mA	
	V_{OL}	Output Low Voltage	0	0.45	V	I_{ol}=-3mA	
	R_{pu}	WPU Resistance	5K-30% 20K-30%	5K+30% 20K+30%			
	R_{pd}	WPD Resistance	5K-30% 20K-30%	5K+30% 20K+30%			
1.8V Operation							
Input	V_{IH}	Input High Voltage	1.27		V		
	V_{IL}	Input Low Voltage		0.58	V		
	I_{IL}	Input Leakage Current	-15	15	µA		
	C_{IN}	Input Pin Capacitance		7	pF		
Output	V_{OH}	Output High Voltage	VCC-0.45		V	I_{oh}=3mA	1
	V_{OL}	Output Low Voltage	0	0.45	V	I_{ol}=-3mA	
	R_{pu}	WPU Resistance	5K-30% 20K-30%	5K+30% 20K+30%			
	R_{pd}	WPD Resistance	5K-30% 20K-30%	5K+30% 20K+30%			
Associated Signals: CLOCKSE_BMCCLK; GPD0; GPD1_ACPRESENT; GPD2_GBE_WAKE_N; GPD3_PWRBTN_N; GPD4_SLP_S3_N; GPD5_SLP_S4_N; GPD6_SLP_A_N; GPD7; GPD8_SUSCLK; GPD9; GPD10_SLP_S5_N; GPD11_GBEPHY; SLP_GBE_N; SLP_SUS_N; SYS_PWROK; SYS_RESET_N; WAKE_N; PCH_PWROK;							
3.3V Operation							
Input	V_{IH}	Input High Voltage	0.625 x V_{CC}		V		
	V_{IL}	Input Low Voltage		0.25x V_{CC}	V		
	I_{IL}	Input Leakage Current	-15	15	µA		
	C_{IN}	Input Pin Capacitance		7	pF		

Table 4-12. DC Characteristics of Single Ended Signals (Sheet 5 of 5)

Type	Symbol	Parameter	Min	Max	Unit	Condition	Notes
Output	V_{OH}	Output High Voltage	V_{CC} -0.45V		V	I_{oh}=3mA	
	V_{OL}	Output Low Voltage	0	0.45	V	I_{ol}=-3mA	
	R_{pu}	WPU Resistance	5K-30% 20K-30%	5K+30% 20K+30%			
	R_{pd}	WPD Resistance	5K-30% 20K-30%	5K+30% 20K+30%			
Associated Signals: CLOCKSE_PMSYNCCLK1; CLOCKSE_PMSYNCCLK2; GPP_L2_TESTCH0_D0; GPP_L3_TESTCH0_D1; GPP_L4_TESTCH0_D2; GPP_L5_TESTCH0_D3; GPP_L6_TESTCH0_D4; GPP_L7_TESTCH0_D5; GPP_L8_TESTCH0_D6; GPP_L9_TESTCH0_D7; GPP_L10_TESTCH0_CLK; GPP_L11_TESTCH1_D0; GPP_L12_TESTCH1_D1; GPP_L13_TESTCH1_D2; GPP_L14_TESTCH1_D3; GPP_L15_TESTCH1_D4; GPP_L16_TESTCH1_D5; GPP_L17_TESTCH1_D6; GPP_L18_TEST1_D7; GPP_L19_TESTCH1_D8; JTAG_TCK; JTAG_TDI; JTAG_TDO; JTAG_TMS; JTAGX; PECI; PLTRST_CPU_N; PM_SYNC; PM_SYNC2; PRDY_N; PREQ_N; THRMTRIP_N; TRIGGER_IN; TRIGGER_OUT.							
1.05V Operation							
Input	V_{IH}	Input High Voltage	0.65 x V_{CC}		V		
	V_{IL}	Input Low Voltage		0.35x V_{CC}	V		
	I_{IL}	Input Leakage Current	-50	50	µA		
	C_{IN}	Input Pin Capacitance		5	pF		
Output	V_{OH}	Output High Voltage	V_{CC} -0.2V		V	I_{oh}=2mA	3
	V_{OL}	Output Low Voltage	0	0.2	V	I_{ol}=-3mA	3
	R_{pu}	WPU Resistance	5K-30% 20K-30%	5K+30% 20K+30%	R_{pu}		
	R_{pd}	WPD Resistance	5K-30% 20K-30%	5K+30% 20K+30%	R_{pd}		
Notes: 1. The V_{OH} specification does not apply to open-collector or open-drain drivers. Signals of this type must have an external pull-up resistor, and that is what determines the high-output voltage level. 2. Input characteristics apply when a signal is configured as Input or to signals that are only Inputs. Output characteristics apply when a signal is configured as an Output or to signals that are only Outputs. 3. For JTAGX, I_{oh}=8mA and I_{ol}=-12 mA							
Associated Signals: INTRUDER_N, RSMRST_N, PCH_PWROK, DSW_PWROK, SRTCRST_N, RTCRST_N							
Input	V_{IH}	Input High Voltage	0.65 x V_{CCRTC}	V_{CCRTC}+0.5	V		4, 6
	V_{IL}	Input Low Voltage	-0.5	0.3 x V_{CCRTC}	V		6
Notes: 1. The V_{OH} specification does not apply to open-collector or open-drain drivers. Signals of this type must have an external pull-up resistor, and that is what determines the high-output voltage level. 2. Input characteristics apply when a signal is configured as Input or to signals that are only Inputs. Output characteristics apply when a signal is configured as an Output or to signals that are only Outputs. 3. V_{CCRTC} is the voltage applied to the V_{CCRTC} well of the PCH. When the system is in G3 state, it is generally supplied by the coin cell battery. In S5 or greater state, it is supplied by VCCSUS3_3. 4. V_{IH} min should not be used as the reference point for T200 timing. See T200 specification for the measurement point detail. 5. These buffers have input hysteresis. V_{IH} levels are for rising edge transitions and V_{IL} levels are for falling edge transitions.							

Table 4-13. Differential Signals Characteristics (Sheet 1 of 4)

Symbol	Parameter	Min	Max	Unit	Conditions	Notes
Associated Signals: PCIe						9, 10
Gen 1						
VTX-DIFF P-P	Differential Peak to Peak Output Voltage	See Table 4-18 of the *PCIe Gen3 Base Specification*, Rev. 1.0.				1

Table 4-13. Differential Signals Characteristics (Sheet 2 of 4)

Symbol	Parameter	Min	Max	Unit	Conditions	Notes
VTX-DIFF P-P - Low	Low power differential Peak to Peak Output Voltage	See Table 4-18 of the *PCIe Gen3 Base Specification*, Rev. 1.0.				
VTX_CM-ACp	TX AC Common Mode Output Voltage (2.5 GT/s)	See Table 4-18 of the *PCIe Gen3 Base Specification*, Rev. 1.0.				
ZTX-DIFF-DC	DC Differential TX Impedance	See Table 4-18 of the *PCIe Gen3 Base Specification*, Rev. 1.0.				
VRX-DIFF p-p	Differential Input Peak to Peak Voltage	See Table 4-24 of the *PCIe Gen3 Base Specification*, Rev. 1.0.				1
VRX_CM-ACp	AC peak Common Mode Input Voltage	See Table 4-24 of the *PCIe Gen3 Base Specification*, Rev. 1.0.				
Gen 2						
VTX-DIFF P-P	Differential Peak to Peak Output Voltage	See Table 4-18 of the *PCIe Gen3 Base Specification*, Rev. 1.0.				
VTX-DIFF P-P - Low	Low power differential Peak to Peak Output Voltage	See Table 4-18 of the *PCIe Gen3 Base Specification*, Rev. 1.0.				
VTX_CM-Acp-p	TX AC Common Mode Output Voltage (5GT/s)	See Table 4-18 of the *PCIe Gen3 Base Specification*, Rev. 1.0.				
ZTX-DIFF-DC	DC Differential TX Impedance	See Table 4-18 of the *PCIe Gen3 Base Specification*, Rev. 1.0.				
VRX-DIFF p-p	Differential Input Peak to Peak Voltage	See Table 4-24 of the *PCIe Gen3 Base Specification*, Rev. 1.0.				
VRX_CM-ACp	AC peak Common Mode Input Voltage	See Table 4-24 of the *PCIe Gen3 Base Specification*, Rev. 1.0.				
Gen 3						
VTX-DIFF P-P	Differential Peak to Peak Output Voltage	See Table 4-18 of the *PCIe Gen3 Base Specification*, Rev. 1.0.				
VTX-DIFF P-P - Low	Low power differential Peak to Peak Output Voltage	See Table 4-18 of the *PCIe Gen3 Base Specification*, Rev. 1.0.				
VTX_CM-Acp-p	TX AC Common Mode Output Voltage (5GT/s)	See Table 4-18 of the *PCIe Gen3 Base Specification*, Rev. 1.0.				
ZTX-DIFF-DC	DC Differential TX Impedance	See Table 4-18 of the *PCIe Gen3 Base Specification*, Rev. 1.0.				
VRX-DIFF p-p	Differential Input Peak to Peak Voltage	See Table 4-22, 4-23 and 4-24 of the the *PCIe Gen3 Base Specification*, Rev. 1.0.				
VRX_CM-ACp	AC peak Common Mode Input Voltage	See Table 4-22 and 4-24 of the the *PCIe Gen3 Base Specification*, Rev. 1.0.				
Associated Signals: SATA						
VIMIN- Gen1i	Minimum Input Voltage - 1.5 Gb/s internal SATA	See Table 57 in the *SerialATA Revision 3.2 Specification*.				2
VIMAX-Gen1i	Maximum Input Voltage - 1.5 Gb/s internal SATA	See Table 57 in the *SerialATA Revision 3.2 Specification*.				2
VIMIN-Gen1m	Minimum Input Voltage - 1.5 Gb/s eSATA	See Table 57 in the *SerialATA Revision 3.2 Specification*.				2
VIMAX-Gen1m	Maximum Input Voltage - 1.5 Gb/s eSATA	See Table 57 in the *SerialATA Revision 3.2 Specification*.				2
VIMIN-Gen2i	Minimum Input Voltage - 3.0 Gb/s internal SATA	See Table 57 in the *SerialATA Revision 3.2 Specification*.				2
VIMAX-Gen2i	Maximum Input Voltage - 3.0 Gb/s internal SATA	See Table 57 in the *SerialATA Revision 3.2 Specification*.				2
VIMIN-Gen2m	Minimum Input Voltage - 3.0 Gb/s eSATA	See Table 57 in the *SerialATA Revision 3.2 Specification*.				2

Table 4-13. Differential Signals Characteristics (Sheet 3 of 4)

Symbol	Parameter	Min	Max	Unit	Conditions	Notes
VIMAX-Gen2m	Maximum Input Voltage - 3.0 Gb/s eSATA	See Table 57 in the *SerialATA Revision 3.2 Specification*.				2
VIMIN-Gen3i	Minimum Input Voltage - 6.0 Gb/s internal SATA	See Table 57 in the *SerialATA Revision 3.2 Specification*.				2
VIMAX-Gen3i	Maximum Input Voltage - 6.0 Gb/s internal SATA	See Table 57 in the *SerialATA Revision 3.2 Specification*.				2
VOMIN-Gen1i,m	Minimum Output Voltage 1.5 Gb/s internal and eSATA	See Table 54 in the *SerialATA Revision 3.2 Specification*.				3
VOMAX-Gen1i,m	Maximum Output Voltage 1.5 Gb/s internal and eSATA	See Table 54 in the *SerialATA Revision 3.2 Specification*.				3
VOMIN-Gen2i,m	Minimum Output Voltage 3.0 Gb/s internal and eSATA	See Table 54 in the *SerialATA Revision 3.2 Specification*.				3
VOMAX-Gen2i,m	Maximum Output Voltage 3.0 Gb/s internal and eSATA	See Table 54 in the *SerialATA Revision 3.2 Specification*.				3
VOMIN-Gen3i	Minimum Output Voltage 6.0 Gb/s internal SATA	See Table 54 in the *SerialATA Revision 3.2 Specification*.				3
VOMAX-Gen3i	Maximum Output Voltage 6.0 Gb/s internal SATA	See Table 54 in the *SerialATA Revision 3.2 Specification*.				3
Associated Signals: USB2						
VDI	Differential Input Sensitivity	See Table 7-7 of the *USB 2.0 Specification*.				4, 6
VCM	Differential Common Mode Range	See Table 7-7 of the *USB 2.0 Specification*.				5, 6
VSE	Single-Ended Receiver Threshold	See Table 7-7 of the *USB 2.0 Specification*.				6
VCRS	Output Signal Crossover Voltage	See Table 7-7 of the *USB 2.0 Specification*.				6
VOL	Output Low Voltage	See Table 7-7 of the *USB 2.0 Specification*.				7
VOH	Output High Voltage	See Table 7-7 of the *USB 2.0 Specification*.				7
VHSSQ	HS Squelch Detection Threshold	See Table 7-7 of the *USB 2.0 Specification*.				7
VHSDSC	HS Disconnect Detection Threshold	See Table 7-7 of the *USB 2.0 Specification*.				7
VHSCM	HS Data Signaling Common Mode Voltage Range	See Table 7-7 of the *USB 2.0 Specification*.				7
VHSOI	HS Idle Level	See Table 7-7 of the *USB 2.0 Specification*.				7
VHSOH	HS Data Signaling High	See Table 7-7 of the *USB 2.0 Specification*.				7
VHSOL	HS Data Signaling Low	See Table 7-7 of the *USB 2.0 Specification*.				7
VCHIRPJ	Chirp J Level	See Table 7-7 of the *USB 2.0 Specification*.				7
VCHIRPK	Chirp K Level	See Table 7-7 of the *USB 2.0 Specification*.				7
Associated Signals: USB3						
VTX-DIFF-PP	Differential Peak to Peak Output Voltage	See Table 6-10 of the *USB 3.0 Specification*.				
VTX-DIFF P-P - Low	Low power differential Peak to Peak Output Voltage	See Table 6-10 of the *USB 3.0 Specification*.				8

Table 4-13. Differential Signals Characteristics (Sheet 4 of 4)

Symbol		Parameter	Min	Max	Unit	Conditions	Notes
VTX_CM-Acp-p		TX AC Common Mode Output Voltage (5GT/s)	See Table 6-11 of the *USB 3.0 Specification*.				
RTX-DIFF-DC		DC Differential TX Impedance	See Table 6-10 of the *USB 3.0 Specification*.				
VRX-DIFF p-p		Differential Input Peak to Peak Voltage	See Table 6-14 of the *USB 3.0 Specification*.				
VRX_CM-AC-p		AC peak Common Mode Input Voltage (5GT/s)	See Table 6-14 of the *USB 3.0 Specification*.				
Associated Signals: CLKOUT_ITPXDP_P/N, CLKOUTNSSCCAP0/1_P/N, CLKOUT_PLAT0/1_P/N, CLKOUT_SRC_P/N_0/15							
Output	V_{Swing}	Differential Output Swing	300	—	mV		12
	V_{Cross}	Crossing Point Voltage	250	550	mV		11,13,14
	V_{Cross_Delta}	Variation of VCROSS	—	140	mV		11,13,17
	V_{Max}	Max Output Voltage	—	1.15	V		11,15
	V_{Min}	Min Output Voltage	-0.3	—	V		11,16

Notes:
1. PCI Express mVdiff p-p = 2*|PCIE[x]_TXP – PCIE[x]_TXN|; PCI Express mVdiff p-p = 2*|CIE[x]_RXP – PCIE[x]_RXN|
2. SATA Vdiff, RX (V_{IMAX}/V_{IMIN}) is measured at the SATA connector on the receiver side (generally, the motherboard connector), where SATA mVdiff p-p = 2*|SATA[x]RXP – SATA[x]RXN|.
3. SATA Vdiff, tx (V_{OMIN}/V_{OMAX}) is measured at the SATA connector on the transmit side (generally, the motherboard connector), where SATA mVdiff p-p = 2*|SATA[x]TXP – SATA[x]TXN|
4. V_{DI} = | USBPx[P] – USBPx[N] |
5. Includes VDI range.
6. Applies to low-speed/full-speed USB.
7. Applies to high-speed USB 2.0.
8. USB 3.0 mVdiff p-p = 2*|USB3Rp[x] – USB3Rn[x]|; USB 3.0 mVdiff p-p = 2*|USB3Tp[x] – USB3Tn[x]|
9. For PCIe, GEN1, GEN and GEN3 correspond to the PCIe base specification revision 1, 2 and 3.
10. PCIe specifications are also applicable to the LAN port.
11. Measurement taken from single-ended waveform on a component test board
12. Measurement taken from differential waveform on a component test board
13. V_{Cross} is defined as the voltage where Clock = Clock#.
14. Only applies to the differential rising edge (that is, Clock rising and Clock# falling).
15. The max voltage including overshoot
16. The min voltage including undershoot
17. The total variation of all V_{Cross} measurements in any particular system. Note that this is a subset of V_{Cross} MIN/MAX (V_{Cross} absolute) allowed. The intent is to limit V_{Cross} induced modulation by setting V_{Cross_Delta} to be smaller than V_{Cross} absolute.

4.3 AC Characteristics

Table 4-14. PCI Express* Interface Timings (Sheet 1 of 2)

Symbol	Parameter	Min	Max	Unit	Figures	Notes
Transmitter and Receiver Timings						
UI (Gen1)	Unit Interval – PCI Express*	399.88	400.12	ps		5
UI (Gen 2)	Unit Interval – PCI Express	199.9	200.1	ps		5
UI (GEN3)	Unit Interval – PCI Express	124.96	125.03	ps		
T_{TX-EYE} (Gen 1 / Gen 2)	Minimum Transmission Eye Width	0.7	—	UI	4-1	1,2
$T_{TX-RISE/Fall}$ (Gen 1)	D+/D- TX Out put Rise/Fall time	0.125	—	UI		1,2

Table 4-14. PCI Express* Interface Timings (Sheet 2 of 2)

Symbol	Parameter	Min	Max	Unit	Figures	Notes
Transmitter and Receiver Timings						
$T_{TX-RISE/Fall}$ (Gen 2)	D+/D- TX Out put Rise/Fall time	0.15	—	UI		1,2
$T_{TX-RISE/Fall}$ (Gen 3)	D+/D- TX Out put Rise/Fall time	0.15	—	UI		
T_{RX-EYE} (Gen 1 / Gen 2)	Minimum Receiver Eye Width	0.40	—	UI	4-2	3,4
$T_{Min-Pulse}$ (Gen 2)	Instantaneous lone Pulse Width	0.9	—	UI		
$T_{Min-Pulse}$ (Gen 3)	Instantaneous lone Pulse Width	0.91	—	UI		

Notes:
1. Specified at the measurement point into a timing and voltage compliance test load and measured over any 250 consecutive TX UIs. (Also refer to the transmitter compliance eye diagram.)
2. A T_{TX-EYE} = 0.70 UI provides for a total sum of deterministic and random jitter budget of $T_{TXJITTER-MAX}$ = 0.30 UI for the Transmitter collected over any 250 consecutive TX UIs. The $T_{TXEYE-MEDIAN-to-MAX-JITTER}$ specification ensures a jitter distribution in which the median and the maximum deviation from the median is less than half of the total TX jitter budget collected over any 250 consecutive TX UIs. It should be noted that the median is not the same as the mean. The jitter median describes the point in time where the number of jitter points on either side is approximately equal as opposed to the averaged time value.
3. Specified at the measurement point and measured over any 250 consecutive UIs. The test load documented in the PCI Express specification 2.0 should be used as the RX device when taking measurements (also refer to the Receiver compliance eye diagram). If the clocks to the RX and TX are not derived from the same reference clock, the TX UI recovered from 3500 consecutive UI must be used as a reference for the eye diagram.
4. A T_{RX-EYE} = 0.40 UI provides for a total sum of 0.60 UI deterministic and random jitter budget for the Transmitter and interconnect collected any 250 consecutive UIs. The $T_{RX-EYE-MEDIAN-to--MAX-JITTER}$ specification ensures a jitter distribution in which the median and the maximum deviation from the median is less than half of the total 0.6 UI jitter budget collected over any 250 consecutive TX UIs. It should be noted that the median is not the same as the mean. The jitter median describes the point in time where the number of jitter points on either side is approximately equal as opposed to the averaged time value. If the clocks to the RX and TX are not derived from the same reference clock, the TX UI recovered from 3500 consecutive UI must be used as the reference for the eye diagram.
5. Nominal Unit Interval is 400 ps for 2.5 GT/s and 200 ps for 5 GT/s.

Figure 4-1. PCI Express Transmitter Eye

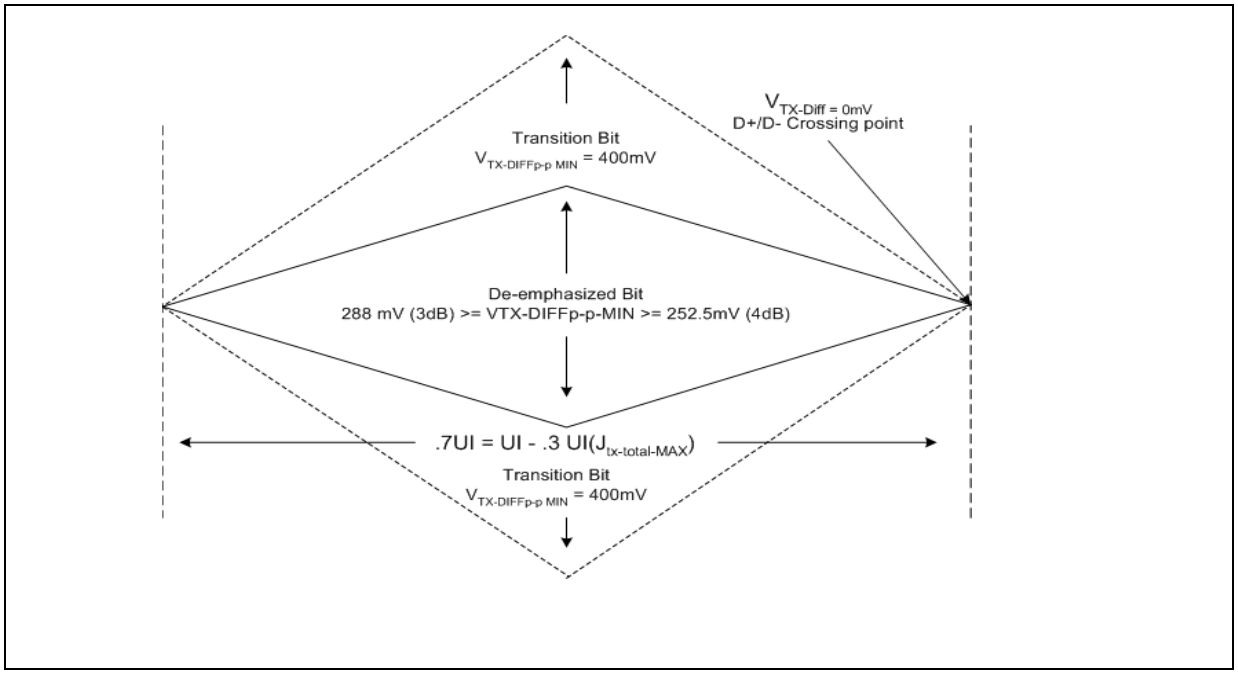


Figure 4-2. PCI Express Receiver Eye

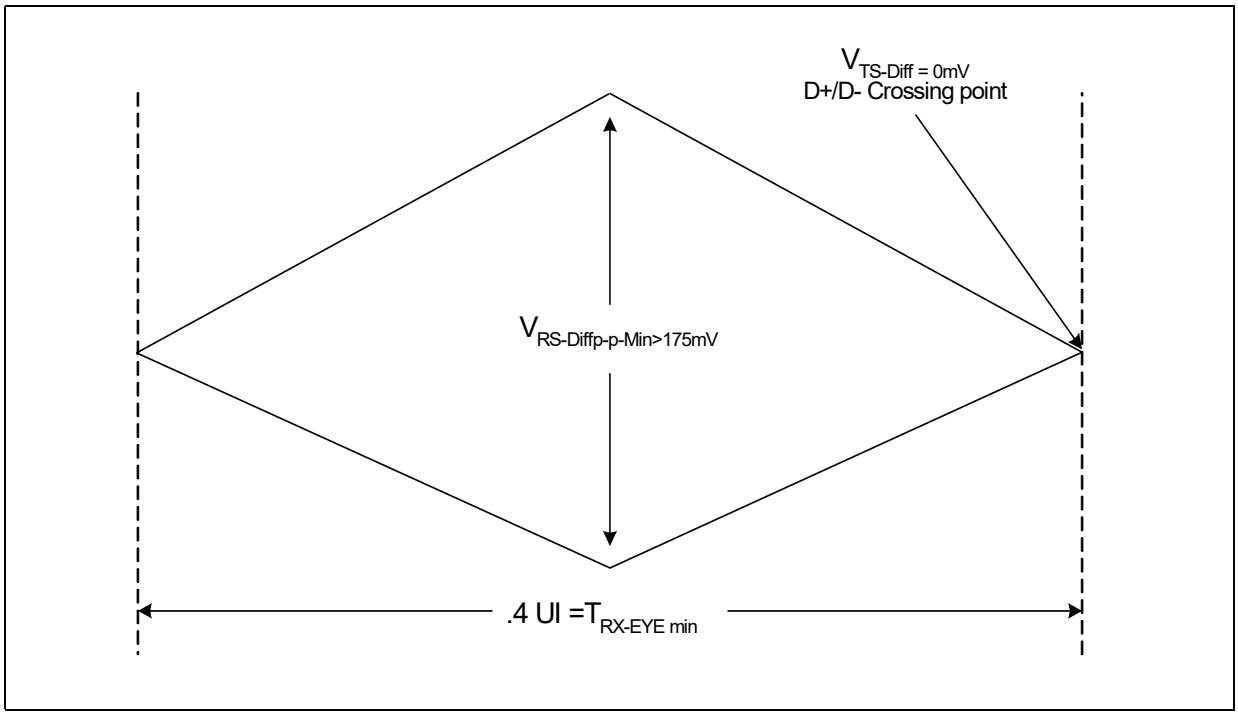


Table 4-15. Intel® C620 Series Chipset Clock Timings (Sheet 1 of 3)

Sym	Parameter	Min	Max	Unit	Notes	Figure
LPC Clock (CLKOUT_LPC[2:0])						
t1	Period	41.16	42.18	ns		4-3
t2	High Time	16	26	ns		4-3
t3	Low Time	16	26	ns		4-3
	Duty Cycle	45	55	%		
	Jitter	—	510	ps	8, 9	
	Rising/Falling Edge Rate	1	4	v/ns	7, 11	4-5
, CLKOUT_PLAT[1:0] _P/N, CLKOUT_ITPXDP_P/N						
Period	Period SSC On	9.847	10.203	ns	19	4-4
Period	Period SSC Off	9.849	10.151	ns		4-4
DtyCyc	Duty Cycle	45	55	%		4-4
V_Swing	Differential Output Swing	300	—	mV		4-4
Slew_rise	Rising Edge Rate	0.6	6	V/ns		4-4
Slew_fall	Falling Edge Rate	0.6	6	V/ns		4-4
	Output-to-output skew between SRC_CLocks [5:0]		87	ps		
	Output-to-output skew between SRC_Clocks [15:6]		87	ps		
	Output-to-output skew between SRC_Clocks [5:0] and SRC_Clocks[15:6]		137	ps		
	Output-to-outputskew between all other clocking relationships		157	ps		
	Cycle-to-cycle jitter		50	ps		

Table 4-15. Intel® C620 Series Chipset Clock Timings (Sheet 2 of 3)

Sym	Parameter	Min	Max	Unit	Notes	Figure
	Common refck RMS Jitter (PCIe Gen2)		2.25	ps		
	Common refclk RMS Jitter (PCIe Gen3)		0.75	ps		
	Common Refclk RMS Jitter (PCIe Gen4)		0.5	ps	18	
SSC	Spread Spectrum	0	0.5	%		
CLKOUT_NSSCCAP[1:0]_P/N						
Slew_rise	Rising Edge Rate	0.6	6	V/ns		4-4
Slew_fall	Falling Edge Rate	0.6	6	V/ns		4-4
V_Swing	Differential Output Swing	300	—	mV		4-4
DtyCyc	Duty Cycle	45	55	%		4-4
CLOCKSE_BMCCLK						
	Period	20.32	21.34	ns		
	T High	7.02	12.51	ns		
	T Low	6.63	12.30	ns		
	Duty Cycle	40	60	%		
	Rising Edge Rate	0.8	4.0	V/ns	5	
	Falling Edge Rate	0.8	4.0	V/ns	5	
	cycle to cycle jitter		410	ps		
SMBus/SMLink0, SMLink0b; SMLink1,SMLink2, SMLink3, SMLink4 (SMBCLK, SML0CLK, SML0BCLK, SML1CLK, SML2CLK, SML3CLK, SML4CLK						
f_{smb}	Operating Frequency	10	105	kHz		
t18	High Time	4.0	50	µs	2	4-4
t19	Low Time	4.7	—	µs		4-4
t20	Rise Time	—	1000	ns		4-4
t21	Fall Time	—	300	ns		4-4
SMLink0, SMLink0b; SMLink1,SMLink2, SMLink3, SMLink4 (SML0CLK, SML0BCLK, SML1CLK, SML2CLK, SML3CLK, SML4CLK) (Fast Mode)						
f_{smb}	Operating Frequency	0	400	kHz		
t18_SMLFM	High Time	0.6	50	µs	2	4-4
t19_SMLFM	Low Time	1.3	—	µs		4-4
t20_SMLFM	Rise Time	—	300	ns		4-4
t21_SMLFM	Fall Time	—	300	ns		4-4
SMLink0, SMLink0b; SMLink1,SMLink2, SMLink3, SMLink4 (SML0CLK, SML0BCLK, SML1CLK, SML2CLK, SML3CLK, SML4CLK)(Fast Mode Plus)						
f_{smb}	Operating Frequency	0	1000	kHz		
t18_SMLFMP	High Time	0.26	—	µs	2	4-4
t19_SMLFMP	Low Time	0.5	—	µs		4-4
t20_SMLFMP	Rise Time	—	120	ns		4-4
t21_SMLFMP	Fall Time	—	120	ns		4-4
HDA_BLK (Intel® High Definition Audio)						
f_{HDA}	Operating Frequency	24.0		MHz		
	Frequency Tolerance	—	100	ppm		

Table 4-15. Intel® C620 Series Chipset Clock Timings (Sheet 3 of 3)

Sym	Parameter	Min	Max	Unit	Notes	Figure
t26a	Input Jitter (refer to Clock Chip Specification)	—	300	ppm		
t27a	High Time (Measured at 0.75 Vcc)	18.75	22.91	ns		4-3
t28a	Low Time (Measured at 0.35 Vcc)	18.75	22.91	ns		4-3
Suspend Clock (SUSCLK)						
f_{susclk}	Operating Frequency	32		kHz	4	
t39	High Time	9.5	—	µs	4	
t39a	Low Time	9.5	—	µs	4	

1. N/A.
2. The maximum high time (t18 Max) provide a simple ensured method for devices to detect bus idle conditions.
3. BCLK Rise and Fall times are measured from 10% VDD and 90% VDD.
4. SUSCLK duty cycle can range from 30% minimum to 70% maximum.
5. Edge rates in a system as measured from 0.8 – 2.0V.
6. The active frequency can be 5 MHz, 50 MHz, or 62.5 MHz depending on the interface speed. Dynamic changes of the normal operating frequency are not allowed.
7. Testing condition: 1 kΩ pull-up to Vcc, 1 kΩ pull-down and 10 pF pull-down and 1/2 inch trace.
8. Jitter is specified as cycle-to-cycle as measured between two rising edges of the clock being characterized. Period min and max includes cycle-to-cycle jitter and is also measured between two rising edges of the clock being characterized.
9. On all jitter measurements care should be taken to set the zero crossing voltage (for rising edge) of the clock to be the point where the edge rate is the fastest. Using a Math function = Average(Derivative(Ch1)) and set the averages to 64, place the cursors where the slope is the highest on the rising edge—usually this lower half of the rising edge. The reason this is defined is for users trying to measure in a system it is impossible to get the probe exactly at the end of the Transmission line with large Flip-Chip components. This results in a reflection induced ledge in the middle of the rising edge and will significantly increase measured jitter.
10. Phase jitter requirement: The designated outputs will meet the reference clock jitter requirements from the *PCI Express Base Specification*. The test is to be performed on a component test board under quiet conditions with all clock outputs on. Jitter analysis is performed using a standardized tool provided by the PCI SIG. Measurement methodology is defined in the Intel document *PCI Express Reference Clock Jitter Measurements.*
11. For LPC, slew rate measurements are done at 20% and 60%
12. Testing condition: 1 kW pull-up to Vcc, 1 kW pull-down and 10 pF pull-down and 1/2 inch trace (see Figure 6-54 for more detail).
13. Total of crystal cut accuracy, frequency variations due to temperature, parasitics, load capacitance variations, and aging is recommended to be less than 90 ppm.
14. Spread Spectrum (SSC) is referenced to rising edge of the clock.
15. When SMLink0 is configured to run in Fast Mode (FM) using soft strap, the supported operating range is 0 kHz ~ 400 kHz, but the typical operating frequency is in the range of 300 kHz – 400 kHz.
16. When SMLink0 is configured to run in Fast Mode Plus (FMP) using a soft strap, the supported operating range is 0 Hz ~ 1 MHz, but the typical operating frequency is in the range of 900 kHz – 1000 KHz. This is the default mode for this interface.
17. Test load: 10 pF to ground.Measured at 0.2 * Vcc - 0.8 * Vcc.
18. Spec is for jitter at pins of PCH. See section 4.4, PCIe Gen4 clock RMS jitter measurements.
19. Using 48Mhz Xtal that meets PDG specification of +/- 30ppm

Note: Refer to PCI Local Bus Specification for measurement details.

Figure 4-3. Clock Timing

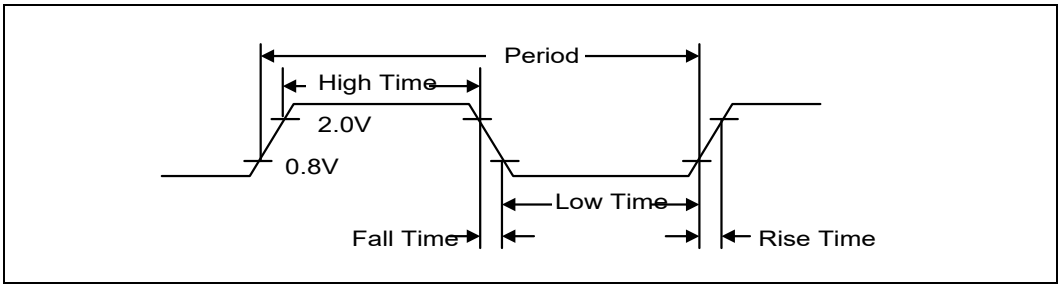
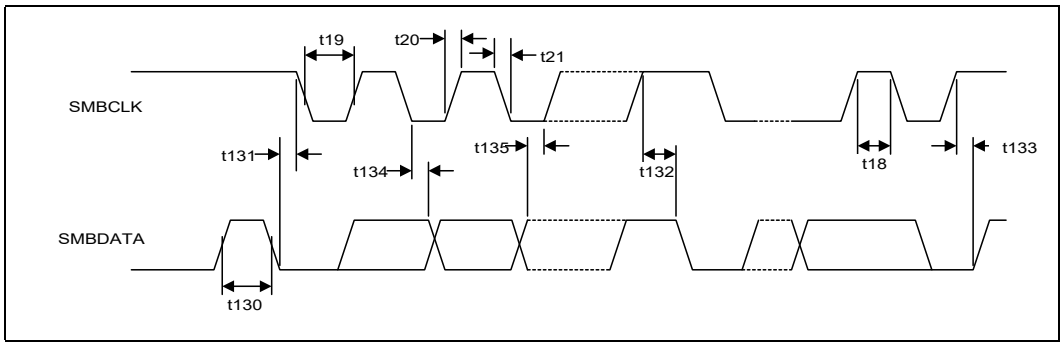


Figure 4-4. SMBus/SMLink Transaction



Note: The t*xx* also refers to t*xx*_SMLFM and txx_SMLFMP, t*xxx* also refers to t*xxx*SMLFM and txx_SMLFMP, SMBCLK also refers to SML[4:0]CLK, and SMBDATA also refers to SML[4:0]DATA.

Figure 4-5. PCH Test Load

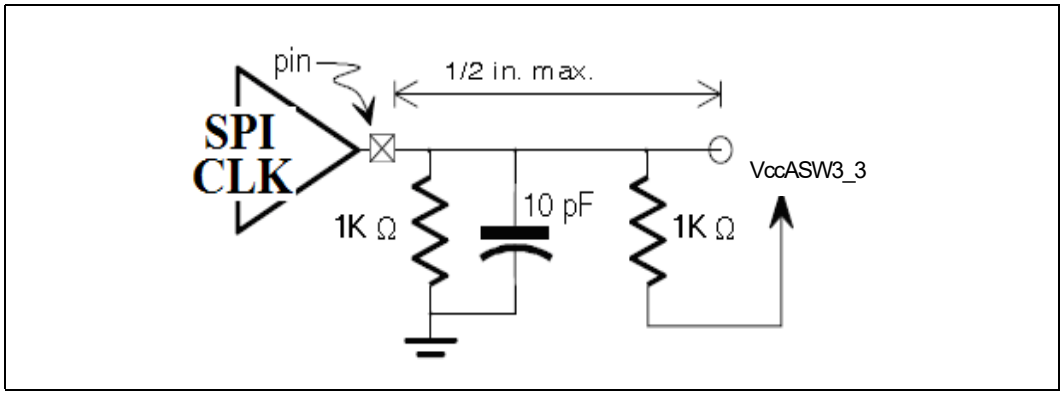


Table 4-16. USB 2.0 Timing (Sheet 1 of 2)

Sym	Parameter	Min	Max	Units	Notes	Figure
Full-Speed Source (Note 7)						
t100	USBPx+, USBPx- Driver Rise Time	4	20	ns	1,6 C_L = 50 pF	4-6
t101	USBPx+, USBPx- Driver Fall Time	4	20	ns	1,6 C_L = 50 pF	4-6
t102	Source Differential Driver Jitter - To Next Transition - For Paired Transitions	–3.5 –4	3.5 4	ns ns	2, 3	4-7
t103	Source SE0 interval of EOP	160	175	ns	4	4-8
t104	Source Jitter for Differential Transition to SE0 Transition	–2	5	ns	5	
t105	Receiver Data Jitter Tolerance - T o Next Transition - For Paired Transitions	–18.5 –9	18.5 9	ns ns	3	4-7
t106	EOP Width: Must accept as EOP	82	—	ns	4	4-8
t107	Width of SE0 interval during differential transition	—	14	ns		
Low-Speed Source (Note 8)						
t108	USBPx+, USBPx – Driver Rise Time	75	300	ns	1,6 C_L = 200 pF C_L = 600 pF	4-6

Table 4-16. USB 2.0 Timing (Sheet 2 of 2)

Sym	Parameter	Min	Max	Units	Notes	Figure
t109	USBPx+, USBPx – Driver Fall Time	75	300	ns	1,6 C_L = 200 pF C_L = 600 pF	4-6
t110	Source Differential Driver Jitter To Next Transition For Paired Transitions	 –25 –14	 25 14	 ns ns	2,3	4-7
t111	Source SE0 interval of EOP	1.25	1.50	µs	4	4-8
t112	Source Jitter for Differential Transition to SE0 Transition	–40	100	ns	5	
t113	Receiver Data Jitter Tolerance - To Next Transition - For Paired Transitions	 –152 –200	 152 200	 ns ns	3	4-7
t114	EOP Width: Must accept as EOP	670	—	ns	4	4-8
t115	Width of SE0 interval during differential transition	—	210	ns		

Notes:
1. Driver output resistance under steady state drive is specified at 28 Ω at minimum and 43 Ω at maximum.
2. Timing difference between the differential data signals.
3. Measured at crossover point of differential data signals.
4. Measured at 50% swing point of data signals.
5. Measured from last crossover point to 50% swing point of data line at leading edge of EOP.
6. Measured from 10% to 90% of the data signal.
7. Full-speed Data Rate has minimum of 11.97Mb/s and maximum of 12.03Mb/s.
8. Low-speed Data Rate has a minimum of 1.48Mb/s and a maximum of 1.52Mb/s.

Table 4-17. USB 3 Interface Transmit and Receiver Timings

Sym	Parameter	Min	Max	Units	Notes	Figure
UI	Unit Interval – USB 3.0 (5.0 GT/s)	199.9	200.1	ps		
T_{TX-EYE}	Minimum Transmission Eye Width	0.625	—	UI		

Figure 4-6. USB Rise and Fall Times

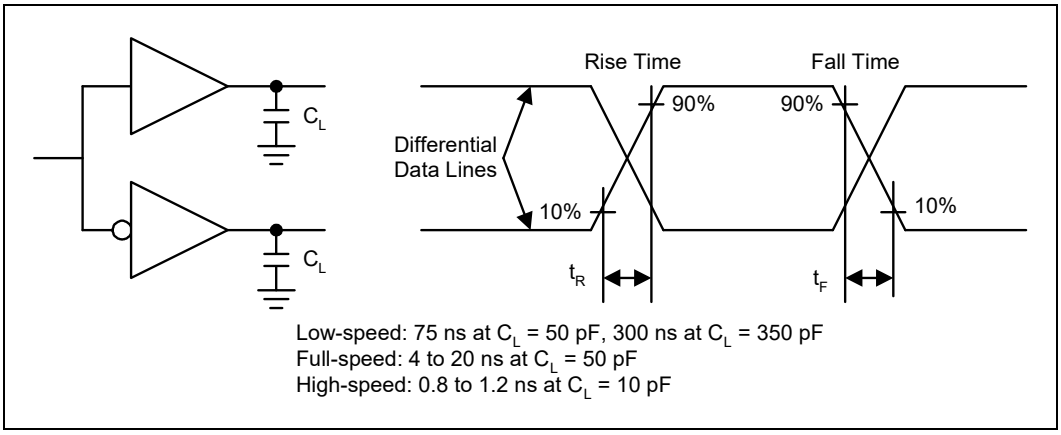


Figure 4-7. USB Jitter

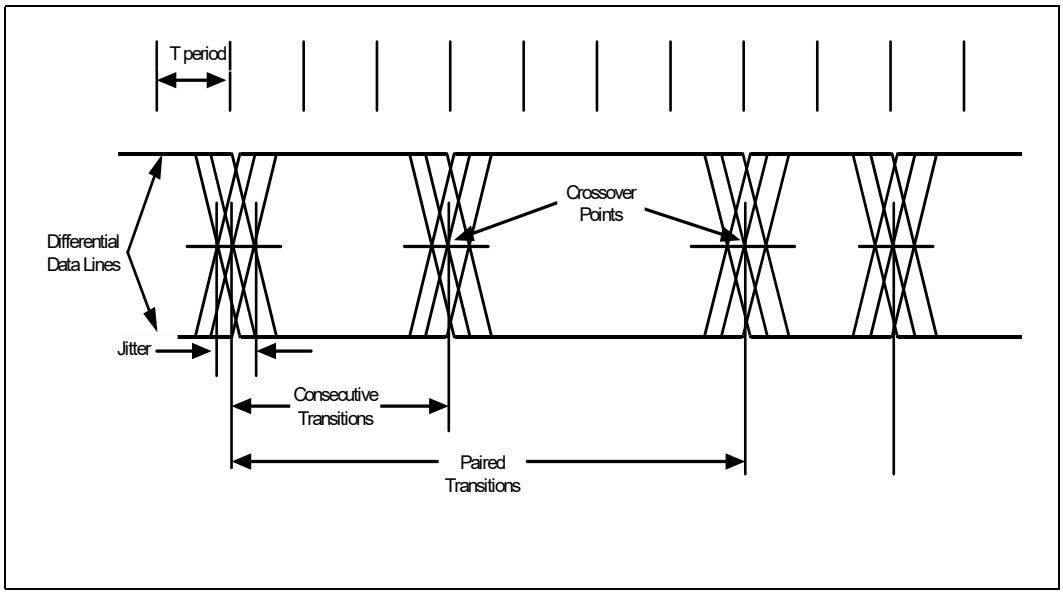


Figure 4-8. USB EOP Width

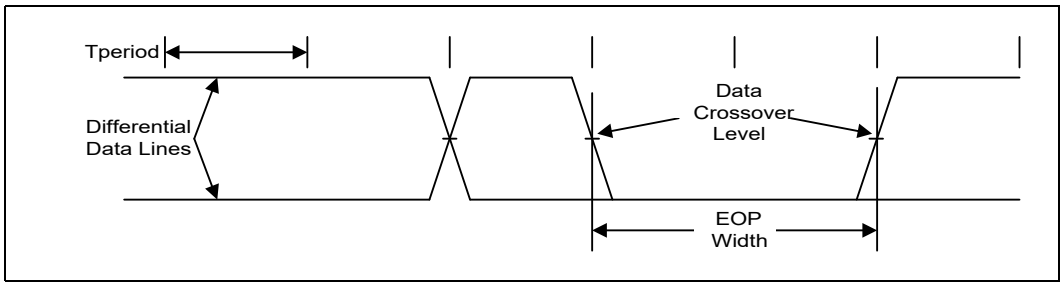


Table 4-18. SATA Interface Timings

Sym	Parameter	Min	Max	Units	Notes	Figure
UI-3	Gen III Operating Data Period (6Gb/s)	166.6083	166.6667	ps		
t120gen3	Rise Time	0.2	0.48	UI	1	
t121gen3	Fall Time	0.2	0.48	UI	2	
t122	TX differential skew	—	20	ps		
t123	COMRESET	304	336	ns	3	
t124	COMWAKE transmit spacing	101.3	112	ns	3	
t125	OOB Operating Data period	646.67	686.67	ns	4	

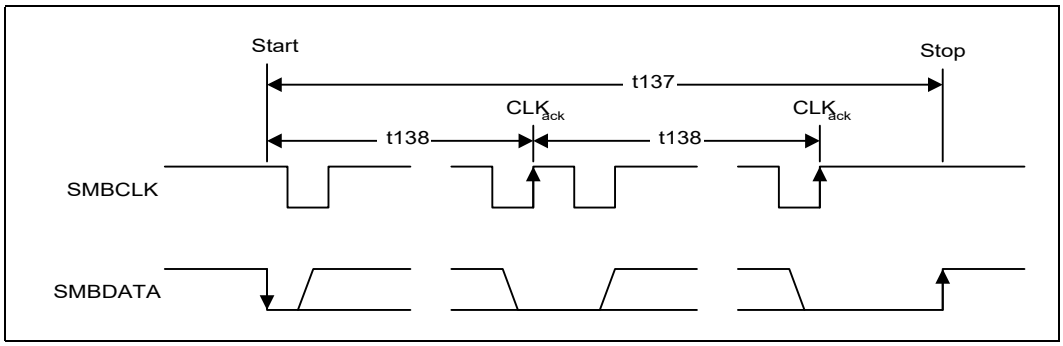
Notes:
1. 20 – 80% at transmitter
2. 80 – 20% at transmitter
3. As measured from 100mV differential crosspoints of last and first edges of burst.
4. Operating data period during out-of-band burst transmissions.

Table 4-19. SMBus and SMLink Timing

Sym	Parameter	Min	Max	Units	Notes	Figure
t130	Bus Free Time Between Stop and Start Condition	4.7	—	µs		4-5
t130SMLFM	Bus Free Time Between Stop and Start Condition	1.3	—	µs	5	4-5
t130SMLFMP	Bus Free Time Between Stop and Start Condition	0.5	—	µs	5	4-5
t131	Hold Time after (repeated) Start Condition. After this period, the first clock is generated.	4.0	—	µs		4-5
t131SMLFM	Hold Time after (repeated) Start Condition. After this period, the first clock is generated.	0.6	—	µs	5	4-5
t131SMLFMP	Hold Time after (repeated) Start Condition. After this period, the first clock is generated.	0.26	—	µs	5	4-5
t132	Repeated Start Condition Setup Time	4.7	—	µs		4-5
t132SMLFM	Repeated Start Condition Setup Time	0.6	—	µs	5	4-5
t132SMLFMP	Repeated Start Condition Setup Time	0.26	—	µs	5	4-5
t133	Stop Condition Setup Time	4.0	—	µs		4-5
t133SMLFM	Stop Condition Setup Time	0.6	—	µs	5	4-5
t133SMLFMP	Stop Condition Setup Time	0.26	—	µs	5	4-5
t134	Data Hold Time	300	—	ns	4	4-5
t134SMLFM	Data Hold Time	0	—	ns	4, 5	4-5
t134SMLFMP	Data Hold Time	0	—	ns	4, 5	4-5
t135	Data Setup Time	250	—	ns		4-5
t135SMLFM	Data Setup Time	100	—	ns	5	4-5
t135SMLFMP	Data Setup Time	50	—	ns	5	4-5
t136	Device Time Out	25	35	ms	1	
t137	Cumulative Clock Low Extend Time (slave device)	—	25	ms	2	4-9
t138	Cumulative Clock Low Extend Time (master device)	—	10	ms	3	4-9
T_{por}	Time in which a device must be operational after power-on reset	—	500	ms		

Notes:
1. A device will timeout when any clock low exceeds this value.
2. t137 is the cumulative time a slave device is allowed to extend the clock cycles in one message from the initial start to stop. If a slave device exceeds this time, it is expected to release both its clock and data lines and reset itself.
3. t138 is the cumulative time a master device is allowed to extend its clock cycles within each byte of a message as defined from start-to-ack, ack-to-ack, or ack-to-stop.
4. t134 has a minimum timing for I^2C of 0 ns, while the minimum timing for SMBus/SMLINK is 300 ns.
5. Timings with the SMLFM designator apply only to SMLink0 and only when SMLink0 is operating in Fast Mode. SMLFP is forFast Mode Plus.

Figure 4-9. SMBus/SMLink Timeout



Note: SMBCLK also refers to SML[1:0]CLK and SMBDATA also refers to SML[1:0]DATA in Figure 4-4.

Table 4-20. Intel® High Definition Audio (Intel® HD Audio) Timing

Sym	Parameter	Min	Max	Units	Notes	Figure
t143	Time duration for which HDA_SDO is valid before HDA_BCLK edge.	7	—	ns		4-10
t144	Time duration for which HDA_SDO is valid after HDA_BCLK edge.	7	—	ns		4-10
t145	Setup time for HDA_SDI[1:0] at rising edge of HDA_BCLK	15	—	ns		4-10
t146	Hold time for HDA_SDI[1:0] at rising edge of HDA_BCLK	0	—	ns		4-10

Figure 4-10. Intel® High Definition Audio (Intel® HD Audio) Input and Output Timings

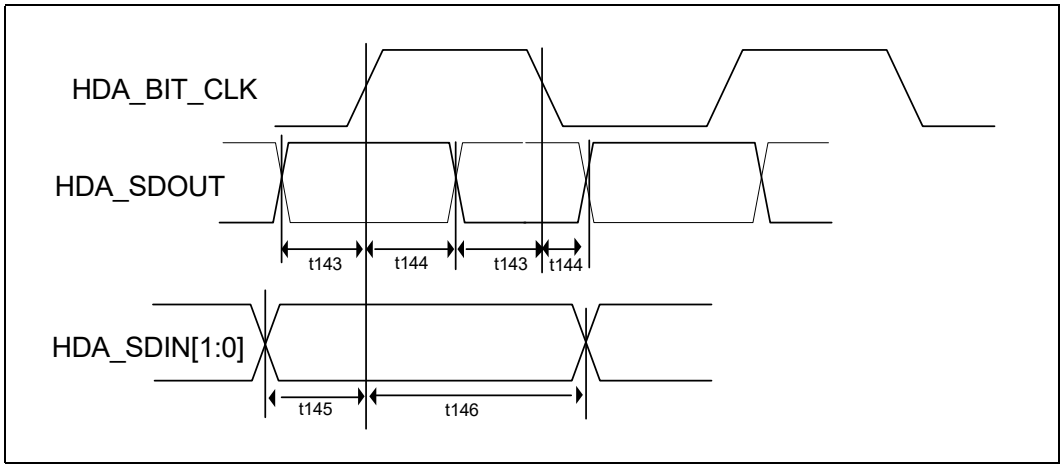


Table 4-21. LPC Timing (Sheet 1 of 2)

Sym	Parameter	Min	Max	Units	Notes	Figure
t150	LAD[3:0] Valid Delay from CLKOUT_LPC[2:0] Rising	2	12.5	ns		4-11
t151	LAD[3:0] Output Enable Delay from CLKOUT_LPC[2:0] Rising	2	—	ns		4-14
t152	LAD[3:0] Float Delay from CLKOUT_LPC[2:0] Rising	—	28	ns		4-13
t153	LAD[3:0] Setup Time to CLKOUT_LPC[2:0] Rising	5	—	ns		4-12

Table 4-21. LPC Timing (Sheet 2 of 2)

Sym	Parameter	Min	Max	Units	Notes	Figure
t154	LAD[3:0] Hold Time from CLKOUT_LPC[2:0] Rising	2.5	—	ns		4-12
t157	LFRAME# Valid Delay from CLKOUT_LPC[2:0] Rising	3	24.67	ns		4-11

Figure 4-11. Valid Delay from Rising Clock Edge

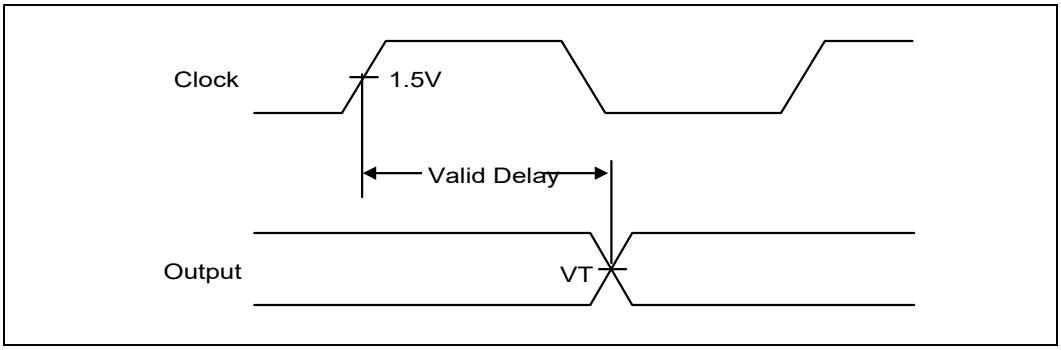


Figure 4-12. Setup and Hold Times

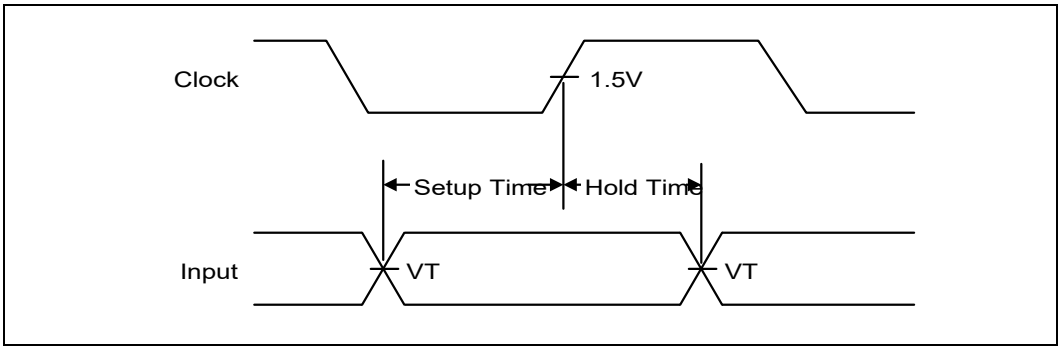


Figure 4-13. Float Delay

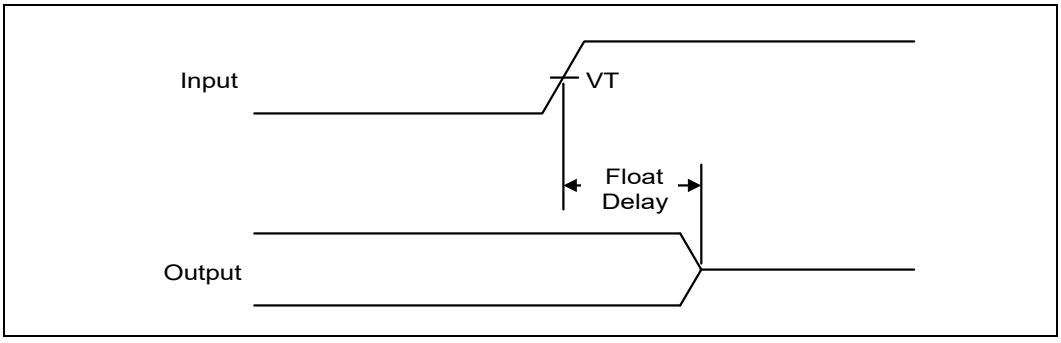


Figure 4-14. Output Enable Delay

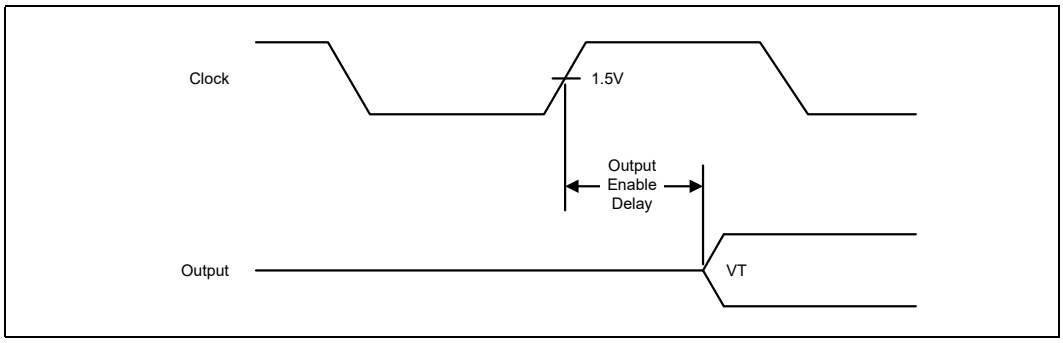


Table 4-22. Miscellaneous Timings

Sym	Parameter	Min	Max	Units	Notes	Figure
t160	SERIRQ Setup Time to PCICLK Rising	7	—	ns		4-16
t161	SERIRQ Hold Time from PCICLK Rising	0	—	ns		
t162	GPIO, USB Resume Pulse Width	2	—	RTCCLK		4-17
t163	SPKR Valid Delay from OSC Rising	—	200	ns		4-15

Figure 4-15. Valid Delay from Rising Clock Edge

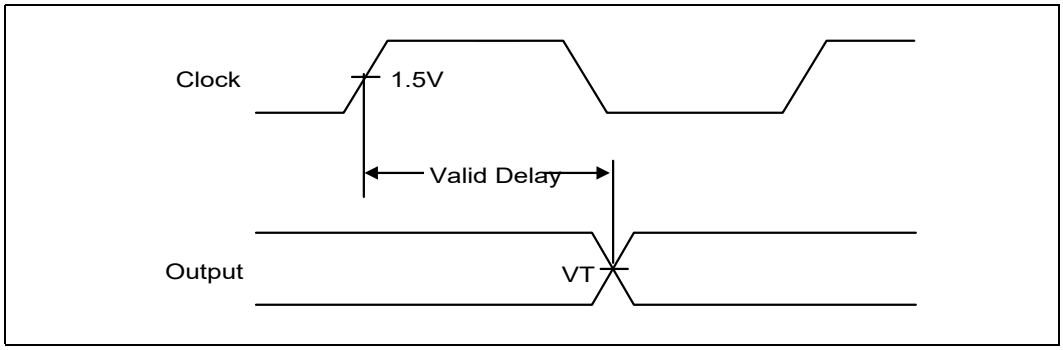


Figure 4-16. Setup and Hold Times

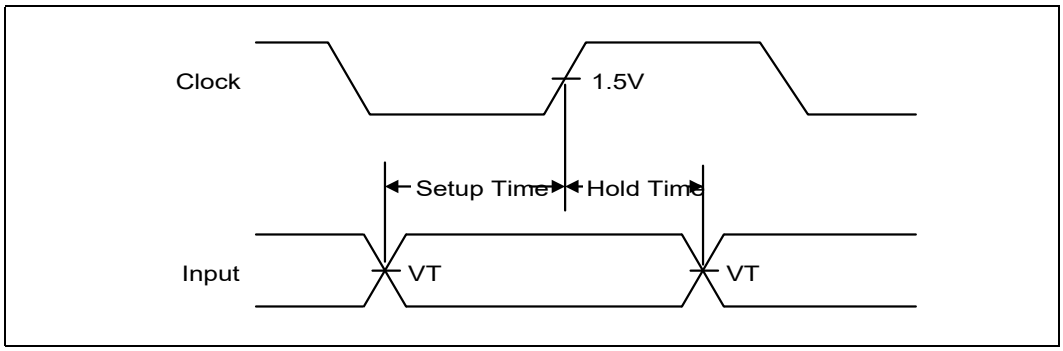


Figure 4-17. Pulse Width

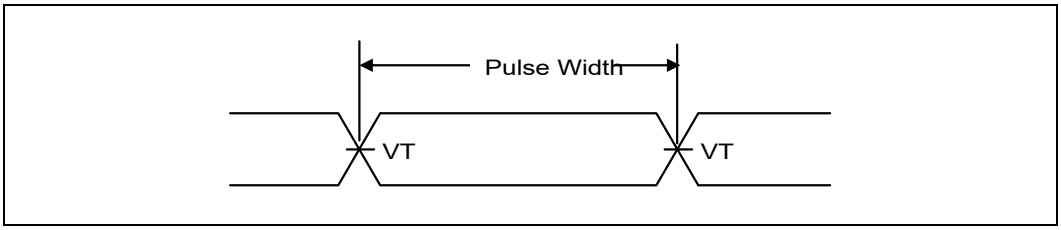


Table 4-23. SPI Timings (17 MHz)

Sym	Parameter	Min	Max	Units	Notes	Figure
t180a	Serial Clock Frequency - 17MHz operation	16	18	MHz		
t183a	Tco of SPI_MOSI and SPI_IO[3:2] with respect to serial clock falling edge at the host	-9	+13.5	ns	1	4-18
t184a	Setup of SPI_MISO and SPI_IO[3:2] with respect to serial clock falling edge at the host	5	—	ns	1	4-18
t185a	Hold of SPI[1:0]_MISO and SPI[1:0]_IO[3:2] with respect to serial clock falling edge at the host	2.5	—	ns	1	4-18
t186a	Setup of SPI0_FLASH_CS[1:0]#, SPI_TPM_CS# assertion with respect to serial clock rising edge at the host	30	—	ns	1	4-18
t187a	Hold of SPI0_FLASH_CS[1:0]#, SPI1_TPM_CS# assertion with respect to serial clock falling edge at the host	30	—	ns	1	4-18
t188a	SPI[1:0]_CLK High time	26	33	ns	1	4-18
t189a	SPI[1:0]_CLK Low time	26	33	ns	1	4-18
	SPI[1:0] CLK rise slew rate	0.4		V/ns	2	
	SPI[1:0] CLK fall slew rate	0.4		V/ns	2	

Notes:
1. Measurement point for low time and high time is taken at 0.5(VccSPI).
2. Cap load is 45pF. Timing is from 10% to 90%.

Table 4-24. SPI Timings (30 MHz) (Sheet 1 of 2)

Sym	Parameter	Min	Max	Units	Notes	Figure
t180b	Serial Clock Frequency - 30 MHz operation	29	31	MHz		
t183b	Tco of SPI[1:0]_MOSI and SPI[1:0]_IO[3:2] with respect to serial clock falling edge at the host	-5	7.5	ns	1	4-18
t184b	Setup of SPI[1:0]_MISO and SPI[1:0]_IO[3:2] with respect to serial clock falling edge at the host	5	—	ns	1	4-18
t185b	Hold of SPI[1:0]_MISO and SPI[1:0]_IO[3:2] with respect to serial clock falling edge at the host	2.5	—	ns	1	4-18
t186b	Setup of SPI0_FLASH_CS[1:0]#, SPI_TPM _CS# assertion with respect to serial clock rising edge at the host	30	—	ns	1	4-18
t187b	Hold of SPI0_FLASH_CS[1:0]#, SPI1_TPM_CS# assertion with respect to serial clock falling edge at the host	30	—	ns	1	4-18
t188b	SPI[1:0]_CLK High time	14	20	ns	1	4-18
t189b	SPI[1:0]_CLK Low time	14	20	ns	1	4-18
	SPI[1:0] CLK rise slew rate	0.4		V/ns	2	

Table 4-24. SPI Timings (30 MHz) (Sheet 2 of 2)

Sym	Parameter	Min	Max	Units	Notes	Figure
	SPI[1:0] CLK fall slew rate	0.4		V/ns	2	

Note:
1. Measurement point for low time and high time is taken at 0.5(VccSPI).
2. Cap load is 45pF. Timing is from 10% to 90%.

Table 4-25. SPI Timings (40 MHz)

Sym	Parameter	Min	Max	Units	Notes	Figure
t180c	Serial Clock Frequency - 40MHz operation	39	41	MHz		
t183c	Tco of SPI[1:0]_MOSI and SPI[1:0]_IO[3:2] with respect to serial clock falling edge at the host	-3.5	5	ns	1,2	4-18
t184c	Setup of SPI[1:0]_MISO and SPI[1:0]_IO[3:2] with respect to serial clock falling edge at the host	5	—	ns	1,2	4-18
t185c	Hold of SPI[1:0]_MISO and SPI[1:0]_IO[3:2] with respect to serial clock falling edge at the host	2.5	—	ns	1,2	4-18
t186c	Setup of SPI0_CS[1:0]#, SPI1_CS# assertion with respect to serial clock rising edge at the host	30	—	ns	1,2	4-18
t187c	Hold of SPI0_CS[1:0]#, SPI1_CS# assertion with respect to serial clock falling edge at the host	30	—	ns	1,2	4-18
t188c	SPI[1:0]_CLK High time	9	16	ns	1,2	4-18
t189c	SPI[1:0]_CLK Low time	9	16	ns	1,2	4-18
	SPI[1:0] CLK rise slew rate	0.4		V/ns	3	
	SPI[1:0] CLK fall slew rate	0.4		V/ns	3	

Note:
1.When using 40 MHz mode ensure target flash component can meet t188c and t189c specifications. Measurement should be taken at a point as close as possible to the package pin.
2. Measurement point for low time and high time is taken at 0.5(VccSPI).
3. Cap load is 45pF. Timing is from 10% to 90%.

Figure 4-18. SPI Timings

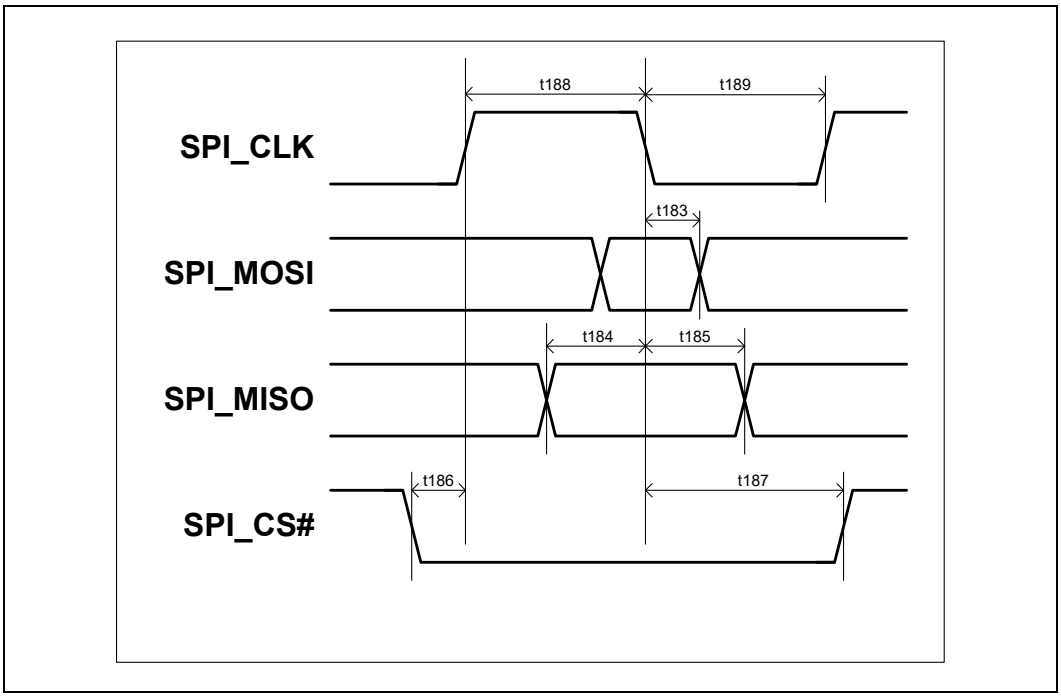


Figure 4-19. eSPI Clock Timings

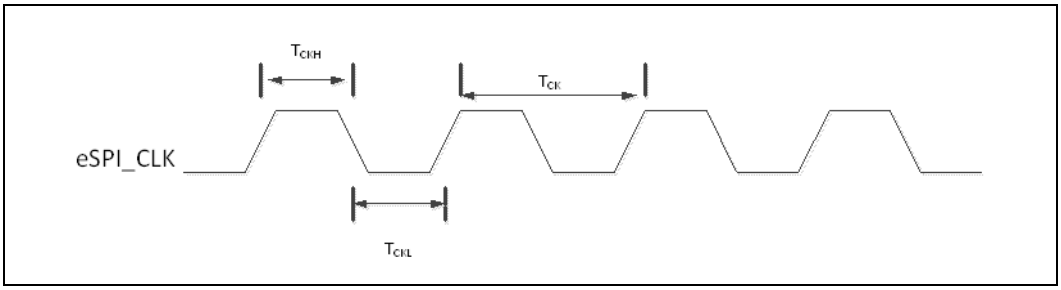


Figure 4-20. eSPI Command timings

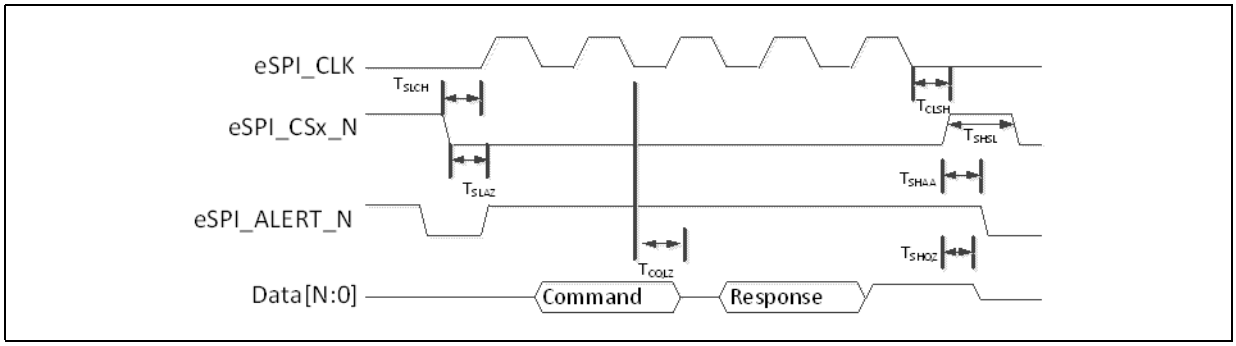


Figure 4-21. eSPI DATA Timings

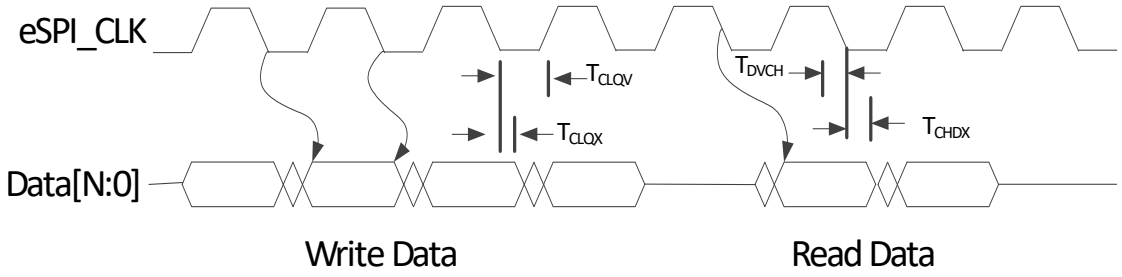


Table 4-26. eSPI Timing

Symbol	Description	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Units
	Frequency	20 Nom		24 Nom		30 Nom		48 Nom		60 Nom		MHz
T_{CK}	Clock Period	48.2	51.7	40.2	43.1	32.2	34.5	20.1	21.6	16.0	17.2	ns
T_{CKH}	Clock High Time	0.4 Tck		0.4 Tck		0.4 Tck		0.34 Tck		0.4 Tck		Tck
T_{CKL}	Clock Low Time	0.4 Tck		0.4 Tck		0.4 Tck		0.4 Tck		0.4T ck		Tck
T_{SLCH}	CS# Setup Tim	75		60		45		30		22		ns
T_{CLSH}	CS# Hold time	50		40		30		20		15		ns
T_{SHSL}	CS# deassertion	50		40		30		20		15		ns
T_{DVCH}	Data in Setup	3.5		3.5		3.5		3.5		3.5		ns
T_{CHDX}	Data in hold	2.5		2.5		2.5		2.5		2.5		ns
T_{CLQZ}	Output disable during turnout		15		12		9		8		6	
T_{CLQV}	Data out valid		3.5		3.5		3.5		3.5		3.5	ns
T_{CLQX}	Data out hold	-2		-2		-2		-2		-2		ns
T_{SHQZ}	Output disable after CS# deassertion		15		12		9		8		6	ns

Notes: Measurement point for all measurements is taken at 0.5 (VccpGPPA).
Read data is with relationship to the PCH, when the setup and hold of the data at the PCH pins, and when the PCH reads in the data.

Figure 4-22. PM_SYNC Timings

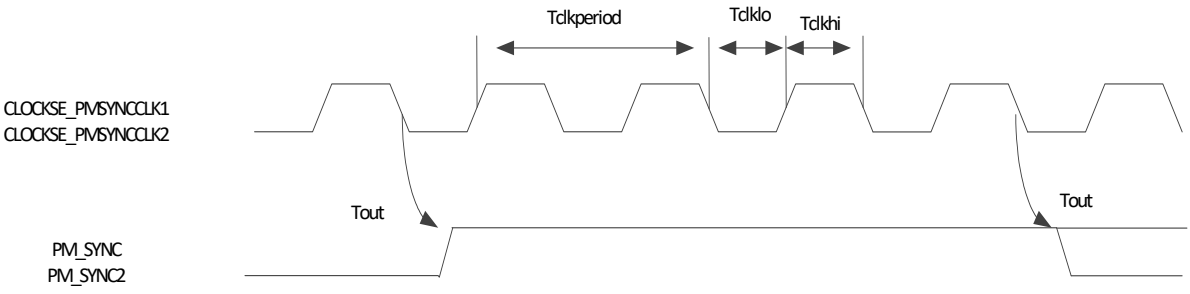


Table 4-27. PM_SYNC Timing

Symbol	Description	Min	Max	Units
	Frequency	24	24	MHz
$T_{CKperiod}$	Clock Period	41.6		ns
T_{Clkhi}	Clock High Time	18	22	ns
T_{Clklo}	Clock Low Time	10	22	ns
T_{out}	Clock out delay	2	10	ns
Note: Measurement point for all measurements is taken at 0.5 Vccprim_1P05.				

Figure 4-23. NCSI Timings

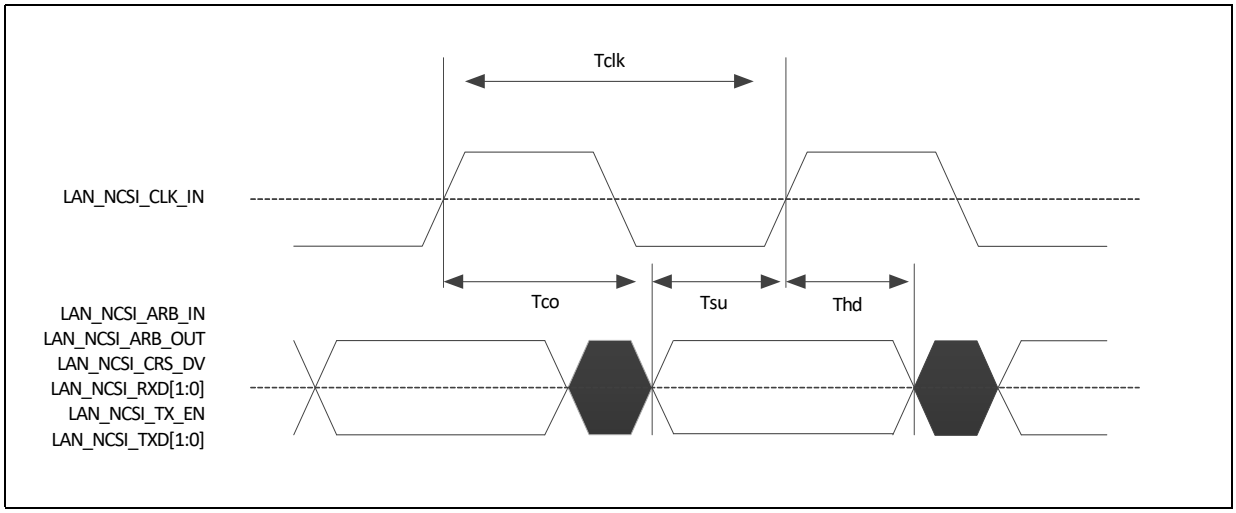


Table 4-28. NCSI Timing

Symbol	Description	Min.	Typ.	Max.	Units
	Frequency	50	50	50	MHz
T_{Clk}	Clock Period		20		ns
T_{Co}	Data out delay	4		9	ns
T_{Su}	data setup to clock	2			ns
T_{Hd}	data hold	2			ns
Note: Measurement point for all measurements is taken at 0.5 x Vccppgk.					

4.4 PCIe* 4.0 Clock RMS Jitter Measurement

Testing for clock jitter on the PCIe source clocks has become more difficult due to the nature of the extremely small jitter that is allowed. The jitter spec. listed in Table 4-15, "Intel® C620 Series Chipset Clock Timings" is the jitter at the pins of the PCH. That can prove to be difficult, so often times the jitter measurement is taken at the destination of the clocks.

Adding to the difficulty in measuring jitter with 0.5 ps or less are the inherent errors in the equipment taking the measurements. If using a digital scope, Intel has observed that the equipment can add more than 0.1 ps of jitter to the measurement, which is a very large percentage of the 0.5 ps spec. if not accounted for.

4.5 Overshoot/Undershoot Guidelines

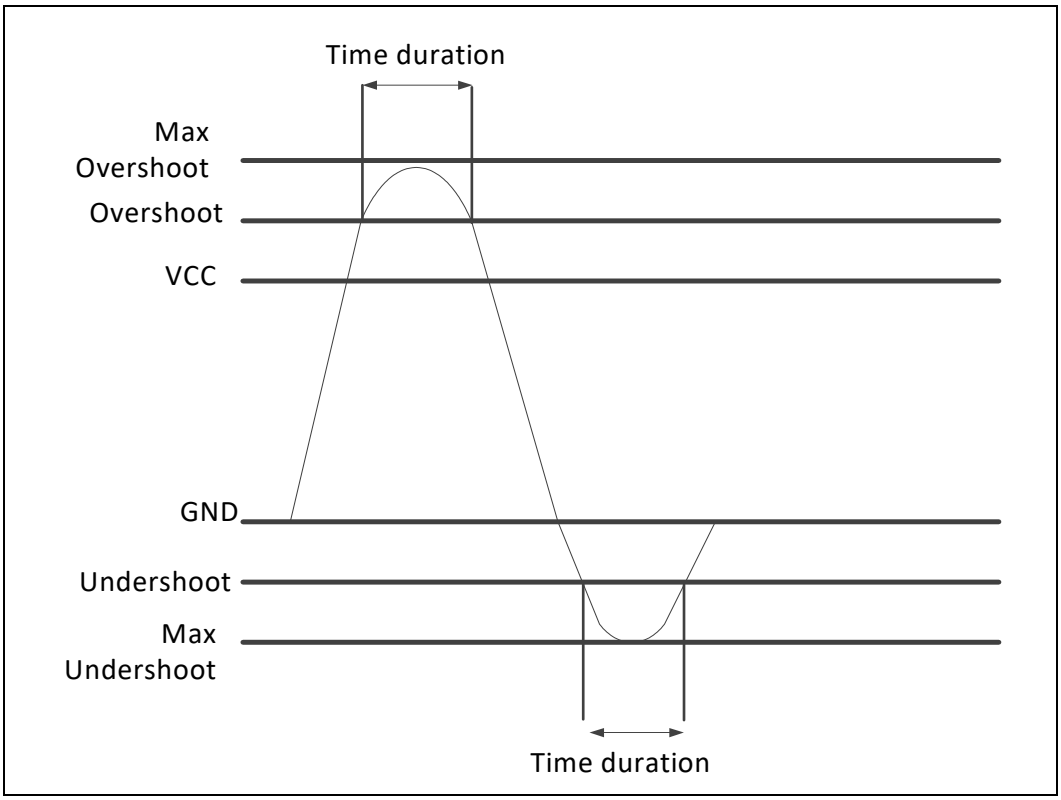
Overshoot (or undershoot) is the absolute value of the maximum voltage above VCC or below VSS. The PCH can be damaged by single and/or repeated overshoot or undershoot events on any input, output, or I/O buffer if the charge is large enough. Baseboard designs that meet signal integrity and timing requirements and that do not exceed the maximum overshoot or undershoot limits listed in the following table ensures reliable I/O performance for the lifetime of the PCH.

Table 4-29. Overshoot/Undershoot Specifications

I/O	Max Overshoot	Time Duration	Overshoot	Time Duration	Under Shoot	Time Duration	Max Under Shoot	Time Duration
3.3V I/O	4.515V	0 ns	3.75V	5 ns	-0.3V	5 ns	-1.13V	0 ns
1.8V I/O	2.48	0 ns	2.16	2.5 ns	-0.3V	2.5 ns	-0.62	0 ns
1.05V	1.48V	0 ns	1.25V	2.5 ns	-0.25V	2.5 ns	-0.48V	0 ns

Notes:
1. These specifications are measured at the PCH pin.

Figure 4-24. Maximum Acceptable Overshoot/Undershoot Waveform

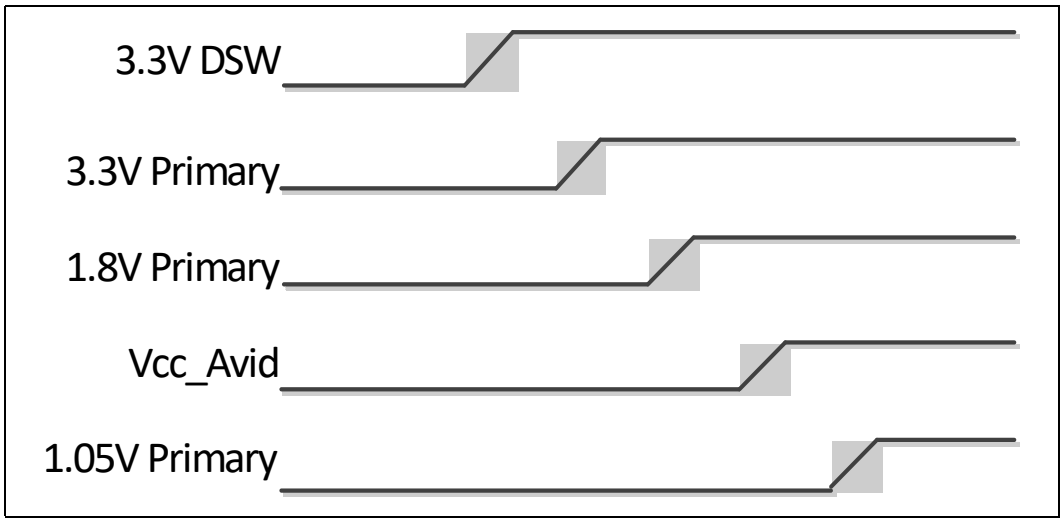


4.6 Single Ended I/O Initial Power Up

The Intel® C620 Series Chipset Single Ended I/Os (SEIO) can be powered by multiple power rails. One set can be powered by either 3.3V primary or 1.8V primary. One set can only be powered by 3.3V Deep Sleep Well (DSW), and the final group is powered by 1.05V primary. When the power rails come up, it is possible to have pulses on the outputs of these SEIOs with the exception of the 1.05V power SEIOs as described below.

The order the power wells ramp up is shown in the following figure. If the deep sleep well operation is not supported, then the 3.3V DSW and 3.3V primary wells will be tied together.

Figure 4-25. Intel® C620 Series Chipset Power Supply Ramp Order



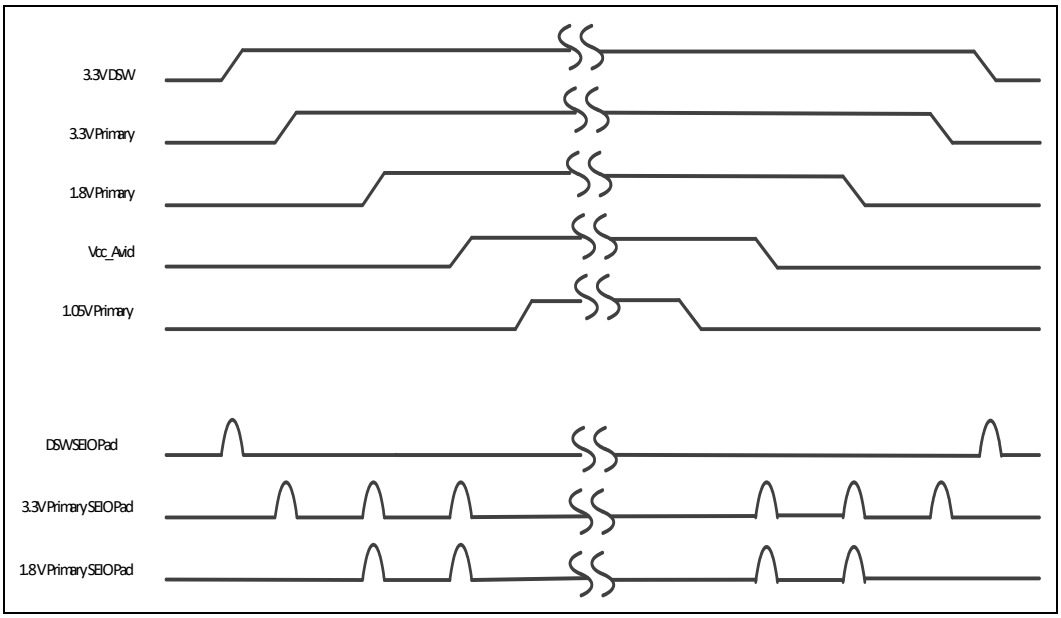
As the 3.3V DSW, 3.3V primary, 1.8V primary, and Vcc_AVID voltages power up, the SEIOs, whose I/O buffers are powered by these voltages, can see pulses on those pins as the power rails ramp. Those SEIO whose output voltage is powered by 1.05V primary (GPP_L pins, P_RDY, etc.) will not have any glitches on them as all other voltages are already up and running by the time the 1.05V I/O rail powers up.

The diagram below shows how the power up looks for those pins with pull downs on them by default (which is the vast majority). The 3.3V DSW powered SEIOs will see a pulse at the ramp of 3.3V DSW, but no further pulses will occur during the power up. At power down, when 3.3V DSW goes low, there can be a pulse on the outputs at that time, too.

The SEIOs powered, whose I/O are powered by 3.3V, can see three distinct pulses. At 3.3V ramp, there will be a pulse on the I/O. After the pulse, the SEIO is defined to be in the 'deterministic' phase, where a 20K pull-down is engaged. When 1.8V ramps, there will be a second pulse on the output, before returning to the deterministic phase. Finally, when Vcc_AVID ramps, there is a final pulse before the I/O moves to it's default initial state. At that time, most SEIOs are GPI, which would results in the pins being non-driven and thus the state of the pad will be determined by the external connection. See chapter 16 for the initial state of the GPIOS.

GPIOs powered by 1.8 will have the same pulses during the ramp of 1.8V and Vcc_AVID, but no pulses when 3.3V powers up as the I/O is unpowered at that time. The deterministic state for these GPIOS

Figure 4-26. SEIO Pulses on Power Up



Pulse duration can vary based upon the ramp time of the voltages, and individual components. The numbers given below are assuming worst case with a voltage ramp of 0.5V/msec or faster. It is possible that no pulses could be observed at all in some case. For the maximum voltage of the pulse could be as high as the voltage rail powering the I/O (3.3V or 1.8V).

3.3V pulse width up to 15 msec

1.8V pulse width up to 0.5 msec

Vcc_AVID pulse width up to 100 usec

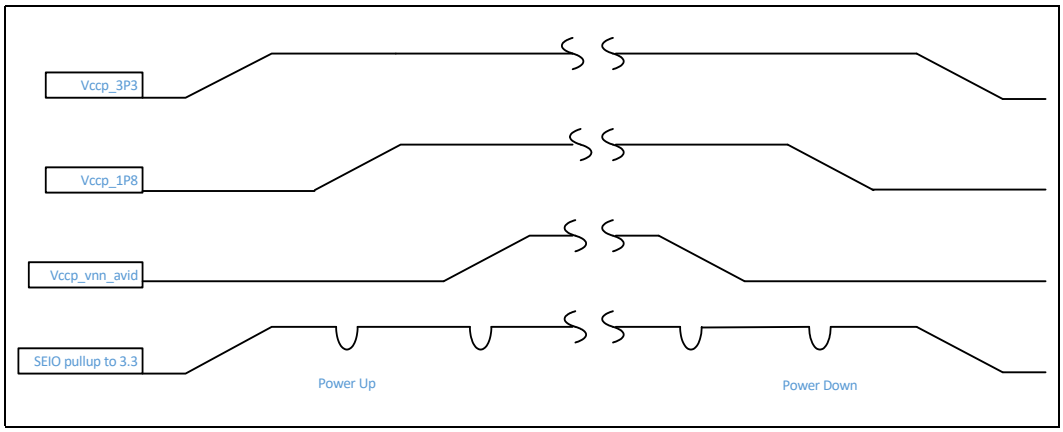
Two SEIO pins have pullups on them by default .. GPP_B0_VID0 and GPP_B1_VID1. The reason these pins have pullups on them is that they supply the Voltage ID (VID) for the power supply that provides Vcc_AVID. By default, all C620 series chipsets can operate at 1V (VID1,VID0 = 11b), so the default is to have VID0 and VID1 = 11b during initial power up. These pins are powered by 3.3V on the I/O. The same kind of glitches can occur on these pins, but instead of low going high, these are high going low glitches. The timings for these glitches are the same as the low-to-high glitches, and it’s possible for the high-to-low glitches to reach ground.

The pulse widths are specified to be the same going low as going high.

1.8V pulse width up to 0.5 msec

Vcc_AVID pulse width up to 100 usec

Figure 4-27. Power-up Glitches on GPIOs with Pull-up Resistors



4.7 Power Sequencing and Reset Signal Timings

Figure 4-28. Power Ramp with DSX Support

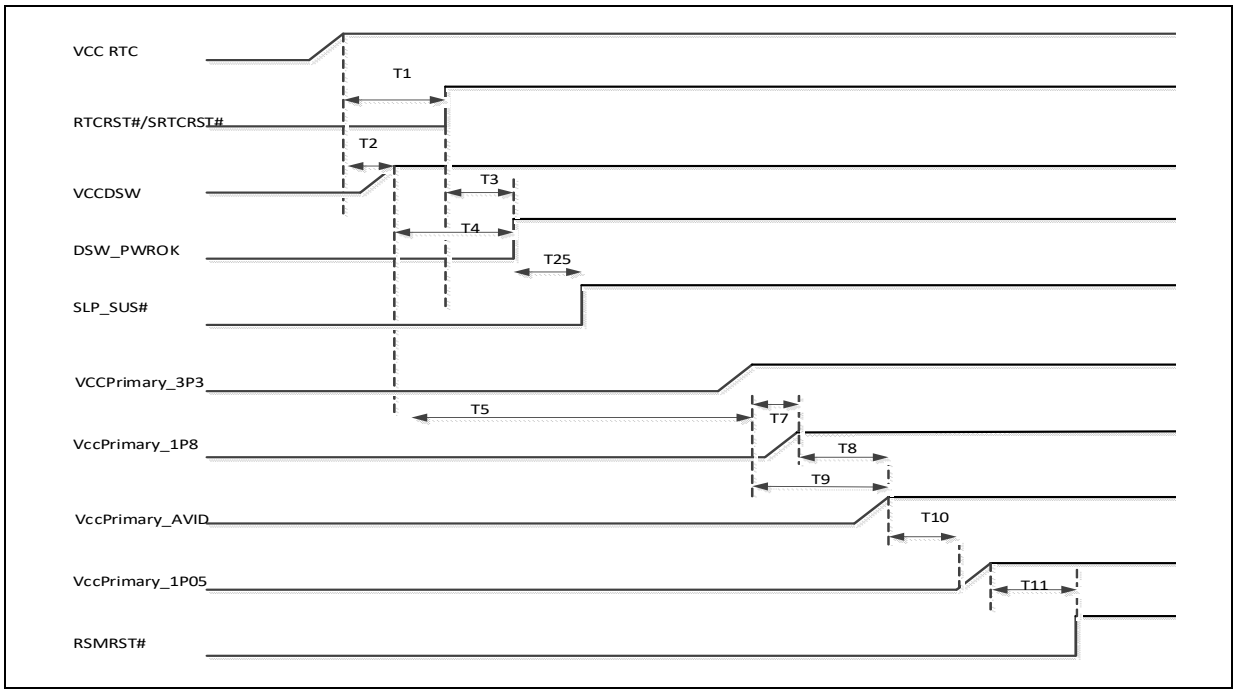


Table 4-30. Power Ramp with DSX Support (Sheet 1 of 2)

Symbol	Description	Min	Max	Unit	Comments
T1	VccRTC = 2.0V to the point in time where the voltage on the RTC resets equal 0.65 times the voltage present on the VccRTC rail during ramp. This measurement should be made from when VccRTC = 2.0V to the first of RTCRST# or SRTCRST# reaching 0.65*VccRTC.	9		ms	

Table 4-30. Power Ramp with DSX Support (Sheet 2 of 2)

Symbol	Description	Min	Max	Unit	Comments
T2	VccRTC at 95% to VccDSW at 95%	9		ms	
T3	RTCRST# to DSW_PWROK	1		ms	
T4	VccDSW at 95% to DSW_PWROK	10		ms	
T5	VccDSW at 95% to VccPRIMARY at 95%	0		ms	No lead needed between VccDSW and 1st primary power rail coming up.
T7	VccPrimary_3P3 to VccPrimary_1P8 at 95%		No limit		
T8	VccPrimary_1P8 at 95% to VccPrimary_AVID at 95%	0		ms	Preference to lead AVID
T9	VccPrimary_3P3 at 95% to VccPrimary_AVID at 95%	0		ms	Preference to lead AVID
T10	VccPrimary_AVID at 95% to Vccprimary_1P05 at 95%	0			VccPrimary_AVID is not to lag VccPrimary_1P05.
T11	VccPrimary_1P05 to RSMRST# high	10		ms	
T25	DSW_PWROK to SLP_SUS#		100	ms	

Note: T25 shows the maximum time for the case on initial boot when RTC power is first applied. It is possible to add up to 4 secs on this if BIOS programs SLP stretching. If the BIOS does not, then any G3 to S5 will always have the same 100 ms maximum time. If the BIOS does add SLP stretching, then T25 will be the new stretched value (100 ms + stretch time) going down into G3 and then back up until RTCRST# is brought low to clear the RTC registers and go back to the default timing.

Figure 4-29. Power Ramp Without DSX Support

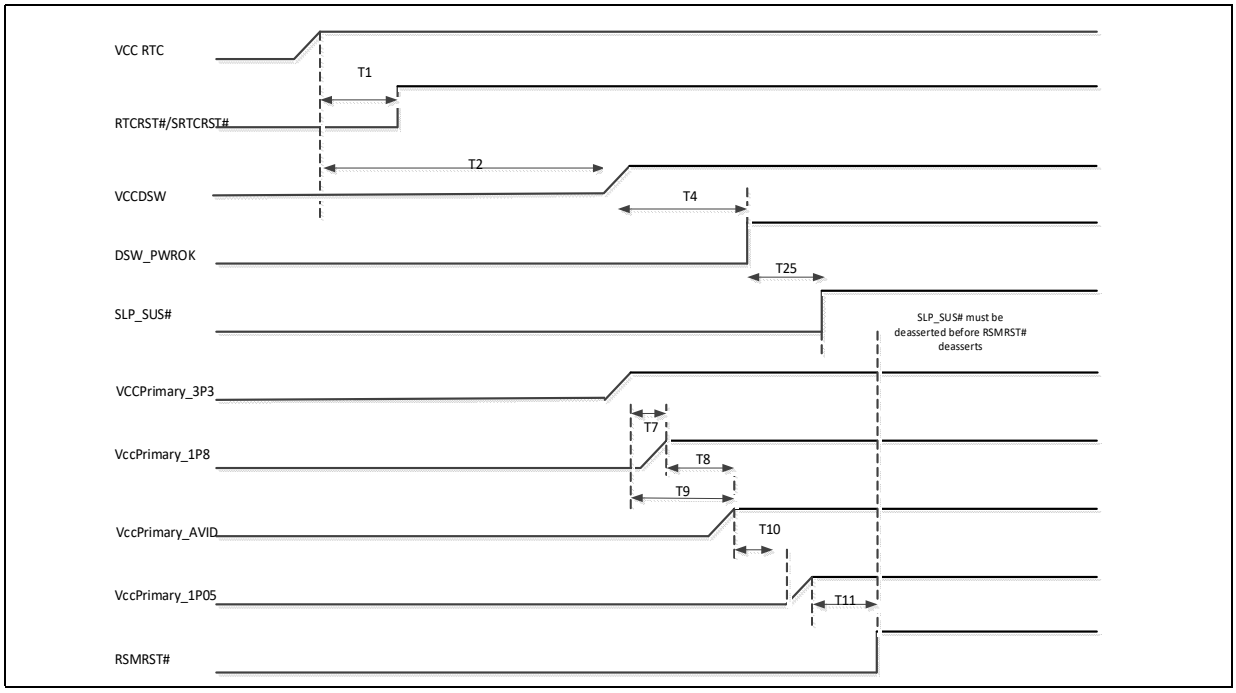


Table 4-31. Power Ramp Without DSX Support

Symbol	Description	Min	Max	Unit	Comments
T1	VccRTC = 2.0V to the point in time where the voltage on the RTC resets equal 0.65 times the voltage present on the VccRTC rail during ramp. This measurement should be made from when VccRTC = 2.0V to the first of RTCRST# or SRTCRST# reaching 0.65*VccRTC.	9		ms	
T2	VccRTC at 95% to VccDSW at 95%	9		ms	
	RTCRST# to DSW_PWROK	0		ms	
T4	VccDSW at 95% to DSW_PWROK	10		ms	
T7	VccPrimary_3P3 to VccPrimary_1P8 at 95%		No limit		
T8	VccPrimary_1P8 at 95% to VccPrimary_AVID at 95%	0		ms	Preference to lead AVID
T9	VccPrimary_3P3 at 95% to VccPrimary_AVID at 95%	0		ms	Preference to lead AVID
T10	VccPrimary_AVID at 95% to Vccprimary_1P05 at 95%	0			VccPrimary_AVID is not to lag VccPrimary_1P05
T11	VccPrimary_1P05 to RSMRST# high	10		ms	
T25	DSW_PWROK to SLP_SUS#		100	ms	

Note: T25 shows the maximum time for the case on initial boot when RTC power is first applied. It is possible to add up to 4 secs on this if BIOS programs SLP stretching. If the BIOS does not, then any G3 to S5 will always have the same 100 ms max time. If the BIOS does add SLP stretching, then T25 will be the new stretched value (100 ms + stretch time) going down into G3 and then back up until RTCRST# is brought low to clear the RTC registers and go back to the default timing.

Figure 4-30. SLP_S5# High to PLTRST# High

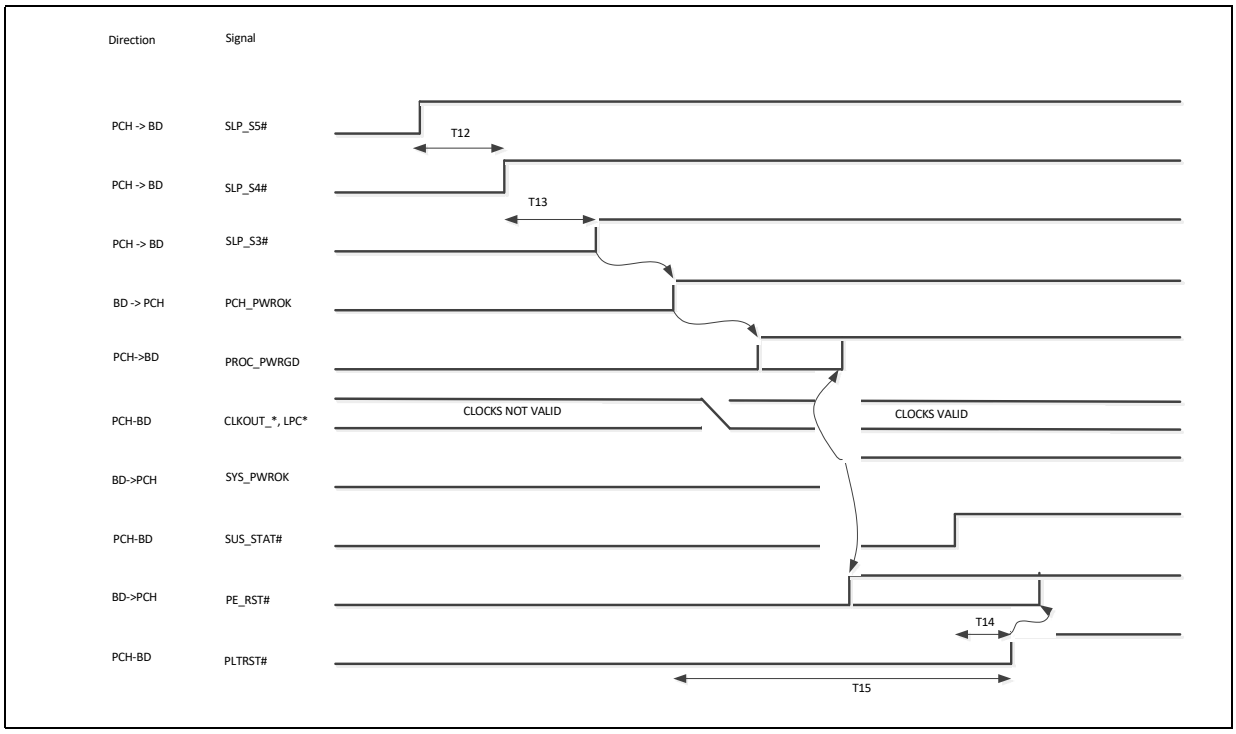


Table 4-32. SLP_S5# High to PLTRST# High

Symbol	Description	Min	Max	Unit	Comments
T12	SLP_S5# high to SLP_S4#	30		us	
T13	SLP_S4# high to SLP_S3#	30		us	
T15	PCH_PWROK# high to PLTRST# high	99		ms	
Note:	PROC_PWRGD may wait until SYS_PWROK goes high if selected by soft strap. See the appropriate Platform Design Guide to the platform requirements here.				
Note:	Board can drive PE_RST# high either with PROC_PWRGD or with PLTRST#. See the appropriate Platform Design Guide to the platform requirements here.				

Figure 4-31. S0 to S5

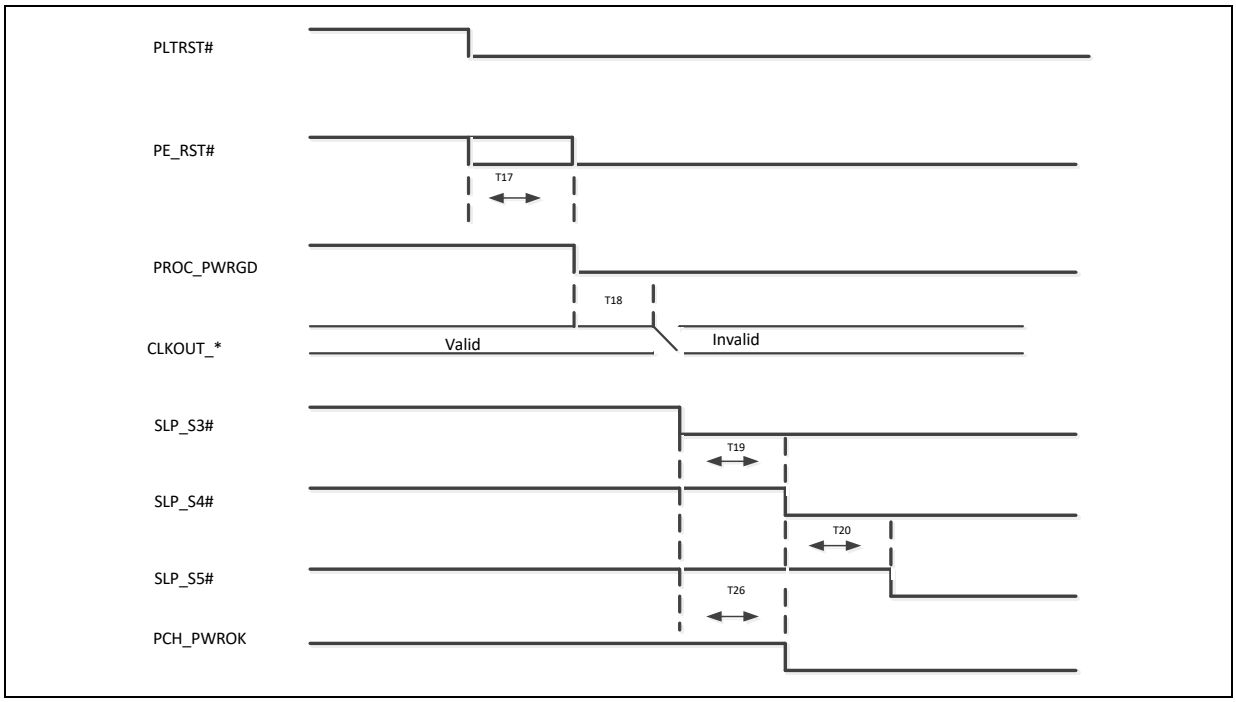


Table 4-33. S0 to S5

Symbol	Description	Min	Max	Unit	Comments
T17	PLTRST# low to PROC_PWRGD low	30		us	
T18	PROC_PWRGD to Clock invalid	10		us	Programmable. See Note below
T19	SLP_S3# to SLP_S4#	30		us	
T20	SLP_S4# to _SLP_S5#	30		us	
T26	SLP_S3# to PCH_PWROK low	0		ns	

Notes: PROC_PWRGD may wait until SYS_PWROK goes high if selected by soft strap. See the appropriate Platform Design Guide to the platform requirements here.
Board can drive PE_RST# high either with PROC_PWRGD or with PLTRST#. See the appropriate Platform Design Guide to the platform requirements here.
PCH_PWROK can go low before or after SLP_S4# and SLP_S5#.
Default for T18 is 10 usec, which is the minimum time. When the PCH is used in Endpoint Only (EPO) mode, the EPO PCH requires that its 100Mhz differential reference input clock remain active for 10ms after PE_RST# goes active on the EPO PCH. If the timing requirement is not met and power isn't subsequently removed from the EPO PCH, a global reset will occur on the EPO PCH. If the boot PCH on the host platform is an Intel® C620 Series Chipset PCH that is using internal or platform hybrid clocking mode, this timing requirement can be met by setting the host boot PCH timing T18 to 10 ms. This is programmed via the host platform boot CPH power management configuration register 1 (PM_CFG) at offset 18h.

Figure 4-32. Powerdown to DeepSX

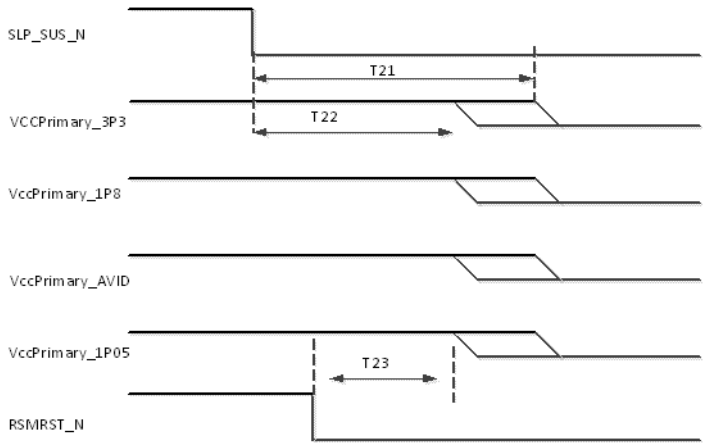


Table 4-34. Powerdown to DeepSX

Symbol	Description	Min	Max	Unit	Comments
T21	SLP_SUS_N low to All Primary rails low		100	ms	
T22	SLP_SUS_N dropping to Primary rails dropping	100		ns	
T23	RSMRST_N dropping to Primary rails dropping	1		us	
Note: There is no relationship that the power rails have to have with each other when going low.					

Figure 4-33. Surprise Powerdown to G3

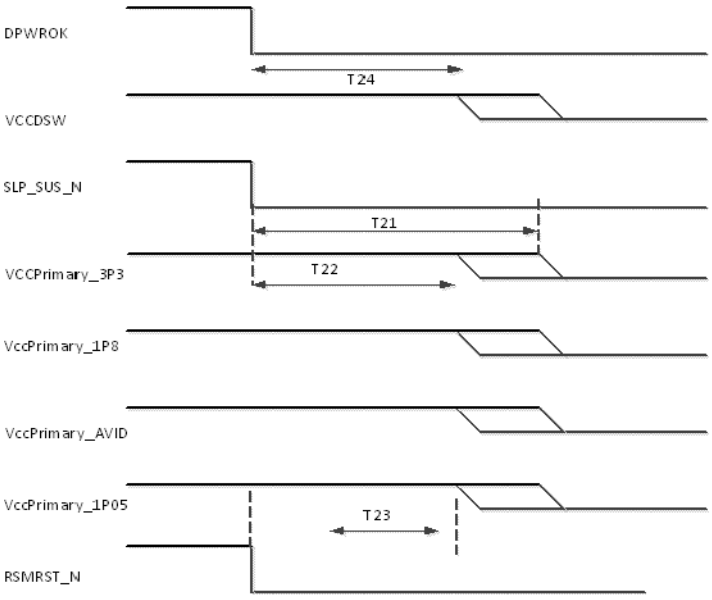


Table 4-35. Surprise Powerdown to G3

Symbol	Description	Min	Max	Unit	Comments
T21	SLP_SUS_N low to All Primary rails low		100	ms	
T22	SLP_SUS_N dropping to Primary rails dropping	100		ns	
T23	RSMRST_N dropping to Primary rails dropping	1		us	
T24	DSW_PWROK dropping before VCCDSW and the VccPRIMARY rail	1		us	See notes
Notes: There is no relationship that the power rails have to have with each other when going low. DSW_PWROK deassertion causes the PCH to internally reset and isolate all shallower wells. As long as T24 is met, T22 and T23 can be violated.					

Figure 4-34. eSPI Reset

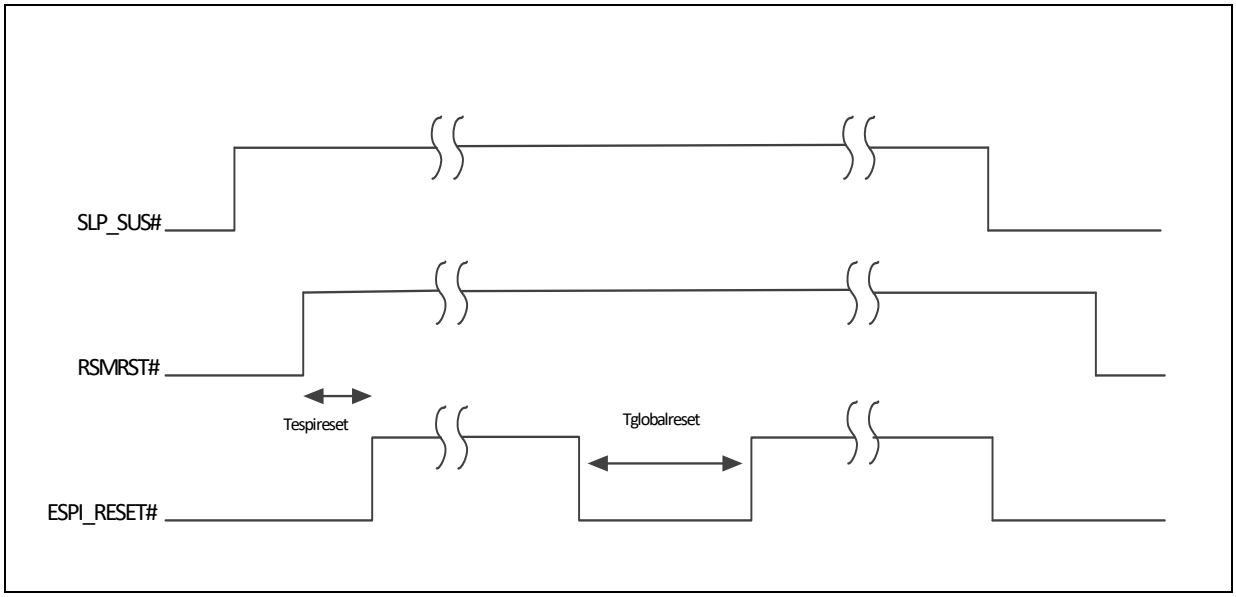


Table 4-36. eSPI Reset from G3

Symbol	Description	Min	Max	Unit	Comments
Tespireset	RSMRST# high to ESPI_RESET# high	800		usec	See note
Tgloblareset	ESPI_RESET# low during a global reset	100		usec	
Note: The Intel® C620 Series Chipset requirements are that RSMRST# goes high after SLP_SUS#, so there is no difference between leaving G3 or DEEP_SX.					

§

5 Power and Ground Signals

This section describes the power rails and ground signals on the PCH.

Name	Description
VCCA_CLKFILT_1P05	Clock filtered supply
VCCA_CLKUNF_1p05	Unfiltered clock supply
VCCA_CLKXTAL_1P05	Filtered supply for 48 MHz main crystal
VCCA_PLL0_1P05	Filtered supply to PLL
VCCA_PLL1_1P05	Filtered supply to PLL
VCCMPHY_1p05	Supply for PCIe, DMI, USB3 and SATA PHYs
VCCP_1P8	1.8 volt supply
VCCP_3p3	Use for fuses, HVCMOS and Group I primary power well
VCCP_HSIOPLL_1P05	Supply for PCIe, SATA PLL
VCCP_HSIOPLLUNF_1P05	Unfiltered supply for PCIe, SATA PLLs
VCCP_UPPLL_1P05	Supply for PCIe Uplink and DMI PLL
VCCP_UPPLLUNF_1P05	Unfiltered supply for PCIe Uplink and DMI PLL
VCCP10GBE_HV_1p8	1.8V supply for 10GbE
VCCP10GBE_LV_1p05	Main power for 10GbE LAN
VCCP10GBEPLL_1P05	Power for 10GbE LAN PLL
VCCPDSW_3p3	Deep Sleep well 3.3V supply
VCCPGPP_1P8	1.8 volt supply for GPIO groups A/B/C/D/E/F/G H/J/K
VCCPGPPA	Switchable 1.8/3.3 volt power well for GPPA
VCCPGPPB	Switchable 1.8/3.3 volt power well for GPPB
VCCPGPPC	Switchable 1.8/3.3 volt power well for GPPC
VCCPGPPD	Switchable 1.8/3.3 volt power well for GPPD
VCCPGPPE	Switchable 1.8/3.3 volt power well for GPPE
VCCPGPPF	Switchable 1.8/3.3 volt power well for GPPF
VCCPGPPG	Switchable 1.8/3.3 volt power well for GPPG
VCCPGPPH	Switchable 1.8/3.3 volt power well for GPPH
VCCPGPPJ	Switchable 1.8/3.3 volt power well for GPPJ
VCCPGPPK	Switchable 1.8/3.3 volt power well for GPPK
VCCPHDA_1p8	1.8V supply for the High Def Audio
VCCPRIM_1P05	Primary power, 1.05V
VCCPRIM_Avid	Primary power, AVID rail, varies from 0.85 to 1.0V
VCCPRTC	**RTC Well Supply.** This rail can drop to 2.0V if all other planes are off. This power is not expected to be shut off unless the RTC battery is removed or drained. VCCRTC nominal voltage is 3.0V. This rail is intended to always come up first and always stay on. It should NOT be power cycled regularly on non-coin battery designs.
VCCPRTCPRIM_3p3	Power to RTC well logic. This will be off in Deep Sleep mode.
VCCPSPI_1P8	1.8 volts to SPI periphery
VCCPUSBDSW_3P3	Deep sleep well 3.3V supply for I/O

Name	Description
VCCRTCEXT	RTC de-coupling capacitor only. This rail should NOT be driven.
VCCPSPI	Power for SPI periphery. Switchable between 3.3 and 1.8V.
VSS	Ground
CGC_DUT_DETECT_N	Package orientation detect

Note: CGC_DUT_DETECT_N, while connected to ground on the package, is not used as a ground return pin. This pin is used for helping in insuring that the Intel C620 chip is correctly oriented. If this pad location is pulled up to some reference voltage, when the PCH is correctly oriented on the package this signal will be pulled to ground. If the orientation is off, then there will not be a ball on the package at this location to connect with the socket, resulting in this signal being high.

§

6 Flexible I/O

6.1 Acronyms

Acronyms	Description
HSIO	High Speed I/O lanes
SSATA	Server Serial Advanced Technology Transport
SATA	Serial Advanced Technology Transport
PCIe	PCI Express
USB	Universal Serial Bus
GbE	Gigabit Ethernet

6.2 Overview

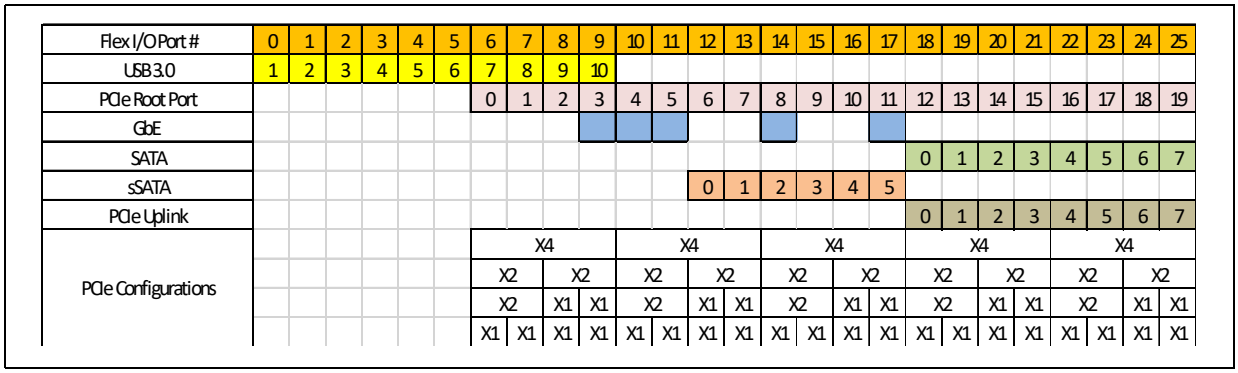
Flexible I/O is an architecture that allows some high speed signals to be statically configured as PCIe* root ports, USB 3.0, GbE, SATA, SSATA and PCIe uplink signals per I/O needs on a platform. Note that the GbE here is the single 1 GbE port that is legacy from past PCHs. It is not related to the new 10/1 GbE functions/ports added to the Intel® C620 Series Chipset PCH.

6.3 Description

The PCH implements a number of HSIO lanes that are split between the different interface: PCIe, USB 3.0, SATA, SSATA and GbE. The picture below summarizes the PCH HSIO lane muxing.

The Flexible I/O is configured through soft straps. Refer to the latest *PCH SPI Programming Guide* for more details on the soft straps.

Figure 6-1. HSIO Muxing on the Intel® C620 Series Chipset



There are 26 HSIO lanes on the PCH, supporting the following port configurations:

1. Up to 20 PCIe root port lanes (multiplexed with USB 3.0, SATA, SSATA ports, and PCIe uplink)
   — Only a maximum of 16 PCIe ports (or devices) can be enabled at any time.
   — Ports 0-3, Ports 4-7, Ports 8-11, Ports 12-15, and Ports 16-19, each can be individually configured as 4x1, 2x2, 1x2 + 2x1, or 1x4.
2. Up to 8 SATA ports
3. Up to 10 USB 3.0 ports (multiplexed with PCIe)
4. 1GbE lane
   — GbE can be mapped into one of the PCIe ports 3-5, port 8, and port 11.
   — When GbE is enabled, there can be at most up to 15 PCIe ports enabled.
5. Up to 6 SSATA ports
6. 8 PCIe uplink lanes for connecting with the CPU. This can either be done by directly connecting to the CPU or wrapping around to the PCIe root ports. One limitation of the connection to the PCIe root ports is that the maximum uplink width can not be more than 4 lanes.

§

7 Memory Mapping

7.1 Overview

This section describes (from the CPU perspective) the memory ranges that the PCH decodes.

7.2 Functional Description

7.2.1 PCI Devices and Functions

The PCH incorporates a variety of PCI devices and functions, as shown in Table 7-1. If for some reason, the particular system platform does not want to support any one of the Device Functions, with the exception of D30:F0, can individually be disabled. The integrated gigabit Ethernet controller will be disabled if no platform LAN connect component is detected (See Chapter 19, "Gigabit Ethernet Controller"). When a function is disabled, it does not appear at all to the software. A disabled function will not respond to any register reads or writes, insuring that these devices appear hidden to software.

Table 7-1. PCI Devices and Functions (Sheet 1 of 2)

Device:Functions #	Function Description
Bus 0: Device 31: Function 0	LPC Interface (eSPI Enable Strap = 0) eSPI Interface (eSPI Enable Strap = 1)
Bus 0: Device 31: Function 3	Intel® HD Audio (Audio, Voice, Speech)
Bus 0: Device 31: Function 4	SMBus Controller
Bus 0: Device 31: Function 5	SPI
Bus 0: Device 31: Function 6	GbE Controller
Bus 0: Device 29: Function 0	PCI Express Port 8
Bus 0: Device 29: Function 1	PCI Express Port 9
Bus 0: Device 29: Function 2	PCI Express Port 10
Bus 0: Device 29: Function 3	PCI Express Port 11
Bus 0: Device 29: Function 4	PCI Express Port 12
Bus 0: Device 29: Function 5	PCI Express Port 13
Bus 0: Device 29: Function 6	PCI Express Port 14
Bus 0: Device 29: Function 7	PCI Express Port 15
Bus 0: Device 28: Function 0	PCI Express Port 0
Bus 0: Device 28: Function 1	PCI Express Port 1
Bus 0: Device 28: Function 2	PCI Express Port 2
Bus 0: Device 28: Function 3	PCI Express Port 3
Bus 0: Device 28: Function 4	PCI Express Port 4
Bus 0: Device 28: Function 5	PCI Express Port 5
Bus 0: Device 28: Function 6	PCI Express Port 6
Bus 0: Device 28: Function 7	PCI Express Port 7
Bus 0: Device 27: Function 0	PCI Express Port 16

Table 7-1. PCI Devices and Functions (Sheet 2 of 2)

Device:Functions #	Function Description
Bus 0: Device 27: Function 1	PCI Express Port 17
Bus 0: Device 27: Function 2	PCI Express Port 18
Bus 0: Device 27: Function 3	PCI Express Port 19
Bus 0: Device 23: Function 0	SATA Controller
Bus 0: Device 22: Function 0	Intel ME HECI #1
Bus 0: Device 22: Function 1	Intel ME HECI #2
Bus 0: Device 22: Function 2	Intel ME IDE-Redirection (IDE-R)
Bus 0: Device 22: Function 3	Intel ME Keyboard and Text (KT) Redirection
Bus 0: Device 22: Function 4	Intel ME HECI #3
Bus 0: Device 20: Function 0	USB 3.0 xHCI Controller
Bus 0: Device 20: Function 2	Thermal Subsystem
Bus 0: Device 17: Function 0	MROM 0 Function
Bus 0: Device 17: Function 1	MROM 1 Function
Bus 0: Device 17: Function 5	SSATA Controller
Bus 0: Device 16: Function 0	IE HECI #1
Bus 0: Device 16: Function 1	IE HECI #2
Bus 0: Device 16: Function 2	IE IDE-Redirection (IDE-R)
Bus 0: Device 16: Function 3	IE Keyboard and Text (KT) Redirection
Bus 0: Device 16: Function 4	IE HECI #3
Bus N+2: Device 0:Function 0	Intel QuickAssist Technology 0
Bus N+3: Device 0:Function 0	Intel QuickAssist Technology 1
Bus N+4: Device 0:Function 0	Intel QuickAssist Technology 2 if routed up NPx16
Bus M+2: Device 0: Function 0	Intel QuickAssist Technology 2 if routed up NPx8
Bus N+5; Device 0: Function [15:0]	10/1 GbE if NPx8 is not enabled
Bus N+4; Device 0: Function [15:0]	10/1 GbE if NPx8 is enabled
Note: When a device or function is disabled, it is not reported to the software and will not respond to any register reads or writes.	

7.2.2 Fixed I/O Address Ranges

Table 7-2, "Fixed I/O Ranges Decoded by PCH" shows the Fixed I/O decode ranges from the CPU perspective. Note that for each I/O range, there may be separate behavior for reads and writes. DMI cycles that go to target ranges that are marked as Reserved will be handled by the PCH; writes are ignored and reads will return all 1's.

Address ranges that are not listed or marked RESERVED are NOT positively decoded by the PCH (unless assigned to one of the variable ranges) and will be internally terminated by the PCH.

Table 7-2. Fixed I/O Ranges Decoded by PCH (Sheet 1 of 3)

I/O Address	Read Target	Write Target	Internal Unit
20h – 21h	Interrupt Controller	Interrupt Controller	Interrupt
24h – 25h	Interrupt Controller	Interrupt Controller	Interrupt
28h – 29h	Interrupt Controller	Interrupt Controller	Interrupt
2Ch – 2Dh	Interrupt Controller	Interrupt Controller	Interrupt

Table 7-2. Fixed I/O Ranges Decoded by PCH (Sheet 2 of 3)

I/O Address	Read Target	Write Target	Internal Unit
2E - 2F	LPC/eSPI	LPC/eSPI	Forwarded to LPC/eSPI
30h – 31h	Interrupt Controller	Interrupt Controller	Interrupt
34h – 35h	Interrupt Controller	Interrupt Controller	Interrupt
38h – 39h	Interrupt Controller	Interrupt Controller	Interrupt
3Ch – 3Dh	Interrupt Controller	Interrupt Controller	Interrupt
40h	Timer/Counter	Timer/Counter	8254 Timer
42h-43h	Timer/Counter	Timer/Counter	8254 Timer
4E-4F	LPC/eSPI	LPC/eSPI	Forwarded to LPC/eSPI
50h	Timer/Counter	Timer/Counter	8254 Timer
52h-53h	Timer/Counter	Timer/Counter	8254 Timer
60h	LPC/eSPI	LPC/eSPI	Forwarded to LPC/eSPI
61h	NMI Controller	NMI Controller	CPU I/F
62h	Microcontroller	Microcontroller	Forwarded to LPC/eSPI
63h	NMI Controller1	NMI Controller1	CPU I/F
64h	Microcontroller	Microcontroller	Forwarded to LPC/eSPI
65h	NMI Controller1	NMI Controller1	CPU I/F
66h	Microcontroller	Microcontroller	Forwarded to LPC/eSPI
67h	NMI Controller[1]	NMI Controller[1]	CPU I/F
70h	RTC Controller	NMI and RTC Controller	RTC
71h	RTC Controller	RTC Controller	RTC
72h	RTC Controller	RTC Controller	RTC
73h	RTC Controller	RTC Controller	RTC
74h	RTC Controller	RTC Controller	RTC
75h	RTC Controller	RTC Controller	RTC
76h-77h	RTC Controller	RTC Controller	RTC
80h	LPC/eSPI or PCIe	LPC/eSPI or PCIe	LPC/eSPI or PCIe
84h - 86h	Reserved	LPC/eSPI or PCIe	LPC/eSPI or PCIe
88h	Reserved	LPC/eSPI or PCIe	LPC/eSPI or PCIe
8Ch - 8Eh	Reserved	LPC/eSPI or PCIe	LPC/eSPI or PCIe
90h	(alias to 80h)	(alias to 80h)	Forwarded to LPC/eSPI
92h	Reset Generator	Reset Generator	CPU I/F
94h - 96h	(Aliases to 8xh)	(Aliases to 8xh)	Forwarded to LPC/eSPI
98h	(Alias to 88h)	(Alias to 88h)	Forwarded to LPC/eSPI
9Ch - 9Eh	(Alias to 8xh)	(Aliases to 8xh)	Forwarded to LPC/eSPI
A0h - A1h	Interrupt Controller	Interrupt Controller	Interrupt
A4h - A5h	Interrupt Controller	Interrupt Controller	Interrupt
A8h - A9h	Interrupt Controller	Interrupt Controller	Interrupt
ACh - ADh	Interrupt Controller	Interrupt Controller	Interrupt
B0h - B1h	Interrupt Controller	Interrupt Controller	Interrupt
B2h - B3h	Power Management	Power Management	Power Management
B4h - B5h	Interrupt Controller	Interrupt Controller	Interrupt
B8h - B9h	Interrupt Controller	Interrupt Controller	Interrupt

Table 7-2. Fixed I/O Ranges Decoded by PCH (Sheet 3 of 3)

I/O Address	Read Target	Write Target	Internal Unit
BCh - BDh	Interrupt Controller	Interrupt Controller	Interrupt
200-207h	Gameport Low	Gameport Low	Forwarded to LPC/eSPI
208-20Fh	Gameport High	Gameport High	Forwarded to LPC/eSPI
4D0h – 4D1h	Interrupt Controller	Interrupt Controller	Interrupt Controller
CF9h	Reset Generator	Reset Generator	Interrupt controller

Note:
1. Only if the Port 61 Alias Enable bit (GCS.P61AE) is set. Otherwise, the target is PCIe.

7.2.3 Variable I/O Decode Ranges

Table 7-3, "Variable I/O Decode Ranges" shows the Variable I/O Decode Ranges. They are set using Base Address Registers (BARs) or other config bits in the various configuration spaces. The PnP software (PCI or ACPI) can use their configuration mechanisms to set and adjust these values.

Warning: The Variable I/O Ranges should not be set to conflict with the Fixed I/O Ranges. Unpredictable results if the configuration software allows conflicts to occur. The PCH does not perform any checks for conflicts.

Table 7-3. Variable I/O Decode Ranges

Range Name	Mappable	Size (Bytes)	Target
ACPI	Anywhere in 64K I/O Space	96	Power Management
IDE Bus Master	Anywhere in 64K I/O Space	16 or 32 Bytes	AMT IDE-R
SMBus	Anywhere in 64K I/O Space	32	SMB Unit
TCO	Anywhere in 64K I/O Space	32	SMB Unit
Parallel Port	3 ranges in 64K I/O Space	8	LPC Peripheral
Serial Port 1	8 Ranges in 64K I/O Space	8	LPC Peripheral
Serial Port 2	8 Ranges in 64K I/O Space	8	LPC Peripheral
Serial Port3	8 Ranges in 64K I/O space	8	LPC Peripheral
Floppy Disk Controller	2 Ranges in 64K I/O Space	8	LPC Peripheral
LPC Generic 1	Anywhere in 64K I/O Space	4 to 256 Bytes	LPC/eSPI
LPC Generic 2	Anywhere in 64K I/O Space	4 to 256 Bytes	LPC/eSPI
LPC Generic 3	Anywhere in 64K I/O Space	4 to 256 Bytes	LPC/eSPI
LPC Generic 4	Anywhere in 64K I/O Space	4 to 256 Bytes	LPC/eSPI
I/O Trapping Ranges	Anywhere in 64K I/O Space	1 to 256 Bytes	Trap
SATA Index/Data Pair	Anywhere in 64K I/O Space	16	SATA Host Controller
SSATA Index/Data Pair	Anywhere in 64K I/O Space	16	SSATA Host Controller
PCI Express Root Ports	Anywhere in 64K I/O Space	I/O Base/Limit	PCI Express Root Ports 1-12
Keyboard and Text (KT)	Anywhere in 64K I/O Space	8	AMT Keyboard and Text
GbE LAN MAC	Anywhere in 65K I/O Space	32	1 GbE LAN MAC Unit

Note: All ranges are decoded directly from DMI.

7.3 Memory Map

Table 7-4, "PCH Memory Decode Ranges (from CPU perspective)" shows (from the CPU perspective) the memory ranges that the PCH will decode. Cycles that arrive from DMI that are not directed to any of the internal memory targets that decode directly from DMI will be master aborted.

PCIe cycles generated by external PCIe masters will be positively decoded unless they fall in the PCI-PCI bridge memory forwarding ranges (those addresses are reserved for PCI peer-to-peer traffic). If the cycle is not in the internal LAN controller's range, it will be forwarded up to DMI. Software must not attempt locks to the PCH's memory-mapped I/O ranges.

Note: Total ports are different for the different SKUs.

Table 7-4. PCH Memory Decode Ranges (from CPU perspective) (Sheet 1 of 2)

Memory Range	Target	Dependency/Comments
000E0000 - 000EFFFF	LPC/eSPI or SPI	Bit 6 in BIOS Decode Enable Register is set
000F0000 - 000FFFFF	LPC/eSPI or SPI	Bit 7 in BIOS Decode Enable Register is set
FECXX000 - FECXX040	I/O(x)APIC inside PCH	X controlled via APIC Range Select (ASEL) field and APIC Enable (AEN) bit.
FEC10000 - FEC17FFF	PCIe port 1	PCIe root port 1 I/OxApic Enable (PAE) set
FEC18000 - FEC1FFFF	PCIe port 2	PCIe root port 2 I/OxApic Enable (PAE) set
FEC20000 - FEC27FFF	PCIe port 3	PCIe root port 3 I/OxApic Enable (PAE) set
FEC28000 - FEC2FFFF	PCIe port 4	PCIe root port 4 I/OxApic Enable (PAE) set
FEC30000 - FEC37FFF	PCIe port 5	PCIe root port 5 I/OxApic Enable (PAE) set
FEC38000 - FEC3FFFF	PCIe port 6	PCIe root port 6 I/OxApic Enable (PAE) set
FEC40000 - FEC47FFF	PCIe port 7	PCIe root port 7 I/OxApic Enable (PAE) set
FEC48000 - FEC4FFFF	PCIe port 8	PCIe root port 8 I/OxApic Enable (PAE) set
FEC50000 - FEC57FFF	PCIe port 9	PCIe root port 9 I/OxApic Enable (PAE) set
FEC58000 - FEC5FFFF	PCIe port 10	PCIe root port 10 I/OxApic Enable (PAE) set
FEF0_0000h - FEFF_FFFFh	LPC/eSPI or SPI	UCPR.UPRE
FFC0 0000 - FFC7 FFFF FF80 0000 - FF87 FFFF	LPC/eSPI or SPI	Bit 8 in BIOS Decode Enable Register
FFC8 0000 – FFCF FFFF FF88 0000 - FF8F FFFF	LPC/eSPI or SPI	Bit 9 in BIOS Decode Enable Register
FFD0 0000 - FFD7 FFFF FF90 0000 - FF97 FFFF	LPC/eSPI or SPI	Bit 10 in BIOS Decode Enable Register is set
FFD8 0000 – FFDF FFFF FF98 0000 - FF9F FFFF	LPC/eSPI or SPI	Bit 11 in BIOS Decode Enable Register is set
FFE0 000 - FFE7 FFFF FFA0 0000 - FFA7 FFFF	LPC/eSPI or SPI	Bit 12 in BIOS Decode Enable Register is set
FFE8 0000 – FFEF FFFF FFA8 0000 – FFAF FFFF	LPC/eSPI or SPI	Bit 13 in BIOS Decode Enable Register is set
FFF0 0000 - FFF7 FFFF FFB0 0000 - FFB7 FFFF	LPC/eSPI or SPI	Bit 14 in BIOS Decode Enable Register is set
FFF8 0000 – FFFF FFFF FFB8 0000 – FFBF FFFF	LPC/eSPI or SPI	Always enabled. The top two 64 KB blocks in this range can be swapped by the PCH.
FF70 0000 - FF7F FFFF FF30 0000 - FF3F FFFF	LPC/eSPI or SPI	Bit 3 in BIOS Decode Enable Register is set

Table 7-4. PCH Memory Decode Ranges (from CPU perspective) (Sheet 2 of 2)

Memory Range	Target	Dependency/Comments
FF60 0000 - FF6F FFFF FF20 0000 - FF2F FFFF	LPC/eSPI or SPI	Bit 2 in BIOS Decode Enable Register is set
FF50 0000 - FF5F FFFF FF10 0000 - FF1F FFFF	LPC/eSPI or SPI	Bit 1 in BIOS Decode Enable Register is set
FF40 0000 - FF4F FFFF FF00 0000 - FF0F FFFF	LPC/eSPI or SPI	Bit 0 in BIOS Decode Enable Register is set
FED0 X000h-FED0 X3FFh	HPET	BIOS determines "fixed" location which is one of four 1 KB ranges where X (in the first column) is 0h, 1h, 2h, or 3h
FED4_0000h – FED4_7FFFh	LPC or SPI (set by strap)	TPM and Trusted Mobile KBC
FED5_0000h – FED5_FFFFh	Intel ME	Always enabled
FED7_0000 – FED7_4FFF	Internal Device	Security feature related
4 KB anywhere in 4-GB range	1GbE Controller (LAN space on Flash)	Enable via standard PCI mechanism (Device 31:Function 6)
64 KB anywhere in 64-bit address range	USB 3.0 Host Controller	Enable via standard PCI mechanism (Device 20, Function 0)
16 KB anywhere in 64-bit addressing space	HD Audio Subsystem	Enable via standard PCI mechanism (Device 31, Function 3)
4 KB anywhere in 64-bit addressing space	HD Audio Subsystem	Enable via standard PCI mechanism (Device 31, Function 3)
64 KB anywhere in 64-bit addressing space	HD Audio Subsystem	Enable via standard PCI mechanism (Device 31, Function 3)
64 KB anywhere in 4-GB range	LPC/eSPI	LPC Generic Memory Range. Enable via setting bit[0] of the LPC Generic Memory Range register (D31:F0:offset 98h)
32 Bytes anywhere in 64-bit address range	SM Bus	Enable via standard PCI mechanism (Device 31: Function 4)
2 KB anywhere above 64 KB to 4 GB range	SATA Host Controller	AHCI memory-mapped registers. Enable via standard PCI mechanism (Device 23: Function 0)
2 KB anywhere above 64 KB to 4 GB range	SSATA Host Controller	AHCI memory-mapped registers. Enable via standard PCI mechanism (Device 17: Function 0)
Memory Base/Limit anywhere in 4GB range	PCI Express Root Ports 1-20	Enable via standard PCI mechanism
Prefetchable Memory Base/ Limit anywhere in 64-bit address range	PCI Express Root Ports 1-20	Enable via standard PCI mechanism
4 KB anywhere in 64-bit address range	Thermal Reporting	Enable via standard PCI mechanism (Device 20: Function 2)
16 Bytes anywhere in 64-bit address range	HECI #1, #2, #3	Enable via standard PCI mechanism (Device 22: Function 0-1, 4)
4 KB anywhere in 4-GB range	AMT Keyboard and Text	Enable via standard PCI mechanism (Device 22: Function 3)

7.3.1 Boot-Block Update Scheme

The PCH supports a "Top-Block Swap" mode that has the PCH swap the top block in the FWH or SPI flash (the boot block) with another location. This allows for safe update of the Boot Block (even if a power failure occurs). When the "top-swap" enable bit is set, the PCH will invert A16 for cycles going to the upper two 64 KB blocks in the FWH or appropriate address lines as selected in Boot Block Size (BOOT_BLOCK_SIZE) soft strap for SPI.

Specifically for FHW, in this mode accesses to FFFF_0000h-FFFF_FFFFh are directed to FFFE_0000h-FFFE_FFFFh and vice versa. When the Top Swap Enable bit is 0, the PCH will not invert A16.

Specifically for SPI, in this mode the "Top-Block Swap" behavior is as described below. When the Top Swap Enable bit is 0, the PCH will not invert any address bit.

Table 7-5. SPI Mode Address Swapping

BOOT_BLOCK_SIZE Value	Accesses to	Being Directed to
000 (64KB)	FFFF_0000h - FFFF_FFFFh	FFFE_0000h - FFFE_FFFFh and vice versa
001 (128KB)	FFFE_0000h - FFFF_FFFFh	FFFC_0000h - FFFD_FFFFh and vice versa
010 (256KB)	FFFC_0000h - FFFF_FFFFh	FFF8_0000h - FFFB_FFFFh and vice versa
011 (512KB)	FFF8_0000h - FFFF_FFFFh	FFF0_0000h - FFF7_FFFFh and vice versa
100 (1MB)	FFF0_0000h - FFFF_FFFFh	FFE0_0000h - FFEF_FFFFh and vice versa
101 - 111	Reserved	Reserved
Note: This bit is automatically set to 0 by RTCRST#, but not by PLTRST#.		

§

8 PCH and System Clocks

8.1 Acronyms

Acronyms	Description
ICC	Integrated Clock Controller
LPC	Low Pin Count
PCH	Platform Controller Hub
PLL	Phase Locked Loop Circuit
SSC	Spread Spectrum Clocking

8.2 Overview

This document describes the signals and different clocking modes that the Intel® C620 Series Chipset PCH supports. How you route the signals is the province of the appropriate platform PDG, as each platform can have different rules and restrictions on how the clocks are routed, connectivity, and modes supported.

8.3 Signal Descriptions

Name	Type	SSC Capable	Description	Connection
CLKOUT_ITPXDP_P CLKOUT_ITPXDP_N	O	Yes	**Differential Intel ITP Debug Clock:** 100 MHz differential output to processor XDP/Intel ITP connector on the platform. Not there in External Mode.	ITP Header
CLKOUT_NSSCCAP0_P CLKOUT_NSSCCAP0_N	O	Yes	**Non-spread capable:** 100 MHz Clock that is capable of being selected as non-spread spectrum different than the SRC and PLAT clocks.	
CLKOUT_NSSCCAP1_P CLKOUT_NSSCCAP1_N	O	Yes	**Non-spread capable:** 100 MHz Clock that is capable of being selected as non-spread spectrum different than the SRC and PLAT clocks	
CLKOUT_PLAT0_P CLKOUT_PLAT0_N	O	Yes	**Platform Clock:** 100 MHz Differential clock for use on the Platform for connections to the PCH. PCIe 4.0 compliant	PCH PCIe
CLKOUT_PLAT1_P CLKOUT_PLAT1_N	O	Yes	**Platform Clock:** 100 MHz Differential clock for use on the Platform for connections to the PCH. PCIe 4.0 compliant	PCH PCIe
CLKOUT_SRC_N_0 CLKOUT_SRC_P_0	O	Yes	**Source Clock:** 100 MHz Differential clocks that are PCIe 4.0 compliant	PCH PCIe, CPU Bclk, CPU PCIe, DBxxxzl
CLKOUT_SRC_N_1 CLKOUT_SRC_P_1	O	Yes	**Source Clock:**100 MHz Differential clocks that are PCIe 4.0 compliant	PCH PCIe, CPU Bclk, CPU PCIe, DBxxxzl
CLKOUT_SRC_N_2 CLKOUT_SRC_P_2	O	Yes	**Source Clock:**100 MHz Differential clocks that are PCIe 4.0 compliant	PCH PCIe, CPU Bclk, CPU PCIe, DBxxxzl
CLKOUT_SRC_N_3 CLKOUT_SRC_P_3	O	Yes	**Source Clock:** 100 MHz Differential clocks that are PCIe 4.0 compliant	PCH PCIe, CPU Bclk, CPU PCIe, DBxxxzl
CLKOUT_SRC_N_4 CLKOUT_SRC_P_4	O	Yes	**Source Clock:** 100 MHz Differential clocks that are PCIe 4.0 compliant	PCH PCIe, CPU Bclk, CPU PCIe, DBxxxzl
CLKOUT_SRC_N_5 CLKOUT_SRC_P_5	O	Yes	**Source Clock:** 100 MHz Differential clocks that are PCIe 4.0 compliant	PCH PCIe, CPU Bclk, CPU PCIe, DBxxxzl

Name	Type	SSC Capable	Description	Connection
CLKOUT_SRC_N_6 CLKOUT_SRC_P_6	O	Yes	**Source Clock:** 100 MHz Differential clocks that are PCIe 4.0 compliant	PCH PCIe, CPU Bclk, CPU PCIe, DBxxxzl
CLKOUT_SRC_N_7 CLKOUT_SRC_P_7	O	Yes	**Source Clock:** 100 MHz Differential clocks that are PCIe 4.0 compliant	PCH PCIe, CPU Bclk, CPU PCIe, DBxxxzl
CLKOUT_SRC_N_8 CLKOUT_SRC_P_8	O	Yes	**Source Clock:** 100 MHz Differential clocks that are PCIe 4.0 compliant	PCH PCIe, CPU Bclk, CPU PCIe, DBxxxzl
CLKOUT_SRC_N_9 CLKOUT_SRC_P_9	O	Yes	**Source Clock:** 100 MHz Differential clocks that are PCIe 4.0 compliant	PCH PCIe, CPU Bclk, CPU PCIe, DBxxxzl
CLKOUT_SRC_N_10 CLKOUT_SRC_P_10	O	Yes	**Source Clock:** 100 MHz Differential clocks that are PCIe 4.0 compliant	PCH PCIe, CPU Bclk, CPU PCIe, DBxxxzl
CLKOUT_SRC_N_11 CLKOUT_SRC_P_11	O	Yes	**Source Clock:** 100 MHz Differential clocks that are PCIe 4.0 compliant	PCH PCIe, CPU Bclk, CPU PCIe, DBxxxzl
CLKOUT_SRC_N_12 CLKOUT_SRC_P_12	O	Yes	**Source Clock:** 100 MHz Differential clocks that are PCIe 4.0 compliant	PCH PCIe, CPU Bclk, CPU PCIe, DBxxxzl
CLKOUT_SRC_N_13 CLKOUT_SRC_P_13	O	Yes	**Source Clock:** 100 MHz Differential clocks that are PCIe 4.0 compliant	PCH PCIe, CPU Bclk, CPU PCIe, DBxxxzl
CLKOUT_SRC_N_14 CLKOUT_SRC_P_14	O	Yes	**Source Clock:** 100 MHz Differential clocks that are PCIe 4.0 compliant	PCH PCIe, CPU Bclk, CPU PCIe, DBxxxzl
CLKOUT_SRC_N_15 CLKOUT_SRC_P_15	O	Yes	**Source Clock:** 100 MHz Differential clocks that are PCIe 4.0 compliant	PCH PCIe, CPU Bclk, CPU PCIe, DBxxxzl
CLKOUT_LPC0_ESPI_CLK	O	No	**LPC Clock 0:** Single ended 24 MHz clock for LPC and eSPI. This output is a dedicated clock that can not be turned off.	LPC, eSPI, TPM
GPP_A10_CLKOUT_LPC1	O	No	**LPC Clock 1:** Single ended 24 MHz clock for devices attached to the LPC or for general usage	LPC
GPP_A16_CLKOUT_LPC2	O	No	**LPC Clock 2:** Single ended 24 MHz clock for devices attached to the LPC or for general usage.	LPC
CLOCKSE_BMCCLK	O	No	**BMC Clock:** Single ended 48 MHz clock, used to provide reference for USB clocking. Expected to be sent to an external controller.	
CLOCKSE_PMSYNCCLK1	O	No	**PMSYNC Clock:** Single ended clock running at 24 MHz. The provides reference clocking for PM_SYNC1.	
CLOCKSE_PMSYNCCLK2	O	No	**PMSYNC Clock:** Single ended clock running at 24 MHz. The provides reference clocking for PM_SYNC2.	
GPD8_SUSCLK	O	No	**Sus Clock:** 32 kHz clock sourced from the RTC.	
GPP_B5_SRCCLKREQ0#	I/O	N/A	**Source Clock Request**: Active low signal that is used to enable/disable one of the 100 MHz CLKOUT_SRC_N/P differential clocks. Multiplexed with a general purpose I/O pin.	
GPP_B6_SRCCLKREQ1#	I/O	N/A	**Source Clock Request**: Active low signal that is used to enable/disable one of the 100 MHz CLKOUT_SRC_N/P differential clocks. Multiplexed with a general purpose I/O pin.	
GPP_B7_SRCCLKREQ2#	I/O	N/A	**Source Clock Request**: Active low signal that is used to enable/disable one of the 100 MHz CLKOUT_SRC_N/P differential clocks. Multiplexed with a general purpose I/O pin.	
GPP_B8_SRCCLKREQ3#	I/O	N/A	**Source Clock Request**: Active low signal that is used to enable/disable one of the 100 MHz CLKOUT_SRC_N/P differential clocks. Multiplexed with a general purpose I/O pin.	
GPP_B9_SRCCLKREQ4#	I/O	N/A	**Source Clock Request**: Active low signal that is used to enable/disable one of the 100 MHz CLKOUT_SRC_N/P differential clocks. Multiplexed with a general purpose I/O pin.	

Name	Type	SSC Capable	Description	Connection
GPP_B10_SRCCLKREQ5#	I/O	N/A	**Source Clock Request:** Active low signal that is used to enable/disable one of the 100 MHz CLKOUT_SRC_N/P differential clocks. Multiplexed with a general purpose I/O pin.	
GPP_H0_SRCCLKREQ6#	I/O	N/A	**Source Clock Request**: Active low signal that is used to enable/disable one of the 100 MHz CLKOUT_SRC_N/P differential clocks. Multiplexed with a general purpose I/O pin.	
GPP_H1_SRCCLKREQ7#	I/O	N/A	**Source Clock Request**: Active low signal that is used to enable/disable one of the 100 MHz CLKOUT_SRC_N/P differential clocks. Multiplexed with a general purpose I/O pin.	
GPP_H2_SRCCLKREQ8#	I/O	N/A	**Source Clock Request**: Active low signal that is used to enable/disable one of the 100 MHz CLKOUT_SRC_N/P differential clocks. Multiplexed with a general purpose I/O pin.	
GPP_H3_SRCCLKREQ9#	I/O	N/A	**Source Clock Request**: Active low signal that is used to enable/disable one of the 100 MHz CLKOUT_SRC_N/P differential clocks. Multiplexed with a general purpose I/O pin.	
GPP_H4_SRCCLKREQ10#	I/O	N/A	**Source Clock Request**: Active low signal that is used to enable/disable one of the 100 MHz CLKOUT_SRC_N/P differential clocks. Multiplexed with a general purpose I/O pin.	
GPP_H5_SRCCLKREQ11#	I/O	N/A	**Source Clock Request**: Active low signal that is used to enable/disable one of the 100 MHz CLKOUT_SRC_N/P differential clocks. Multiplexed with a general purpose I/O pin.	
GPP_H6_SRCCLKREQ12#	I/O	N/A	**Source Clock Request**: Active low signal that is used to enable/disable one of the 100 MHz CLKOUT_SRC_N/P differential clocks. Multiplexed with a general purpose I/O pin.	
GPP_H7_SRCCLKREQ13#	I/O	N/A	**Source Clock Request**: Active low signal that is used to enable/disable one of the 100 MHz CLKOUT_SRC_N/P differential clocks. Multiplexed with a general purpose I/O pin.	
GPP_H8_SRCCLKREQ14#	I/O	N/A	**Source Clock Request**: Active low signal that is used to enable/disable one of the 100 MHz CLKOUT_SRC_N/P differential clocks. Multiplexed with a general purpose I/O pin.	
GPP_H9_SRCCLKREQ15#	I/O	N/A	**Source Clock Request**: Active low signal that is used to enable/disable one of the 100 MHz CLKOUT_SRC_N/P differential clocks. Multiplexed with a general purpose I/O pin.	
XTAL_IN	I	N/A	**Crystal Input:** Input connection for 48 MHz crystal to PCH oscillator circuit	
XTAL_OUT	O	N/A	**Crystal Output:** Output connection for 48 MHz crystal to PCH oscillator circuit	
XCLK_BIASREF	I/O	N/A	**Differential Clock Bias Reference**: Used to set BIAS reference for differential clocks	
LAN_XTAL_IN	I	N/A	**Crystal Input:** Input connection for the 25 MHz crystal used for the 10 GbE LAN	
LAN_XTAL_OUT	O	N/A	**Crystal Output:** Output connection for the 25 MHz crystal used for the 10 GbE LAN	
EXTCLKN EXTCLKP	I	N/A	**External Clocks** Differential 100 MHz clock that is used when in the external clocking mode. When the PCH is configured to be in the iSCLK or Platform Hybrid mode, these pins must be No Connects.	

Notes:
1. The SRCCLKRQ# signals can be configured to map to any of the PCH PCI Express Root Ports that are using any of the CLKOUT_SRC_P/N differential pairs.

8.4 Clock Signal Planes and States

Table 8-1. Clock Signal Planes and States

Signal Name	Power Plane	During Reset	Immediately After Reset	S3/S4/S5	Deep Sx
CLKOUT_ITPXDP_P CLKOUT_ITPXDP_N	Primary	Toggling	Toggling	Driven Low	OFF
CLKOUT_NSSCCAP[1:0]_P CLKOUT_NSSCCAP[1:0]_N	Primary	Toggling	Toggling	Driven Low	OFF
CLKOUT_PLAT[1:0]_P CLKOUT_PLAT[1:0]_N	Primary	Toggling	Toggling	Driven Low	OFF
CLKOUT_SRC_P[15:0] CLKOUT_SRC_N[15:0]	Primary	Toggling	Toggling	Driven Low	OFF
CLKOUT_LPC[2:0]	Primary	Toggling	Toggling	Driven Low	OFF
CLOCKSE_BMC	Primary	Toggling	Toggling	Driven Low	OFF
CLOCKSE_PMSYNCCLK[2:1]	Primary	Toggling	Toggling	Driven Low	OFF
SUSCLK	Primary	Toggling	Toggling	Toggling	OFF

Note:
1. Signal names have been shortened to only reflect those with clock functions.

8.5 Clocking Overview

The Intel® C620 Series Chipset works with three different clocking solutions for the Intel® Xeon® Processor Scalable Family-Based Platform: native isCLK clocking, platform hybrid clocking and external clocking.

- Native isCLK is the implementation where the Intel® C620 Series Chipset provides a complete clocking solution. No external components can be used.
- With platform hybrid, the PCH operates the same as in Native isCLK with the following exception: one or more 100 MHz clocks from the Intel® C620 Series Chipset are routed to external clock buffers to increase the number of 100 MHz clocks for the system. In platform hybrid all clocks can come from the buffers, or there can be a mix of clocks from the buffers and 100 MHz clocks from the PCH.
- External clocking effectively replaces the Intel® C620 Series Chipset clock controller with a CK420BQ and appropriate clock buffers for the 100 MHz differential clocks. The singled ended clocks will still come out of the Intel® C620 Series Chipset, and the 24 MHz clock must be used from the Intel® C620 Series Chipset.
- There is no special setup or programming needed for isCLK vs Platform Hybrid. However, for external clocking multiple softstraps are used to enable running in native isCLK mode vs external clocking mode.

8.6 Clock Interfaces

8.6.1 Crystal Oscillators and Clock Inputs

There are three crystals used in the PCH: 32 kHz used for the RTC, 25 MHz that supplies the clock to the 10 GbE LAN, and 48 MHz that provides clocking for the Intel QuickAssist Technology functions, sPCH functions (DMI, USB, PCIe, SATA, etc), and internal data paths. In addition, there is a differential clock input that is only used when in External Clock mode. For all configurations, the RTC and 48 MHz crystals must be connected. The 25-MHz crystal is not needed for implementations that do not have the

10 Gb Ethernet Controller enabled. In external mode, EXTCLKN/P is driven by an external 100-MHz differential clock. The other clocks are still sourced from the 48-MHz Crystal.

8.6.2 100-MHz Differential Clocks

CLKOUT_NSSCCAP[1:0] are a pair of 100 MHz differential clocks that can have spread spectrum enabled and disabled on them independently of the rest of the 100 MHz differential clocks. This can be done because these clocks are sourced from a different PLL than the rest of the 100 MHz clocks. One effect of this is that these clocks can not be used to drive the BLCK inputs on the CPU, drive clock inputs to PCIe slots/devices that connect to the CPU, or clocks inputs to PCIe slots/devices that connect to the PCH. They are best considered independent clocks that can be used by components that do not need them to be used for transferring data between the CPUs and the PCHs. For example, if you had a SAS controller on the board that needed 100 MHz non-spread clocks for transferring data between the controller and the hard drives, the CLKOUT_NSSCCAP clocks could be used for this purpose.

The rest of the clocks (CLKOUT_ITPXDP, CLKOUT_PLAT[1:0], CLKOUT_SRC_N/P_[15:0]) are all generated by the same PLL, and operate the same with regards to spread spectrum. SSC is either on for all these clocks, or off. The PLL for these clocks also provides the clocking for USB3. The USB3 specification requires SSC to be turned on with a certain range, resulting in these clocks all, by default, having spread spectrum enabled operating in that mode. It is possible to disable SSC on these clocks, but that will end up in non-compliance with the USB3 specification.

While all are 100 MHz clocks, there are some differences between the clocks with regards to power and ground connections. For connections to the CPU BCLKs, CLKOUT_SRC clocks must be used. See the appropriate Platform Design Guide for more details on clock selection.

In Platform Hybrid mode, it is possible to use either the CLKOUT_SRC clocks to drive the clock buffers, or the CLKOUT_PLAT[1:0] can be used. The preference would be to use the CLKOUT_SRC clocks. Again, see your PDG for guidelines.

All 100 MHz differential clocks are only active in the S0 state. When in Sx state, they will be driven low.

The CLKOUT_SRC clocks can have their outputs enabled/disabled either via FITC setups, HECI commands or with the SRCCLKREQ# pins. The SRCCLKREQ# pins are a good option for PCIe slots as they can be used to externally enable or disable the output of the CLKOUT_SRC pins depending on whether a card is in the slot or not.

8.6.3 Single Ended Clocks

The Intel® C620 Series Chipset will provide single ended clocks for LPC and eSPI, 48 MHz clock for USB/BMC support, PM_Sync clocks for connection to the CPU, and Susclks in all clocking configurations: isCLK, Platform Hybrid and External Clock. The LPC/eSPI, PM_SYNC, and 48 MHz clocks are operational in S0 state, but not in Sx state. Susclk is operational in the Sx states. The source of the LPC/eSPI, 48 MHz, and PM_Sync clocks comes from the 48 MHz crystal, so that crystal is needed in all three clocking modes. The Susclk comes from the RTC crystal.

The CLKOUT_LPC[2:0] clocks are not identical in operation. CLKOUT_LPC0 is intended to be used for driving your TPM device in addition to one LPC device. This clock is always free running. It can not be started or stopped. Because of that, this clock is not enabled on a rising edge or disabled on a falling edge, resulting in the beginning clocks

and ending clocks not meeting specs with regards to clock periods/high times at their start and stop. These non-compliant clocks will occur for many clock periods while PLTRST# is low, when the platform is in reset. When PLTRST# is high, the CLKOUT_LPC0 clock will operate per specification.

When the Intel® C620 Series Chipset is configured to be run in endpoint only mode, there are restrictions on the clocking supported. In this configuration, only external clocking is allowed. Native isCLK and Platform Hybrid is not permitted.

8.7 Integrated Clock Controller FW Accessible Registers Summary

The integrated clock controller FW accessible registers are distributed withing the PCH Private Configuration Space which is accessible through the PCH sideband interface using the following Target Port (Destination port) Identification of 0xDC

Table 8-2. Summary of Integrated Clock Controller FW Accessible Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
1000h	1003h	Timing Control SRC Clock (TMCSRCCLK)—Offset 1000h	0h
1004h	1007h	Timing Control SRC Clock Register 2 (TMCSRCCLK2)—Offset 1004h	0h
1008h	100Bh	Enable Control CLKREQ (ENCCKRQ)—Offset 1008h	0h
100Ch	100Fh	Mask Control CLKREQ (MSKCKRQ)—Offset 100Ch	0h
1020h	1023h	ICC Security (ICCSEC)—Offset 1020h	0h
1024h	1027h	CLKREQ Mapping to CLKOUT_SRC (CKRQMAPSRC)—Offset 1024h	76543210h
1028h	102Bh	CLKREQ Mapping to CLKOUT_SRC Register 2 (CKRQMAPSRC2)—Offset 1028h	76543210h
102Ch	102Fh	Power Management (PM)—Offset 102Ch	0h
1034h	1037h	ICC Debug (ICCDBG)—Offset 1034h	0h
2000h	2003h	USB3Gen2PCIe PLL Control (G2PLLCTRL)—Offset 2000h	0h

8.7.1 Timing Control SRC Clock (TMCSRCCLK)—Offset 1000h

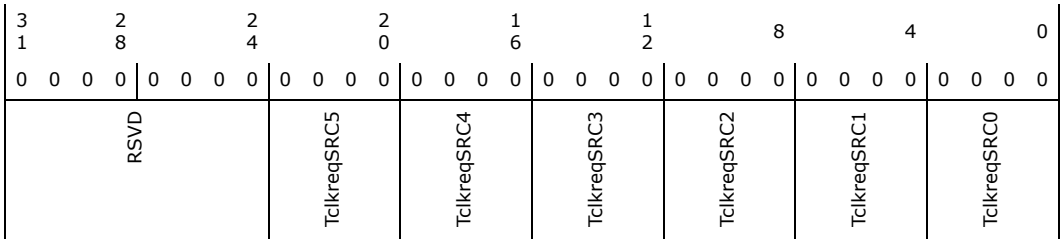
Controls minimum enable timing of PCIe SRC Clocks for support of PCIe L1Off using CLKREQ#

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23:20	0h RW	**Minimum Time from Effective Mapped CLKREQ# assertion to CLKOUT_SRC5 ungated (TclkreqSRC5):** See description for TclkreqSRC0 field[3:0]
19:16	0h RW	**Minimum Time from Effective Mapped CLKREQ# assertion to CLKOUT_SRC4 ungated (TclkreqSRC4):** See description for TclkreqSRC0 field[3:0]
15:12	0h RW	**Minimum Time from Effective Mapped CLKREQ# assertion to CLKOUT_SRC3 ungated (TclkreqSRC3):** See description for TclkreqSRC0 field[3:0]
11:8	0h RW	**Minimum Time from Effective Mapped CLKREQ# assertion to CLKOUT_SRC2 ungated (TclkreqSRC2):** See description for TclkreqSRC0 field[3:0]
7:4	0h RW	**Minimum Time from Effective Mapped CLKREQ# assertion to CLKOUT_SRC1 ungated (TclkreqSRC1):** See description for TclkreqSRC0 field[3:0]

Bit Range	Default and Access	Field Name (ID): Description
3:0	0h RW	***Note:*** Minimum Time from Effective Mapped CLKREQ# assertion to CLKOUT_SRC0 ungated (TclkreqSRC0): Minimum elapsed time from the Effective Mapped CLKREQ# signal assertion to the corresponding PCIe SRC Clock being allowed to toggle. This minimum time of CLKOUT_SRCn[P/N] staying gated at low voltage allows for the PCIe devices voltage rail to ramp to safe level on the exit from L1Off state before being driven with clock from the PCH. 0000: 0 us minimum elapsed time from Effective Mapped CLKREQ# assertion to corresponding CLKOUT_SRCn ungated - Default - This setting is for PCIe device that does not support L1Off. This setting must also be used when corresponding CLKOUT_SRCn is configured for non-PCIe clock use, i.e., as MEX reference clock, etc. 0001: 5 us minimum elapsed time ... 0010: 10 us minimum elapsed time ... 0011: 15 us minimum elapsed time ... 0100: 20 us minimum elapsed time ... 0101: 25 us minimum elapsed time ... 0110: 30 us minimum elapsed time ... 0111: 35 us minimum elapsed time ... 1000: 40 us minimum elapsed time ... 1001: 45 us minimum elapsed time ... 1010: 50 us minimum elapsed time ... 1011: 60 us minimum elapsed time ... 1100: 70 us minimum elapsed time ... 1101: 80 us minimum elapsed time ... 1110: 90 us minimum elapsed time ... 1111: 100 us minimum elapsed time ... ***Notes:*** The effective time to restart the CLKOUT_SRCn should be the larger of the programmed minimum delay, or of the normal natural latency to restart the clock from the current powered state of ICC. The effective time must not be cummulative. 0us minimum elapsed time setting results in the normal latency that would be incurred to restart the CLKOUT_SRCn from the current state of ICC, i.e., if PLL source is already up and only the CLKOUT_SRCn is being gated, then the effective latency is the normal delay to ungate the clock at the output buffer. If PLL source has been power managed off, then the effective latency is the normal delay to restart the PLL, divider, and to ungate the clock at the output buffer. No other additional delay is incurred. Non-zero minimum elapsed time setting should result in similar delay as 0us minimum case described above when the normal latency for clock restart is naturally larger than the programmed setting. Non-zero minimum elapsed time setting should only result in the programmed latency when the normal latency for clock restart, with respect to the current state of ICC, is smaller than the programmed setting. BIOS is expected to program this register field based on its discovery of the PCIe device capabilities, such as device support for L1Off, and device Tpowerup timing. Trefclk_min is enforced by ICC only when the PCIe link is exiting from L1.OFF state where the device is unpowered. ICC will not enforce Trefclk_min when the link is exiting from other L1 improved states, such as L1.SNOOZE where the device is still powered for lower latency L1 improved exit. PCIe root port communicates this port state distinction to ICC. Port state information is communicated per port and is mapped 1:1 with the raw SRCCLKREQB_n from GPIO, i.e., port 0 state is associated with SRCCLKREQB_0. Thus the mapping function of CLKREQ# to CLKOUTSRC also needs to be applied to port state information prior to use.

8.7.2 Timing Control SRC Clock Register 2 (TMCSRCCLK2)—Offset 1004h

Controls minimum enable timing of PCIe SRC Clocks for support of PCIe L1Off using CLKREQ#

8.7.3 Enable Control CLKREQ (ENCCKRQ)—Offset 1008h

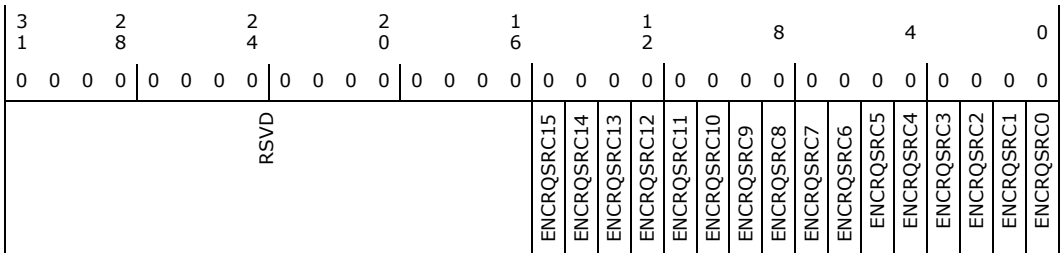
Controls enabling of Mapped CLKREQ#s for SRC clocks

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15	0h RW	**Enable Maskable Mapped CLKREQ# for CLKOUT_SRC15 (ENCRQSRC15):** Enable dynamic control of CLKOUT_SRC15 by the Maskable Mapped CLKREQ#. See description for ENCRQSRC0.
14	0h RW	**Enable Maskable Mapped CLKREQ# for CLKOUT_SRC14 (ENCRQSRC14):** Enable dynamic control of CLKOUT_SRC13 by the Maskable Mapped CLKREQ#. See description for ENCRQSRC0.
13	0h RW	**Enable Maskable Mapped CLKREQ# for CLKOUT_SRC13 (ENCRQSRC13):** Enable dynamic control of CLKOUT_SRC13 by the Maskable Mapped CLKREQ#. See description for ENCRQSRC0.
12	0h RW	**Enable Maskable Mapped CLKREQ# for CLKOUT_SRC12 (ENCRQSRC12):** Enable dynamic control of CLKOUT_SRC12 by the Maskable Mapped CLKREQ#. See description for ENCRQSRC0.
11	0h RW	**Enable Maskable Mapped CLKREQ# for CLKOUT_SRC11 (ENCRQSRC11):** Enable dynamic control of CLKOUT_SRC11 by the Maskable Mapped CLKREQ#. See description for ENCRQSRC0.
10	0h RW	**Enable Maskable Mapped CLKREQ# for CLKOUT_SRC10 (ENCRQSRC10):** Enable dynamic control of CLKOUT_SRC10 by the Maskable Mapped CLKREQ#. See description for ENCRQSRC0.
9	0h RW	**Enable Maskable Mapped CLKREQ# for CLKOUT_SRC9 (ENCRQSRC9):** Enable dynamic control of CLKOUT_SRC9 by the Maskable Mapped CLKREQ#. See description for ENCRQSRC0.
8	0h RW	**Enable Maskable Mapped CLKREQ# for CLKOUT_SRC8 (ENCRQSRC8):** Enable dynamic control of CLKOUT_SRC8 by the Maskable Mapped CLKREQ#. See description for ENCRQSRC0.
7	0h RW	**Enable Maskable Mapped CLKREQ# for CLKOUT_SRC7 (ENCRQSRC7):** Enable dynamic control of CLKOUT_SRC7 by the Maskable Mapped CLKREQ#. See description for ENCRQSRC0.
6	0h RW	**Enable Maskable Mapped CLKREQ# for CLKOUT_SRC6 (ENCRQSRC6):** Enable dynamic control of CLKOUT_SRC6 by the Maskable Mapped CLKREQ#. See description for ENCRQSRC0.
5	0h RW	**Enable Maskable Mapped CLKREQ# for CLKOUT_SRC5 (ENCRQSRC5):** Enable dynamic control of CLKOUT_SRC5 by the Maskable Mapped CLKREQ#. See description for ENCRQSRC0.

Bit Range	Default and Access	Field Name (ID): Description
4	0h RW	**Enable Maskable Mapped CLKREQ# for CLKOUT_SRC4 (ENCRQSRC4):** Enable dynamic control of CLKOUT_SRC4 by the Maskable Mapped CLKREQ#. See description for ENCRQSRC0.
3	0h RW	**Enable Maskable Mapped CLKREQ# for CLKOUT_SRC3 (ENCRQSRC3):** Enable dynamic control of CLKOUT_SRC3 by the Maskable Mapped CLKREQ#. See description for ENCRQSRC0.
2	0h RW	**Enable Maskable Mapped CLKREQ# for CLKOUT_SRC2 (ENCRQSRC2):** Enable dynamic control of CLKOUT_SRC2 by the Maskable Mapped CLKREQ#. See description for ENCRQSRC0.
1	0h RW	**Enable Maskable Mapped CLKREQ# for CLKOUT_SRC1 (ENCRQSRC1):** Enable dynamic control of CLKOUT_SRC1 by the Maskable Mapped CLKREQ#. See description for ENCRQSRC0.
0	0h RW	**Enable Maskable Mapped CLKREQ# for CLKOUT_SRC0 (ENCRQSRC0):** Enable dynamic control of CLKOUT_SRC0 by the Maskable Mapped CLKREQ#. This register bit may be updated dynamically. Hardware design must support this usage model. 0: Disable dynamic control of CLKOUT_SRCn - Default 1: Enable dynamic control of CLKOUT_SRCn by the maskable mapped CLKREQ# When enabled by this register bit, CLKOUT_SRCn is subject to gating/ungating control by the maskable mapped CLKREQ#. Furthermore, when latency tolerance permits, the maskable mapped CLKREQ# de-assertion allows for PLL source shutdown following the gating of CLKOUT_SRCn. Actual PLL shutdown accounts for votes from all other clock consumers of shared PLL source. When the maskable mapped CLKREQ# re-asserts, the minimum latency to restart the output clock is subject to the L1.Off Trefclkmin field TclkreqSRCn, register TMCSRCCLK. When disabled by this register bit, CLKOUT_SRCn is insensitive to the state of the maskable mapped CLKREQ#. CLKOUT_SRCn is either always running after ICC initialization, or always gated as globally configured via the corresponding CLKEN bit (residing inside iSCLK HIP register set.) ***Note:*** The output of this processing is referred to as Effective Mapped CLKREQ#.

8.7.4 Mask Control CLKREQ (MSKCKRQ)—Offset 100Ch

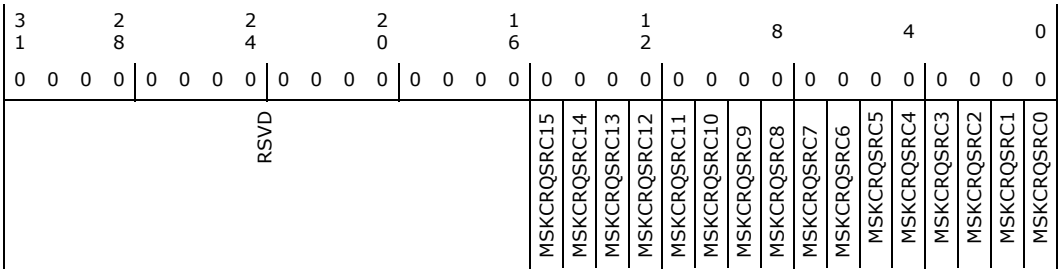
Controls Masking of Mapped CLKREQ#s for SRC clocks

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15	0h RW	**Mask Enable of Mapped CLKREQ# for CLKOUT_SRC15 (MSKCRQSRC15):** Force to inactive state the Mapped CLKREQ# for CLKOUT_SRC15. See description for MSKCRQSRC0.
14	0h RW	**Mask Enable of Mapped CLKREQ# for CLKOUT_SRC14 (MSKCRQSRC14):** Force to inactive state the Mapped CLKREQ# for CLKOUT_SRC14. See description for MSKCRQSRC0.
13	0h RW	**Mask Enable of Mapped CLKREQ# for CLKOUT_SRC13 (MSKCRQSRC13):** Force to inactive state the Mapped CLKREQ# for CLKOUT_SRC13. See description for MSKCRQSRC0.
12	0h RW	**Mask Enable of Mapped CLKREQ# for CLKOUT_SRC12 (MSKCRQSRC12):** Force to inactive state the Mapped CLKREQ# for CLKOUT_SRC12. See description for MSKCRQSRC0.
11	0h RW	**Mask Enable of Mapped CLKREQ# for CLKOUT_SRC11 (MSKCRQSRC11):** Force to inactive state the Mapped CLKREQ# for CLKOUT_SRC11. See description for MSKCRQSRC0.
10	0h RW	**Mask Enable of Mapped CLKREQ# for CLKOUT_SRC10 (MSKCRQSRC10):** Force to inactive state the Mapped CLKREQ# for CLKOUT_SRC10. See description for MSKCRQSRC0.
9	0h RW	**Mask Enable of Mapped CLKREQ# for CLKOUT_SRC9 (MSKCRQSRC9):** Force to inactive state the Mapped CLKREQ# for CLKOUT_SRC9. See description for MSKCRQSRC0.
8	0h RW	**Mask Enable of Mapped CLKREQ# for CLKOUT_SRC8 (MSKCRQSRC8):** Force to inactive state the Mapped CLKREQ# for CLKOUT_SRC8. See description for MSKCRQSRC0.
7	0h RW	**Mask Enable of Mapped CLKREQ# for CLKOUT_SRC7 (MSKCRQSRC7):** Force to inactive state the Mapped CLKREQ# for CLKOUT_SRC7. See description for MSKCRQSRC0.
6	0h RW	**Mask Enable of Mapped CLKREQ# for CLKOUT_SRC6 (MSKCRQSRC6):** Force to inactive state the Mapped CLKREQ# for CLKOUT_SRC6. See description for MSKCRQSRC0.
5	0h RW	**Mask Enable of Mapped CLKREQ# for CLKOUT_SRC5 (MSKCRQSRC5):** Force to inactive state the Mapped CLKREQ# for CLKOUT_SRC5. See description for MSKCRQSRC0.
4	0h RW	**Mask Enable of Mapped CLKREQ# for CLKOUT_SRC4 (MSKCRQSRC4):** Force to inactive state the Mapped CLKREQ# for CLKOUT_SRC4. See description for MSKCRQSRC0.
3	0h RW	**Mask Enable of Mapped CLKREQ# for CLKOUT_SRC3 (MSKCRQSRC3):** Force to inactive state the Mapped CLKREQ# for CLKOUT_SRC3. See description for MSKCRQSRC0.
2	0h RW	**Mask Enable of Mapped CLKREQ# for CLKOUT_SRC2 (MSKCRQSRC2):** Force to inactive state the Mapped CLKREQ# for CLKOUT_SRC2. See description for MSKCRQSRC0.
1	0h RW	**Mask Enable of Mapped CLKREQ# for CLKOUT_SRC1 (MSKCRQSRC1):** Force to inactive state the Mapped CLKREQ# for CLKOUT_SRC1. See description for MSKCRQSRC0.
0	0h RW	**Mask Enable of Mapped CLKREQ# for CLKOUT_SRC0 (MSKCRQSRC0):** Force to inactive state the Mapped CLKREQ# for CLKOUT_SRC0. This register bit may be updated dynamically. Hardware design must support this usage model. 0: No mask applied on the Mapped CLKREQ# for CLKOUT_SRCn - Default 1: Mask Mapped CLKREQ# for CLKOUT_SRCn to inactive state ***Note:*** The output of this processing is referred to as Maskable Mapped CLKREQ# Maskable Mapped CLKREQ# = Mapped CLKREQ# OR MSKCRQSRCn

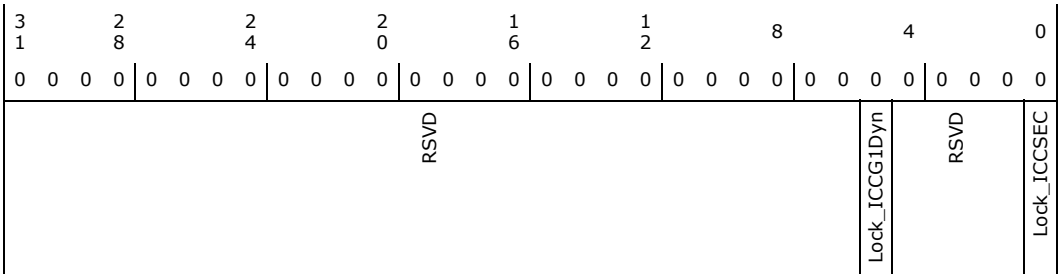
8.7.5 ICC Security (ICCSEC)—Offset 1020h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:6	0h RO	Reserved.
5	0h RW/L	**Lock Bit for Group 1 of Dynamically Configured ICC Registers (Lock_ICCG1Dyn):** This lock bit covers registers TMCSRCCLK, TMCSRCCLK2, ENCCKRQ 0: Target endpoint will accept all incoming requests as normal. 1: Target endpoint will deny incoming requests addressed to above listed ICC Registers, unless those requests are attributed with Intel ME SAI or PMC SAI. When denying a request, target endpoint will generate a completion packet as follows: Read will complete without data as Unsuccessful on IOSFSB (normally read completes with data as Successful on IOSFSB), Non-posted Write will complete without data as Unsuccessful on IOSFSB (normally non-posted write completes without data as Successful on IOSFSB) and the Write does not take effect at the register (i.e., register content is not affected.)
4:1	0h RO	Reserved.
0	0h RW/L	**Lock Bit for ICC Security Register (Lock_ICCSEC):** This lock bit covers register ICCSEC, i.e., this same register where this lock bit resides. 0: Target endpoint will accept all incoming requests as normal. 1: Target endpoint will deny incoming requests addressed to ICCSEC register, unless those requests are attributed with Intel ME SAI. When denying a request, target endpoint will generate a completion packet as follows: Read will complete without data as Unsuccessful on IOSFSB (normally read completes with data as Successful on IOSFSB), Non-posted Write will complete without data as Unsuccessful on IOSFSB (normally non-posted write completes without data as Successful on IOSFSB) and the Write does not take effect at the register (i.e., register content is not affected.)

8.7.6 CLKREQ Mapping to CLKOUT_SRC (CKRQMAPSRC)—Offset 1024h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 76543210h

31	28	24	20	16	12	8	4	0
0111	0110	0101	0100	0011	0010	0001	0000
CRQSELSRC7	CRQSELSRC6	CRQSELSRC5	CRQSELSRC4	CRQSELSRC3	CRQSELSRC2	CRQSELSRC1	CRQSELSRC0

Bit Range	Default and Access	Field Name (ID): Description
31:28	7h RW	**CLKRQ# Select for CLKOUT_SRC7 (CRQSELSRC7):** Select version of external input CLKRQ# for dynamic control of the output CLKOUT_SRC7. 0000: SRCCLKREQB_0 controls CLKOUT_SRC7 0001: SRCCLKREQB_1 controls CLKOUT_SRC7 0010: SRCCLKREQB_2 controls CLKOUT_SRC7 0011: SRCCLKREQB_3 controls CLKOUT_SRC7 0100: SRCCLKREQB_4 controls CLKOUT_SRC7 0101: SRCCLKREQB_5 controls CLKOUT_SRC7 0110: SRCCLKREQB_6 controls CLKOUT_SRC7 0111: SRCCLKREQB_7 controls CLKOUT_SRC7 - Default 1000-1111: Reserved ***Note:*** The output of this processing is referred to as Mapped CLKREQ#.
27:24	6h RW	**CLKRQ# Select for CLKOUT_SRC6 (CRQSELSRC6):** Select version of external input CLKRQ# for dynamic control of the output CLKOUT_SRC6. 0000: SRCCLKREQB_0 controls CLKOUT_SRC6 0001: SRCCLKREQB_1 controls CLKOUT_SRC6 0010: SRCCLKREQB_2 controls CLKOUT_SRC6 0011: SRCCLKREQB_3 controls CLKOUT_SRC6 0100: SRCCLKREQB_4 controls CLKOUT_SRC6 0101: SRCCLKREQB_5 controls CLKOUT_SRC6 0110: SRCCLKREQB_6 controls CLKOUT_SRC6 - Default 0111: SRCCLKREQB_7 controls CLKOUT_SRC6 1000-1111: Reserved ***Note:*** The output of this processing is referred to as Mapped CLKREQ#.
23:20	5h RW	**CLKRQ# Select for CLKOUT_SRC5 (CRQSELSRC5):** Select version of external input CLKRQ# (or internally generated version when applicable) for dynamic control of the output CLKOUT_SRC5. 0000: SRCCLKREQB_0 controls CLKOUT_SRC5 0001: SRCCLKREQB_1 controls CLKOUT_SRC5 0010: SRCCLKREQB_2 controls CLKOUT_SRC5 0011: SRCCLKREQB_3 controls CLKOUT_SRC5 0100: SRCCLKREQB_4 controls CLKOUT_SRC5 0101: SRCCLKREQB_5 controls CLKOUT_SRC5 - Default 0110: SRCCLKREQB_6 controls CLKOUT_SRC5 0111: SRCCLKREQB_7 controls CLKOUT_SRC5 1000: UFSCLKREQB controls CLKOUT_SRC5 - Use when CLKOUT_SRC5 is configured to source UFS reference clock (i.e., 19.2 MHz) 1001: Reserved 1010-1111: Reserved ***Note:*** The output of this processing is referred to as Mapped CLKREQ#.

Bit Range	Default and Access	Field Name (ID): Description
19:16	4h RW	**CLKRQ# Select for CLKOUT_SRC4 (CRQSELSRC4):** Select version of external input CLKRQ# for dynamic control of the output CLKOUT_SRC4. 0000: SRCCLKREQB_0 controls CLKOUT_SRC4 0001: SRCCLKREQB_1 controls CLKOUT_SRC4 0010: SRCCLKREQB_2 controls CLKOUT_SRC4 0011: SRCCLKREQB_3 controls CLKOUT_SRC4 0100: SRCCLKREQB_4 controls CLKOUT_SRC4 - Default 0101: SRCCLKREQB_5 controls CLKOUT_SRC4 0110: SRCCLKREQB_6 controls CLKOUT_SRC4 0111: SRCCLKREQB_7 controls CLKOUT_SRC4 1000: Reserved 1001: Reserved 1010-1111: Reserved ***Note:*** The output of this processing is referred to as Mapped CLKREQ#.
15:12	3h RW	**CLKRQ# Select for CLKOUT_SRC3 (CRQSELSRC3):** Select version of external input CLKRQ# for dynamic control of the output CLKOUT_SRC3. 0000: SRCCLKREQB_0 controls CLKOUT_SRC3 0001: SRCCLKREQB_1 controls CLKOUT_SRC3 0010: SRCCLKREQB_2 controls CLKOUT_SRC3 0011: SRCCLKREQB_3 controls CLKOUT_SRC3 - Default 0100: SRCCLKREQB_4 controls CLKOUT_SRC3 0101: SRCCLKREQB_5 controls CLKOUT_SRC3 0110: SRCCLKREQB_6 controls CLKOUT_SRC3 0111: SRCCLKREQB_7 controls CLKOUT_SRC3 1000-1111: Reserved ***Note:*** The output of this processing is referred to as Mapped CLKREQ#.
11:8	2h RW	**CLKRQ# Select for CLKOUT_SRC2 (CRQSELSRC2):** Select version of external input CLKRQ# for dynamic control of the output CLKOUT_SRC2. 0000: SRCCLKREQB_0 controls CLKOUT_SRC2 0001: SRCCLKREQB_1 controls CLKOUT_SRC2 0010: SRCCLKREQB_2 controls CLKOUT_SRC2 - Default 0011: SRCCLKREQB_3 controls CLKOUT_SRC2 0100: SRCCLKREQB_4 controls CLKOUT_SRC2 0101: SRCCLKREQB_5 controls CLKOUT_SRC2 0110: SRCCLKREQB_6 controls CLKOUT_SRC2 0111: SRCCLKREQB_7 controls CLKOUT_SRC2 1000-1111: Reserved ***Note:*** The output of this processing is referred to as Mapped CLKREQ#.
7:4	1h RW	**CLKRQ# Select for CLKOUT_SRC1 (CRQSELSRC1):** Select version of external input CLKRQ# for dynamic control of the output CLKOUT_SRC1. 0000: SRCCLKREQB_0 controls CLKOUT_SRC1 0001: SRCCLKREQB_1 controls CLKOUT_SRC1 - Default 0010: SRCCLKREQB_2 controls CLKOUT_SRC1 0011: SRCCLKREQB_3 controls CLKOUT_SRC1 0100: SRCCLKREQB_4 controls CLKOUT_SRC1 0101: SRCCLKREQB_5 controls CLKOUT_SRC1 0110: SRCCLKREQB_6 controls CLKOUT_SRC1 0111: SRCCLKREQB_7 controls CLKOUT_SRC1 1000-1111: Reserved ***Note:*** The output of this processing is referred to as Mapped CLKREQ#.
3:0	0h RW	**CLKRQ# Select for CLKOUT_SRC0 (CRQSELSRC0):** Select version of external input CLKRQ# for dynamic control of the output CLKOUT_SRC0. 0000: SRCCLKREQB_0 controls CLKOUT_SRC0 - Default 0001: SRCCLKREQB_1 controls CLKOUT_SRC0 0010: SRCCLKREQB_2 controls CLKOUT_SRC0 0011: SRCCLKREQB_3 controls CLKOUT_SRC0 0100: SRCCLKREQB_4 controls CLKOUT_SRC0 0101: SRCCLKREQB_5 controls CLKOUT_SRC0 0110: SRCCLKREQB_6 controls CLKOUT_SRC0 0111: SRCCLKREQB_7 controls CLKOUT_SRC0 1000-1111: Reserved ***Note:*** The output of this processing is referred to as Mapped CLKREQ#.

8.7.7 CLKREQ Mapping to CLKOUT_SRC Register 2 (CKRQMAPSRC2)—Offset 1028h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 76543210h

31	28	24	20	16	12	8	4	0
0 1 1 1	0 1 1 0	0 1 0 1	0 1 0 0	0 0 1 1	0 0 1 0	0 0 0 1	0 0 0 0	
CRQSELSRC15	CRQSELSRC14	CRQSELSRC13	CRQSELSRC12	CRQSELSRC11	CRQSELSRC10	CRQSELSRC9	CRQSELSRC8	

Bit Range	Default and Access	Field Name (ID): Description
31:28	7h RW	**CLKRQ# Select for CLKOUT_SRC15 (CRQSELSRC15):** Select version of external input CLKRQ# for dynamic control of the output CLKOUT_SRC15. 0000: SRCCLKREQB_8 controls CLKOUT_SRC15 0001: SRCCLKREQB_9 controls CLKOUT_SRC15 0010: SRCCLKREQB_10 controls CLKOUT_SRC15 0011: SRCCLKREQB_11 controls CLKOUT_SRC15 0100: SRCCLKREQB_12 controls CLKOUT_SRC15 0101: SRCCLKREQB_13 controls CLKOUT_SRC15 0110: SRCCLKREQB_14 controls CLKOUT_SRC15 0111: SRCCLKREQB_15 controls CLKOUT_SRC15 - Default 1000-1111: Reserved ***Note:*** The output of this processing is referred to as Mapped CLKREQ#.
27:24	6h RW	**CLKRQ# Select for CLKOUT_SRC14 (CRQSELSRC14):** Select version of external input CLKRQ# for dynamic control of the output CLKOUT_SRC14. 0000: SRCCLKREQB_8 controls CLKOUT_SRC14 0001: SRCCLKREQB_9 controls CLKOUT_SRC14 0010: SRCCLKREQB_10 controls CLKOUT_SRC14 0011: SRCCLKREQB_11 controls CLKOUT_SRC14 0100: SRCCLKREQB_12 controls CLKOUT_SRC14 0101: SRCCLKREQB_13 controls CLKOUT_SRC14 0110: SRCCLKREQB_14 controls CLKOUT_SRC14 - Default 0111: SRCCLKREQB_15 controls CLKOUT_SRC14 1000-1111: Reserved ***Note:*** The output of this processing is referred to as Mapped CLKREQ#.
23:20	5h RW	**CLKRQ# Select for CLKOUT_SRC13 (CRQSELSRC13):** Select version of external input CLKRQ# for dynamic control of the output CLKOUT_SRC13. 0000: SRCCLKREQB_8 controls CLKOUT_SRC13 0001: SRCCLKREQB_9 controls CLKOUT_SRC13 0010: SRCCLKREQB_10 controls CLKOUT_SRC13 0011: SRCCLKREQB_11 controls CLKOUT_SRC13 0100: SRCCLKREQB_12 controls CLKOUT_SRC13 0101: SRCCLKREQB_13 controls CLKOUT_SRC13 - Default 0110: SRCCLKREQB_14 controls CLKOUT_SRC13 0111: SRCCLKREQB_15 controls CLKOUT_SRC13 1000-1111: Reserved ***Note:*** The output of this processing is referred to as Mapped CLKREQ#.

Bit Range	Default and Access	Field Name (ID): Description
19:16	4h RW	**CLKRQ# Select for CLKOUT_SRC12 (CRQSELSRC12):**Select version of external input CLKRQ# for dynamic control of the output CLKOUT_SRC12. 0000: SRCCLKREQB_8 controls CLKOUT_SRC12 0001: SRCCLKREQB_9 controls CLKOUT_SRC12 0010: SRCCLKREQB_10 controls CLKOUT_SRC12 0011: SRCCLKREQB_11 controls CLKOUT_SRC12 0100: SRCCLKREQB_12 controls CLKOUT_SRC12 - Default 0101: SRCCLKREQB_13 controls CLKOUT_SRC12 0110: SRCCLKREQB_14 controls CLKOUT_SRC12 0111: SRCCLKREQB_15 controls CLKOUT_SRC12 1000-1111: Reserved ***Note:*** The output of this processing is referred to as Mapped CLKREQ#.
15:12	3h RW	**CLKRQ# Select for CLKOUT_SRC11 (CRQSELSRC11):** Select version of external input CLKRQ# for dynamic control of the output CLKOUT_SRC11. 0000: SRCCLKREQB_8 controls CLKOUT_SRC11 0001: SRCCLKREQB_9 controls CLKOUT_SRC11 0010: SRCCLKREQB_10 controls CLKOUT_SRC11 0011: SRCCLKREQB_11 controls CLKOUT_SRC11 - Default 0100: SRCCLKREQB_12 controls CLKOUT_SRC11 0101: SRCCLKREQB_13 controls CLKOUT_SRC11 0110: SRCCLKREQB_14 controls CLKOUT_SRC11 0111: SRCCLKREQB_15 controls CLKOUT_SRC11 1000-1111: Reserved ***Note:*** The output of this processing is referred to as Mapped CLKREQ#.
11:8	2h RW	**CLKRQ# Select for CLKOUT_SRC10 (CRQSELSRC10):** Select version of external input CLKRQ# for dynamic control of the output CLKOUT_SRC10. 0000: SRCCLKREQB_8 controls CLKOUT_SRC10 0001: SRCCLKREQB_9 controls CLKOUT_SRC10 0010: SRCCLKREQB_10 controls CLKOUT_SRC10 - Default 0011: SRCCLKREQB_11 controls CLKOUT_SRC10 0100: SRCCLKREQB_12 controls CLKOUT_SRC10 0101: SRCCLKREQB_13 controls CLKOUT_SRC10 0110: SRCCLKREQB_14 controls CLKOUT_SRC10 0111: SRCCLKREQB_15 controls CLKOUT_SRC10 1000-1111: Reserved ***Note:*** The output of this processing is referred to as Mapped CLKREQ#.
7:4	1h RW	**CLKRQ# Select for CLKOUT_SRC9 (CRQSELSRC9):** Select version of external input CLKRQ# for dynamic control of the output CLKOUT_SRC9. 0000: SRCCLKREQB_8 controls CLKOUT_SRC9 0001: SRCCLKREQB_9 controls CLKOUT_SRC9 - Default 0010: SRCCLKREQB_10 controls CLKOUT_SRC9 0011: SRCCLKREQB_11 controls CLKOUT_SRC9 0100: SRCCLKREQB_12 controls CLKOUT_SRC9 0101: SRCCLKREQB_13 controls CLKOUT_SRC9 0110: SRCCLKREQB_14 controls CLKOUT_SRC9 0111: SRCCLKREQB_15 controls CLKOUT_SRC9 1000-1111: Reserved ***Note:*** The output of this processing is referred to as Mapped CLKREQ#.
3:0	0h RW	**CLKRQ# Select for CLKOUT_SRC8 (CRQSELSRC8):** Select version of external input CLKRQ# for dynamic control of the output CLKOUT_SRC8. 0000: SRCCLKREQB_8 controls CLKOUT_SRC8 - Default 0001: SRCCLKREQB_9 controls CLKOUT_SRC8 0010: SRCCLKREQB_10 controls CLKOUT_SRC8 0011: SRCCLKREQB_11 controls CLKOUT_SRC8 0100: SRCCLKREQB_12 controls CLKOUT_SRC8 0101: SRCCLKREQB_13 controls CLKOUT_SRC8 0110: SRCCLKREQB_14 controls CLKOUT_SRC8 0111: SRCCLKREQB_15 controls CLKOUT_SRC8 1000-1111: Reserved ***Note:*** The output of this processing is referred to as Mapped CLKREQ#.

8.7.8 Power Management (PM)—Offset 102Ch

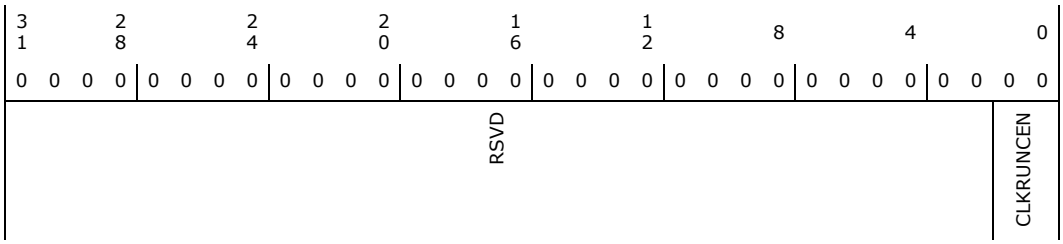
Controls the power management features of clocks.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:2	0h RO	Reserved.
1:0	0h RW	**CLKRUN Control Enable (CLKRUNCEN):** Controls the enabling of support for CLKRUN protocol. 0 - The corresponding CLKOUT_LPC is free-running, unaffected by CLKRUN protocol. 1 - The corresponding CLKOUT_LPC is shut off when CLKRUN protocol turns off LPC clocks.

8.7.9 ICC Debug (ICCDBG)—Offset 1034h

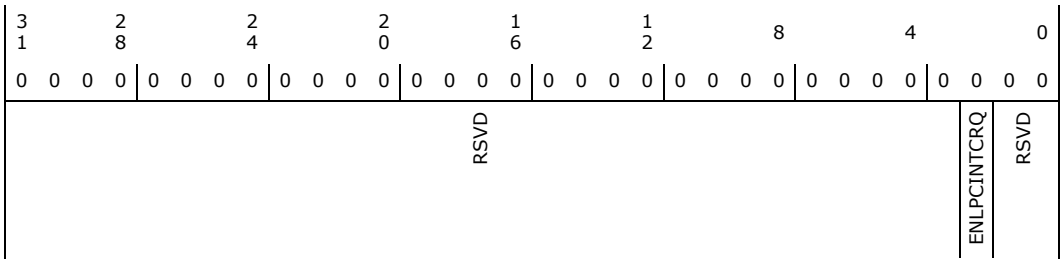
Debug / Chicken Bits for ICC logic

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:3	0h RO	Reserved.
2	0h RW	**Enable LPCINTCLKREQ for CLKOUT_LPCINT and CLKOUT_LPCINTDLY (ENLPCINTCRQ):** Enable dynamic control of CLKOUT_LPCINT and CLKOUT_LPCINTDLY by LPCINTCLKREQ from legacy function. 0: Disable dynamic control of CLKOUT_LPCINT and CLKOUT_LPCINTDLY - Default 1: Enable dynamic control of CLKOUT_LPCINT and CLKOUT_LPCINTDLY When enabled by this register bit, CLKOUT_LPCINT and CLKOUT_LPCINTDLY are subject to gating/ungating control by LPCINTCLKREQ from the legacy function. When disabled by this register bit, these clocks are insensitive to the state of LPCINTCLKREQ from the legacy function, and are always running after ICC initialization. **Implementation Note:** Effective LPCINTCLKREQ = LPCINTCLKREQ + ! ENLPCINTCRQ.
1:0	0h RO	Reserved.

8.7.10 USB3Gen2PCIe PLL Control (G2PLLCTRL)—Offset 2000h

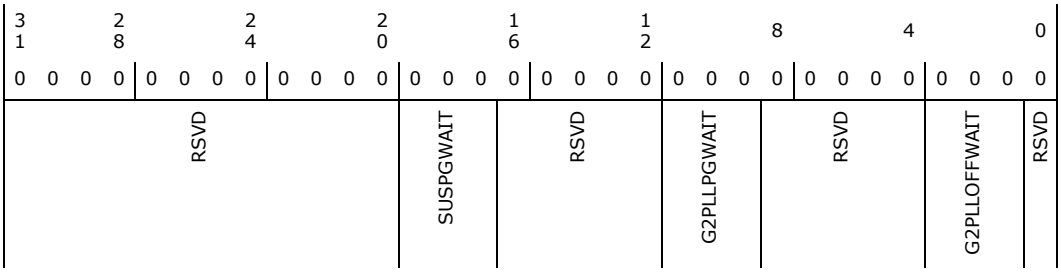
Controls USB3Gen2PCIe PLL and its output clocks behavior. This offset is lockable by setting LOCK_G2PLLC bit (ICCSEC offset 1020h bit 10).

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:20	0h RO	Reserved.
19:17	0h RW	**Run-time S0 SUS PG Wait (SUSPGWAIT):** Upon the USB3Gen2PCIe PLL shutdown and power gated, a timer is started. Any wake event occurring before the timer expires causes the timer to reset to SUSPGWAIT value. Once the timer expires and there are no wake events, the SUS well power to ModPHY USB3Gen2PCIe PLL can be gated. 000: 0 us - Default 001: 1 us 010: 2 us 011: 4 us 100: 8 us 101: 16 us 110: 32 us 111: Reserved
16:12	0h RO	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
11:9	0h RW	**USB3Gen2PCIe PLL PG Wait (G2PLLPGWAIT):** Upon the USB3Gen2PCIe PLL shutdown, a timer is started. Any wake event occurring before the timer expires causes the timer to reset to G2PLLPGWAIT value. Once the timer expires and there are no wake events, the USB3Gen2PCIe PLL can be power gated. 000: 0 us - Default 001: 1 us 010: 2 us 011: 4 us 100: 8 us 101: 16 us 110: 32 us 111: Reserved
8:4	0h RO	Reserved.
3:1	0h RW	**USB3Gen2PCIe PLL OFF Wait (G2PLLOFFWAIT):** Upon the event that all conditions (other than this G2PLLOFFWAIT timer itself) are satisfied for USB3Gen2PCIe PLL shutdown, a timer is started. Any wake event occurring before the timer expires causes the timer to reset to G2PLLOFFWAIT value. Once the timer expires and there are no wake events, the USB3Gen2PCIe PLL can be shutdown. 000: 0 us - Default 001: 1 us 010: 2 us 011: 4 us 100: 8 us 101: 16 us 110: 32 us 111: Reserved
0	0h RO	Reserved.

§

9 Power Management

9.1 Acronyms

Acronyms	Description
PMC	Power Management Controller
STD	Suspend To Disk
STR	Suspend To RAM
EOS	End of SMI
VLW	Virtual Legacy Wire
DSW	Deep SX Well
ADR	Asynchronous DRAM Refresh

9.2 References

Specification	Location
Advanced Configuration and Power Interface, Version 4.0a (ACPI)	http://www.acpi.info/spec.htm

9.3 Overview

The Power Management Controller that handles all PCH power management related activities. This unit administers power management functions of the PCH including interfacing with other logic and controllers on the platform to perform power state transitions (such as SLP_S3# and PLTRST#); configure, manage and respond to wake events; aggregate and report latency tolerance information for devices and peripherals connected to and integrated into the PCH.

9.4 Signal Description

Name	Type	Description
GPD1_ACPRESENT	I	**ACPRESENT:** This input pin indicates when the platform is plugged into AC power or not. The PCH uses this information to implement the Deep Sx policies. For example, the platform may be configured to enter Deep Sx when in S4 or S5 and only when running on battery. This is powered by Deep Sx Well.
GPP_A12_BM_BUSY_N_SXEXITHLDOFF_N	I	**Bus Master Busy:** Generic bus master activity indication driven into the PCH. Can be configured to set the PM1_STS.BM_STS bit. Can also be configured to assert indications transmitted from the PCH to the processor using the PMSYNCH pin.
DSW_PWROK	I	**DSW PWROK**: Power OK Indication for the VCCDSW3_3 voltage rail. When DSW_PWROK goes high, the PCH will drive SLP_SUS# high. Because SLP_SUS# is now defined as needing to be high at least by the time RSMRST# goes high, DSW_PWROK now must lead RSMRST# This signal is in the RTC well.
GPD2_GBE_WAKE_N	I	**GbE WAKE**: is an active low wake indicator from the GbE PHY to the one GbE MAC. ***Note:*** External pull-up required.

Name	Type	Description
GPD11_GBEPHY	O	**GBE PHY Power Control:** This is used to indicate that power needs to be restored to the I219 PHY
PCH_PWROK	I	**PCH Power OK:** When asserted, PCH_PWROK is an indication to the PCH that all of its core power rails have been stable for at least 5 ms. PCH_PWROK can be driven asynchronously. When PCH_PWROK is negated, the PCH asserts PLTRST#. ***Note:*** PCH_PWROK must not glitch, even if RSMRST# is low.
GPP_B13_PLTRST_N	O	**Platform Reset:** The PCH asserts PLTRST# to reset devices on the platform (such as SIO, PCIe devices, LAN, processor, and so forth.). The PCH asserts PLTRST# during power-up and when S/W initiates a hard reset sequence through the Reset Control register (I/O port CF9h). The PCH drives PLTRST# active a minimum of 1 ms when initiated through the Reset Control register (I/O port CF9h). ***Note:*** PCI/PCIe* 2.0 specification requires that the power rails associated with PCI/PCIe* (typically the 3.3V, 5V, and 12V core well rails) have been valid for 100 ms prior to PLTRST# de-assertion. System designers must ensure the requirement is met on the platform.
GPP_A11_PME_N	I / OD	**Power Management Event**: Driven by devices to wake the system or issue SCI.
GPD3_PWRBTN_N	I	**Power Button:** The Power Button will cause SMI# or SCI to indicate a system request to go to a sleep state. If the system is already in a sleep state, this signal will cause a wake event. If PWRBTN# is pressed for more than four seconds, this will cause an unconditional transition (power button override) to the S5 state. Override will occur even if the system is in the S3-S4 states. This signal has an internal pull-up resistor and has an internal 16 ms de-bounce on the input. ***Note:*** Upon entry to S5 due to a power button override, if Deep Sx is enabled and conditions are met, the system will transition to Deep Sx.
RSMRST_N	I	Resume Well Reset: When high, this pin indicates that all power rails associated with the PCH Primary well are valid and stable. This signal must be asserted for at least T201 after the primary wells are valid**.**
GPD6_SLP_A_N	O	**SLP_A#:** Used to control power to the active sleep well. The PCH no longer has an ASW well, but there could still be one on the board. ***Note:*** There is no corresponding APWROK signal input to the PCH, but the PCH does have an internally generated version of APWROK that is timed from SLP_A#.
SLP_GBE_N	O	**GBE Sub-System Sleep Control:** When SLP_GBE# is de-asserted it indicates that the PHY device must be powered. When SLP_GBE# is asserted, power can be shut off to the PHY device. SLP_GBE# will always be de-asserted in S0 and anytime SLP_A# is de-asserted.
GPP_B12_GLB_RST_WARN_N	O	**Global Reset Warn**: An early indication that the PCH has detected a condition that will cause it to signal the system to perform a global reset.
GPD_D4_SLP_S3_N	O	**S3 Sleep Control:** SLP_S3# is for power plane control. This signal shuts off power to all non-critical systems when in S3 (Suspend To RAM), S4 (Suspend to Disk), or S5 (Soft Off) states.
GPD_D5_SLP_S4_N	O	**S4 Sleep Control**: SLP_S4# is for power plane control. This signal shuts power to all non-critical systems when in the S4 (Suspend to Disk) or S5 (Soft Off) state. ***Note:*** This pin must be used to control the DRAM power in order to use the PCH DRAM power-cycling feature.
GPD_D10_SLP_S5_N	O	**S5 Sleep Control:** SLP_S5# is for power plane control. This signal is used to shut power off to all non-critical systems when in the S5 (Soft Off) states.
SLP_SUS_N	O	**Deep Sx Indication:** When asserted (driven low), this signal indicates PCH is in Deep Sx state where internal Sus power is shut off for enhanced power saving. When de-asserted (driven high), this signal indicates exit from Deep Sx state and Sus power can be applied to PCH. If Deep Sx is not supported, then this pin can be left unconnected. ***Note:*** This pin is in the DSW power well.

Name	Type	Description
GPP_A15_SUSACK_N	I	**SUSACK#:** If Deep Sx is supported, the BMC/motherboard controlling logic must change SUSACK# to match SUSWARN# once the BMC/motherboard controlling logic has completed the preparations discussed in the description for the SUSWARN# pin. ***Note:*** SUSACK# is only required to change in response to SUSWARN# if Deep Sx is supported by the platform.
GPD_SUSCLK	O	**Suspend Clock:** This clock is an output of the RTC generator circuit to use by other chips for refresh clock.
GPP_A13_SUSWARN_N_SUSWRDNACK	O	**SUSWARN#:** This pin asserts low when the PCH is planning to enter the Deep Sx power state and remove Primary power (using SLP_SUS#). The BMC/motherboard controlling logic must observe edges on this pin, preparing for SUS well power loss on a falling edge and preparing for Primary well related activity (host/Intel ME wakes and runtime events) on a rising edge. SUSACK# must be driven to match SUSWARN# once the above preparation is complete. SUSACK# should be asserted within a minimal amount of time from SUSWARN# assertion as no wake events are supported if SUSWARN# is asserted but SUSACK# is not asserted. Platforms supporting Deep Sx, but not wishing to participate in the handshake during wake and Deep Sx entry may tie SUSACK# to SUSWARN#. This pin is multiplexed with SUSPWRDNACK since it is not needed in Deep Sx supported platforms.
GPP_A13_SUSWARN_N_SUSWRDNACK	O	**SUSPWRDNACK:** Active high. Asserted by the PCH on behalf of the Intel ME when it does not require the PCH Primary well to be powered. Platforms are not expected to use this signal when the PCH Deep Sx feature is used.
GPP_A12_BMBUSY_N_SXEXITHLDOFF_N	I	**Sx Exit Holdoff Delay**: Delay exit from Sx state after SLP_A# is de-asserted. In LPC mode, the SPI bus will be tristated so that an external device can access the SPI component. In the eSPI/LPC interface is in eSPI mode this input has no effect.
SYS_PWROK	I	**System Power OK:** This generic power good input to the PCH is driven and utilized in a platform-specific manner. While PCH_PWROK always indicates that the core wells of the PCH are stable, SYS_PWROK is used to inform the PCH that power is stable to some other system component(s) and the system is ready to start the exit from reset.
SYS_RESET_N	I	**System Reset**: This pin forces an internal reset after being de-bounced. The PCH will reset immediately if the SMBus is idle; otherwise, it will wait up to 25 ms ±2 ms for the SMBus to idle before forcing a reset on the system.
WAKE_N	I / OD	**PCI Express Wake Event in Sx:** Input Pin in Sx. Sideband wake signal on PCI Express asserted by components requesting wake up. During S0, this pin functions as an OBFF indicator (open drain in S0). ***Note:*** External pull-up required.
GPP_A8_CLKRUN_N	I / OD	**LPC Clock Run:** Used to control CLKOUT_LPC[2:1]. Connects to peripherals that need to request clock restart or prevention of clock stopping.
GPP_K10_PE_RST_N	I	**PE_RST_N:** Reset signal into the QAT and 10/1 GbE functions. This signal must be driven/connected any time QAT or 10/1GbE functionality is enabled.
GPP_B0_CORE_VID_0, GPP_B1_CORE_VID_1	OD	**CORE VID 0/1**: OD outputs that are used to encode one of 4 possible voltages that the Vcc_AVID plane should be running at. This is SKU dependent. If AVID is not supported on a SKU, the these pins will default to 11b (with external pull-up resistors).
GPP_I4_DO_RESET_IN_N	I	**DO_RESET_IN_N**: Used by non-legacy PCHs. The DO_RESET_OUT_N from the legacy PCH drives this signal. Default for this pin is GPIO, so it is not expected to have this native function on the pin unless multi-PCH functionality is being explicitly supported.
GPP_I5_DO_RESET_OUT_N	IOD	**DO_RESET_OUT_N**: Signal from the legacy PCH to the non-legacy PCH(s)used to coordinate power up and power down sequencing between the PCHs. Default for this pin is GPIO, so it is not expected to have this native function on the pin unless multi-PCH functionality is being explicitly supported. Default for this pin is GPIO, so it is not expected to have this native function on the pin unless multi-PCH functionality is being explicitly supported.

Name	Type	Description
GPP_I6_RESET_DONE	IOD	**RESET_DONE**: Output of the non-legacy PCHs and input to the Legacy PCH. Used to indicate that the non-legacy PCHs have accomplished their appropriate reset sequences and the legacy PCH can continue on.
GPP_G17_ADR_COMPLETE	O	Active when the ADR timer expires to indicate that the self refresh is complete, or that the memory can now be put in battery backup mode.

9.5 Integrated Pull-ups and Pull-downs

Signal	Resistor Type	Value	Notes
GPD1_ACPRESENT	Pull-down	15 kΩ - 40 kΩ	1
GPD2_GBE_WAKE_N	Pull-down	15 kΩ - 40 kΩ	1
GPD3_PWRBTN_N	Pull-up	15 kΩ - 40 kΩ	
GPP_A11_PME_N	Pull-up	15 kΩ - 40 kΩ	
GPP_A15_SUSACK_N	Pull-up	15 kΩ - 40 kΩ	
WAKE_N	Pull-down	15 kΩ - 40 kΩ	1, 2

Notes:
1. Pull-down is configurable and can be enabled in Deep Sx state; refer to DSX_CFG register (RCBA+3334h) for more details.
2. Open drain when driven as OBFF indication
3. Non-DSW pull-ups and pull-downs are not valid until all voltages are valid.

9.6 I/O Signal Planes and States

Signal Name	Power Plane	During Reset	Immediately after Reset	S3/S4/S5	Deep Sx
BMBUSY_N	Primary	Undriven	Undriven	Undriven	Off
RSMRST_N	RTC	Undriven	Undriven	Undriven	Undriven
PCH_PWROK	RTC	Undriven	Undriven	Undriven	Undriven
SYS_PWROK	Primary	Undriven	Undriven	Undriven	Off
DSW_PWROK	RTC	Undriven	Undriven	Undriven	Undriven
GLB_RST_WARN_N[1,6]	Primary	Driven High	Driven High	Driven High	Off
SLP_S3_N[6]	DSW	Driven Low	Driven High	Driven Low	Driven Low
SLP_S4_N[6]	DSW	Driven Low	Driven High	Driven High/ Driven Low[2]	Driven High/ Driven Low[9]
SLP_S5_N[6]	DSW	Driven Low	Driven High	Driven High/ Driven Low[3]	Driven High/ Driven Low[9]
SLP_GBE_N[6]	DSW	Driven Low	Driven Low	Driven High/ Driven Low[7]	Driven High/ Driven Low[7]
SLP_A_N[6]	DSW	Driven Low	Driven High	Driven High/ Driven Low[12]	Driven High/ Driven Low[12]
SLP_SUS_N[6]	DSW	Driven Low	Driven High	Driven High	Driven Low
SUSCLK[10]	DSW	Driven Low	Toggling	Toggling	Toggling[10]
SUSWARN_N/ SUSPWRDNACK[6,10]	Primary	Driven Low	Driven Low	Driven Low[5]	Off
SUSACK_N	Primary	Internal Pull-up	Internal Pull-up	Internal Pull-up	Off
ACPRESENT_N[6,10]	DSW	Undriven / Driven Low[4]	Undriven	Undriven	Undriven / Driven Low[8]

Signal Name	Power Plane	During Reset	Immediately after Reset	S3/S4/S5	Deep Sx
WAKE_N	DSW	Undriven	Undriven	Undriven	Undriven / Driven Low[8]
GBE_WAKE_N	DSW	Undriven	Undriven	Undriven	Undriven / Driven Low[8]
GPD11_GBEPHY[10]	DSW	Driven Low	Driven Low	Driven Low	Driven Low
PME_N	Primary	Internal Pull-up	Internal Pull-up	Internal Pull-up	Off
PWRBTN_N	DSW	Internal Pull-up	Internal Pull-up	Internal Pull-up	Internal Pull-up
SYS_RESET_N	Primary	Undriven	Undriven	Undriven	Off
PLTRST_N	Primary	Driven Low	Driven High	Driven Low	Off
SXEXITHLDOFF_N	Primary	Undriven	Undrive	Undriven	Off
ADR_COMPLETE	Primary	0	0/1	0/1	Off

Notes:
1. Driven High during S0 and driven Low during S0 CS.
2. SLP_S4# is driven high in S3, driven low in S4/S5.
3. SLP_S5# is driven high in S3/S4, driven low in S5.
4. In non-Deep Sx mode, pin is driven low.
5. Based on wake events and ME state. SUSPWRDNACK is always '0' while in M0 or M3, but can be driven to '0' or '1' while in Moff state. SUSPWRDNACK is the default mode of operation. If Deep Sx is supported, then subsequent boots will default to SUSWARN#.
6. The pin requires glitch-free output sequence. The pad should only be pulled low momentarily when the corresponding buffer power supply is not stable.
7. Based on wake event and ME state.
8. Pull-down is configurable and can be enabled in Deep Sx state; refer to DSX_CFG register (RCBA+3334h) for more details.
9. When platform enters Deep Sx, the SLP_S4# and SLP_S5# pin will retain the value it held prior to Deep Sx entry.
10. Internal weak pull resistor is default off but configurable (pu/pd/none) after boot.
11. The CORE_VID pin output can be 0 or 1 depending on fuse.
12. Pin state is a function of whether the platform is configured to have ME on or off in Sx.
13. Signal names have been shortened to just the native functions.

9.7 Functional Description

9.7.1 Features

- Support for *Advanced Configuration and Power Interface, Version 4.0a* providing power and thermal management
  - — ACPI 24-Bit Timer SCI and SMI# Generation
- PCI PME# signal for wake up from low-power states
- System Sleep State Control
  - — ACPI S3 state – Suspend to RAM
  - — ACPI S4 state – Suspend-to-Disk
  - — ACPI G2/S5 state – Soft Off
  - — Power Failure Detection and Recovery
  - — Deep Sx
- Intel Management Engine Power Management Support
  - — Wake events from the Intel Management Engine (enabled from all S-States including catastrophic S5 conditions)

- SLP_S0# signal for external platform VR power gating or EC power management handling during lower power condition

9.7.2 PCH and System Power States

Table 9-1 shows the power states defined for PCH-based platforms. The state names generally match the corresponding ACPI states.

Table 9-1. General Power States for Systems Using the PCH

State/ Substates	Legacy Name/Description
G0/S0/C0	**Full On**: Processor operating. Individual devices may be shut down or be placed into lower power states to save power.
G0/S0/Cx	**Cx State**: Cx states are processor power states within the S0 system state that provide for various levels of power savings. The processor manages c-state itself. The actual c-state is not passed to the PCH. Only c-state related messages are sent to the PCH and PCH will base its behavior on the actual data passed.
G1/S3	**Suspend-To-RAM:** The system context is maintained in system DRAM, but power is shut off to non-critical circuits. Memory is retained and refreshes continue. All external clocks stop except RTC.
G1/S4	**Suspend-To-Disk:** The context of the system is maintained on the disk. All power is then shut off to the system except for the logic required to resume.
G2/S5	**Soft Off:** System context is not maintained. All power is shut off except for the logic required to restart. A full boot is required when waking.
Deep Sx	**Deep Sx:** An optional low power state where system context may or may not be maintained depending upon entry condition. All power is shut off except for minimal logic that allows exiting Deep Sx. If Deep Sx state was entered from S3 state, then the resume path will place system back into S3. If Deep Sx state was entered from S4 state, then the resume path will place system back into S4. If Deep Sx state was entered from S5 state, then the resume path will place system back into S5.
G3	**Mechanical OFF:** System context not maintained. All power is shut off except for the RTC. No "Wake" events are possible. This state occurs if the user turns off a mechanical switch, or if the system power supply is at a level that is insufficient to power the "waking" logic. When system power returns, transition will depend on the state just prior to the entry to G3 and the AFTERG3_EN bit in the GEN_PMCON_3 register (D31:F0, offset A4). Refer to Table 9-7 for more details.

Table 9-2 shows the transitions rules among the various states.

Note: Transitions among the various states may appear to temporarily transition through intermediate states. For example, in going from S0 to S4, it may appear to pass through the G1/S3 state. These intermediate transitions and states are not listed in the Table 9-2.

Table 9-2. State Transition Rules for the PCH (Sheet 1 of 2)

Present State	Transition Trigger	Next State
G0/S0/C0	• OPI Msg • SLP_EN bit set • Power Button Override[2,5] • Mechanical Off/Power Failure	• G0/S0/Cx • G1/Sx or G2/S5 state • G2/S5 • G3
G0/S0/Cx	• OPI Msg • Power Button Override[2,5] • Mechanical Off/Power Failure	• G0/S0/C0 • S5 • G3
G1/S3	• Any Enabled Wake Event • Power Button Override[2,5] • Conditions met as described in Section 9.7.7.6.1 and Section 9.7.7.6.2 • Mechanical Off/Power Failure	• G0/S0/C0[2] • G2/S5 • Deep Sx • G3

Table 9-2. State Transition Rules for the PCH (Sheet 2 of 2)

Present State	Transition Trigger	Next State
G1/S4	• Any Enabled Wake Event • Power Button Override[2,5] • Conditions met as described in Section 9.7.7.6.1 and Section 9.7.7.6.2 • Mechanical Off/Power Failure	• G0/S0/C0[2] • G2/S5 • Deep Sx • G3
G2/S5	• Any Enabled Wake Event • Conditions met as described in Section 9.7.7.6.1 and Section 9.7.7.6.2 • Mechanical Off/Power Failure	• G0/S0/C0[2] • Deep Sx • G3
G2/Deep Sx	• Any Enabled Wake Event • ACPRESENT Assertion • Mechanical Off/Power Failure	• G0/S0/C0[2] • G1/S3, G1/S4 or G2/S5 (see Section 9.7.7.6.2) • G3
G3	• Power Returns	• S0/C0 (reboot) or G2/S5[3] (stay off until power button pressed or other wake event)[1,2]

Notes:
1. Some wake events can be preserved through power failure.
2. Includes all other applicable types of events that force the host into and stay in G2/S5.
3. If the system was in G1/S4 before G3 entry, then the system will go to S0/C0 or G1/S4.
4. Upon entry to S5 due to a power button override, if Deep Sx is enabled and conditions are met per Section 9.7.7.6, the system will transition to Deep Sx.

9.7.3 System Power Planes

The system has several independent power planes, as described in Table 9-3.

Note: When a particular power plane is shut off, it should go to a 0V level.

Table 9-3. System Power Plane (Sheet 1 of 2)

Plane	Controlled By	Description
Processor	SLP_S3# signal	The SLP_S3# signal can be used to cut the power to the processor completely.
Main (Applicable to Platform, PCH does not have a Main well)	SLP_S3# signal	When SLP_S3# goes active, power can be shut off to any circuit not required to wake the system from the S3 state. Since the S3 state requires that the memory context be preserved, power must be retained to the main memory. The processor, devices on the PCI bus, LPC I/F, and PCI Express will typically be power-gated when the Main power plane is shut, although there may be small subsections powered. ***Note:*** The PCH power id not controlled by the SLP_S3# signal, but instead by the SLP_SUS# signal.
Memory	SLP_S4# signal SLP_S5# signal	When SLP_S4# goes active, power can be shut off to any circuit not required to wake the system from the S4. Since the memory context does not need to be preserved in the S4 state, the power to the memory can also be shut down. When SLP_S5# goes active, power can be shut off to any circuit not required to wake the system from the S5 state. Since the memory context does not need to be preserved in the S5 state, the power to the memory can also be shut.
Intel ME	SLP_A#	SLP_A# signal is asserted when the ME platform goes to M-Off. Depending on the platform, this pin may be used to control power to various devices that are part of the ME sub-system in the platform.
LAN	SLP_GBE#	This signal is asserted in Sx/M-Off when both host and Intel ME WoL are not supported. This signal can be use to control power to the Intel GbE PHY.

Table 9-3. System Power Plane (Sheet 2 of 2)

Plane	Controlled By	Description
Suspend Well	SLP_SUS#	This signal is asserted when the Sus rails can be externally shut off for enhanced power saving.
DEVICE[n]	Implementation Specific	Individual subsystems may have their own power plane. For example, GPIO signals may be used to control the power to disk drives, audio amplifiers, or the display screen.

9.7.4 SMI#/SCI Generation

Upon any enabled SMI event taking place while the End of SMI bit is set, the PCH will clear the EOS bit and assert SMI to the processor, which will cause it to enter SMM space. SMI assertion is performed using a Virtual Legacy Wire message. Prior system generations (those based upon legacy processors) used an actual SMI# pin.

Once the SMI VLW has been delivered, the PCH takes no action on behalf of active SMI events until Host software sets the EOS bit. At that point, if any SMI events are still active, the PCH will send another SMI VLW message.

The SCI is a level-mode interrupt that is typically handled by an ACPI-aware operating system. In non-APIC systems (which is the default), the SCI IRQ is routed to one of the 8259 interrupts (IRQ 9, 10, or 11). The 8259 interrupt controller must be programmed to level mode for that interrupt.

In systems using the APIC, the SCI can be routed to interrupts 9, 10, 11, 20, 21, 22, or 23. The interrupt polarity changes depending on whether it is on an interrupt shareable with a PIRQ or not. The interrupt remains asserted until all SCI sources are removed.

Table 9-4 shows which events can cause an SMI and SCI.

Note: Some events can be programmed to cause either an SMI or SCI. The usage of the event for SCI (instead of SMI) is typically associated with an ACPI-based system. Each SMI or SCI source has a corresponding enable and status bit.

Table 9-4. Causes of SMI and SCI (Sheet 1 of 3)

Cause	SCI	SMI	Additional Enables (Note 1)	Where Reported
PME#	Yes	Yes	PME_EN=1	PME_STS
PME_B0 (Internal, Bus 0, PME-Capable Agents)	Yes	Yes	PME_B0_EN=1	PME_B0_STS
PCI Express* PME Messages	Yes	Yes	PCI_EXP_EN=1 (Not enabled for SMI)	PCI_EXP_STS
PCI Express Hot-Plug Message	Yes	Yes	HOT_PLUG_EN=1 (Not enabled for SMI)	HOT_PLUG_STS
Power Button Press	Yes	Yes	PWRBTN_EN=1	PWRBTN_STS
Power Button Override (Note 6)	Yes	No	None	PRBTNOR_STS
RTC Alarm	Yes	Yes	RTC_EN=1	RTC_STS
ACPI Timer overflow (2.34 seconds)	Yes	Yes	TMROF_EN=1	TMROF_STS
GPIO (Note 8)	Yes	Yes		
GBE_WAKE#	Yes	Yes	LAN_WAKE#_EN=1	LAN_WAKE#_STS
TCO SCI message from processor	Yes	No	None	TCOSCI_STS
TCO SCI Logic	Yes	No	TCOSCI_EN=1	TCOSCI_STS
TCO SMI Logic	No	Yes	TCO_EN=1	TCO_STS

Table 9-4. Causes of SMI and SCI (Sheet 2 of 3)

Cause	SCI	SMI	Additional Enables (Note 1)	Where Reported
TCO SMI –	No	Yes	None	NEWCENTURY_STS
TCO SMI – TCO TIMEROUT	No	Yes	None	TIMEOUT
TCO SMI – OS writes to TCO_DAT_IN register	No	Yes	None	SW_TCO_SMI
TCO SMI – Message from processor	No	Yes	None	OPISMI_STS
TCO SMI – NMI occurred (and NMIs mapped to SMI)	No	Yes	NMI2SMI_EN=1	NMI2SMI_STS
TCO SMI – INTRUDER# signal goes active	No	Yes	INTRD_SEL=10	INTRD_DET
TCO SMI – Change of the BIOSWE (D31:F0:DCh, Bit 0) bit from 0 to 1	No	Yes	BLE=1	BIOSWR_STS
TCO SMI – Write attempted to BIOS	No	Yes	BIOSWE=1	BIOSWR_STS
BIOS_RLS written to 1 (Note 7)	Yes	No	GBL_EN=1	GBL_STS
GBL_RLS written to	No	Yes	BIOS_EN=1	BIOS_STS
Write to B2h register	No	Yes	APMC_EN = 1	APM_STS
Periodic timer expires	No	Yes	PERIODIC_EN=1	PERIODIC_STS
64 ms timer expires	No	Yes	SWSMI_TMR_EN=1	SWSMI_TMR_STS
Enhanced USB Legacy Support Event	No	Yes	LEGACY_USB2_EN = 1	LEGACY_USB2_STS
Serial IRQ SMI reported	No	Yes	None	SERIRQ_SMI_STS
Device monitors match address in its range	No	Yes	None	DEVTRAP_STS
SMBus Host Controller	No	Yes	SMB_SMI_EN Host Controller Enabled	SMBus host status reg.
SMBus Slave SMI message	No	Yes	None	SMBUS_SMI_STS
SMBus SMBALERT# signal active	No	Yes	None	SMBUS_SMI_STS
SMBus Host Notify message received	No	Yes	HOST_NOTIFY_INTREN	SMBUS_SMI_STS HOST_NOTIFY_STS
Access microcontroller 62h/66h	No	Yes	MCSMI_EN	MCSMI_STS
SLP_EN bit written to 1	No	Yes	SLP_SMI_EN=1	SLP_SMI_STS
SPI Command Completed	No	Yes	None	SPI_STS
eSPI SCI/SMI Request	Yes	Yes	eSPI_SCI_EN See eSPI section.	eSPI_SCI_STS eSPI_SMI_STS
Software Generated GPE	Yes	Yes	SWGPE_EN=1	SWGPE_STS
ME	Yes	Yes	ME_SCI_EN=1 ME_SCI_EN=0; ME_SMI_EN=1;	ME_SCI_STS ME_SMI_STS
GPIO Lockdown Enable bit changes from '1' to '0'	No	Yes	GPIO_UNLOCK_SMI_EN=1	GPIO_UNLOCK_SMI_STS
USB3 (xHCI) SMI Event	No	Yes	XHCI_SMI_EN=1	XHCI_SMI_STS
Wake Alarm Device Timer	Yes	Yes	WADT_EN	WADT_STS

Table 9-4. Causes of SMI and SCI (Sheet 3 of 3)

Cause	SCI	SMI	Additional Enables (Note 1)	Where Reported

Notes:
1. SCI_EN must be 1 to enable SCI, except for BIOS_RLS. SCI_EN must be 0 to enable SMI.
2. SCI can be routed to cause interrupt 9:11 or 20:23 (20:23 only available in APIC mode).
3. GBL_SMI_EN must be 1 to enable SMI.
4. EOS must be written to 1 to re-enable SMI for the next 1.
5. The PCH must have SMI fully enabled when the PCH is also enabled to trap cycles. If SMI is not enabled in conjunction with the trap enabling, then hardware behavior is undefined.
6. When a power button override first occurs, the system will transition immediately to S5. The SCI will only occur after the next wake to S0 if the residual status bit (PRBTNOR_STS) is not cleared prior to setting SCI_EN.
7. GBL_STS being set will cause an SCI, even if the SCI_EN bit is not set. Software must take great care not to set the BIOS_RLS bit (which causes GBL_STS to be set) if the SCI handler is not in place.
8. Refer to GPIO chapter for specific GPIOs enabled for SCIs and/or SMIs.

9.7.4.1 PCI Express SCI

PCI Express ports and the processor have the ability to cause a Power Management Event (PME) using messages. When a PME message is received, the PCH will set the PCI_EXP_STS bit. If the PCI_EXP_EN bit is also set, the PCH can cause an SCI using the GPE1_STS register.

9.7.4.2 PCI Express Hot-Plug

PCI Express has a hot-plug mechanism and is capable of generating a SCI using the GPE1 register. It is also capable of generating an SMI. However, it is not capable of generating a wake event. This is not a surprise hot plug.

9.7.5 C-States

PCH-based systems implement C-states by having the processor control the states. The chipset exchanges messages with the processor as part of the C-state flow, but the chipset does not directly control any of the processor impacts of C-states, such as voltage levels or processor clocking. In addition to the messages, the PCH also provides additional information to the processor using a sideband pin (PM_SYNC, PM_SYNC2).

9.7.6 Dynamic 24 MHz Clock Control

The 24 MHz clock can be dynamically controlled independent of any other low-power state. The Dynamic 24 MHz Clock control is handled using the CLKRUN# signal.

9.7.6.1 Conditions for Checking the 24 MHz Clock

When there is a lack of activity, the PCH has the capability to stop the 24 MHz clocks to conserve power. “Clock activity” is defined as any activity that would require the 24 MHz clock to be running.

Any of the following conditions will indicate that it is **not okay** to stop the 24 MHz clock:

- Cycles on LPC
- SERIRQ activity

9.7.6.2 Conditions for Maintaining the 24 MHz Clock

LPC or any other devices that wish to maintain the 24 MHz clock running will observe the CLKRUN# signal de-asserted, and then must re-assert it (drive it low) within 92 clocks.

- When the PCH has tri-stated the CLKRUN# signal after de-asserting it, the PCH then checks to see if the signal has been re-asserted (externally).
- After observing the CLKRUN# signal asserted for one clock, the PCH again starts asserting the signal.
- If an internal device needs the PCI bus, the PCH asserts the CLKRUN# signal.

9.7.6.3 Conditions for Stopping the 24 MHz Clock

- When there is a lack of activity (as defined above) for ninety 24 MHz clock cycles, the PCH de-asserts (drive high) CLKRUN# for one clock and then tri-states the signal.
- If no device drives CLKRUN# low within 93 clock cycles after it has been de-asserted, the PCH will stop the 24 MHz clocks.

9.7.6.4 Conditions for Re-starting the 24 MHz Clock

- A peripheral asserts CLKRUN# to indicate that it needs the 24 MHz clock re-started.
- Observing the CLKRUN# signal asserted externally for one (free running) clock, the PCH again starts driving CLKRUN# asserted.

If an internal source requests the clock to be re-started, the PCH re-asserts CLKRUN#, then the PCH will start the 24 MHz clocks.

9.7.7 Sleep States

9.7.7.1 Sleep State Overview

The PCH directly supports different sleep states (S1–S5), which are entered by methods such as setting the SLP_EN bit or due to a Power Button press. The entry to the Sleep states is based on several assumptions:

- The G3 state cannot be entered using any software mechanism. The G3 state indicates a complete loss of power.

9.7.7.2 Initiating Sleep State

Sleep states (S1–S5) are initiated by:

- Masking interrupts, turning off all bus master enable bits, setting the desired type in the SLP_TYP field, and then setting the SLP_EN bit. The hardware then attempts to gracefully put the system into the corresponding Sleep state.
- Pressing the PWRBTN# Signal for more than four seconds to cause a power button Override event. In this case the transition to the S5 state is less graceful, since there are no dependencies on OPI messages from the processor or on clocks other than the RTC clock.
- Assertion of the THERMTRIP# signal will cause a transition to the S5 state. This can occur when system is in S0 or S1 state.
- Shutdown by integrated manageability functions (ASF/Intel AMT)
- Internal watchdog timer timeout events

- SMBus unconditional shutdown

Table 9-5. Sleep Types

Sleep Type	Comment
S3	The PCH asserts SLP_S3#. The SLP_S3# signal controls the power to non-critical circuits. Power is only retained to devices needed to wake from this sleeping state, as well as to the memory.
S4	The PCH asserts SLP_S3# and SLP_S4#. The SLP_S4# signal shuts off the power to the memory subsystem. Only devices needed to wake from this state should be powered.
S5	The PCH asserts SLP_S3#, SLP_S4# and SLP_S5#.

9.7.7.3 Exiting Sleep States

Sleep states (S1–S5) are exited based on wake events. The wake events forces the system to a full on state (S0), although some non-critical subsystems might still be shut off and have to be brought back manually. For example, the hard disk may be shut off during a sleep state and have to be enabled using a GPIO pin before it can be used.

Upon exit from the PCH-controlled sleep states, the WAK_STS bit is set. The possible causes of wake events (and their restrictions) are shown in Table 9-6.

Table 9-6. Causes of Wake Events (Sheet 1 of 2)

Cause	How Enabled	Wake from Sx	Wake from Deep Sx	Wake from Sx After Power Loss (Note 1)	Wake from "Reset" Types (Note 2)
RTC Alarm	Set RTC_EN bit in PM1_EN register.	Yes	Yes	Yes	No
Power Button	Always enabled as Wake event.	Yes	Yes	Yes	Yes
Any GPIOs enabled for wake from the set of GPP_A to GPP_E		Yes	No	No	No
GBE_WAKE#	Enabled natively (unless pin is configured to be in GPIO mode)	Yes	Yes	Yes	Yes
GBE	Will use PME#. Wake enable set with LAN logic.	Yes	No	Yes	No
Intel® High Definition Audio	Event sets PME_B0_STS bit; PM_B0_EN must be enabled. Can not wake from S5 state if it was entered due to power failure or power button override.	Yes	No	Yes	No
Primary PME#	PME_B0_EN bit in GPE0_EN[127:96] register.	Yes	No	Yes	No
Secondary PME#	Set PME_EN bit in GPE0_EN[127:96] register.	Yes	No	Yes	No
PCI Express WAKE# pin	PCIEXPWAK_DIS bit.	Yes	Yes	Yes	No
SMBALERT#	(Note 4)	Yes	No	Yes	Yes
SMBus Slave Wake Message (01h)	Wake/SMI# command always enabled as a Wake event. ***Note:*** SMBus Slave Message can wake the system from S3–S5, as well as from S5 due to Power Button Override.	Yes	No	Yes	Yes
SMBus Host Notify message received	HOST_NOTIFY_WKEN bit SMBus Slave Command register. Reported in the SMB_WAK_STS bit in the GPE0_STS register.	Yes	No	Yes	Yes

Table 9-6. Causes of Wake Events (Sheet 2 of 2)

Cause	How Enabled	Wake from Sx	Wake from Deep Sx	Wake from Sx After Power Loss (Note 1)	Wake from "Reset" Types (Note 2)
Intel® ME Non-Maskable Wake	Always enabled as a wake event.	Yes	No	Yes	Yes
Integrated WoL Enable Override	WoL Enable Override bit (in Configuration Space).	Yes	No	Yes	Yes
Wake Alarm Device	WADT_EN in GPE0_EN[127:96]	Yes	Yes	No	No

Notes:
1. This column represents what the PCH would honor as wake events but there may be enabling dependencies on the device side which are not enabled after a power loss.
2. Reset Types include: Power Button override, Intel ME initiated power button override, Intel ME initiated host partition reset with power down, Intel ME Watchdog Timer, SMBus unconditional power down, processor thermal trip, PCH catastrophic temperature event.
3. SMBALERT# signal is muxed with a GPIO pin that defaults to GPIO mode. Hence, SMBALERT# related wakes are possible only when this GPIO is configured in native mode, which means that BIOS must program this GPIO to operate in native mode before this wake is possible. Because GPIO configuration is in the resume well, wakes remain possible until one of the following occurs: BIOS changes the pin to GPIO mode, a G3 occurs or Deep Sx entry occurs.

9.7.7.4 PCI Express WAKE# Signal and PME Event Message

PCI Express ports can wake the platform from any sleep state (S1, S3, S4, or S5 or Deep Sx) using the WAKE# pin. WAKE# is treated as a wake event, but does not cause any bits to go active in the GPE_STS register.

PCI Express ports and the processor have the ability to cause PME using messages. These are logically OR'd to set the single PCI_EXP_STS bit. When a PME message is received, the PCH will set the PCI_EXP_STS bit. If the PCI_EXP_EN bit is also set, the PCH can cause an SCI via GPE0_STS register.

9.7.7.5 Sx-G3-Sx, Handling Power Failures

Depending on when the power failure occurs and how the system is designed, different transitions could occur due to a power failure.

The AFTERG3_EN bit provides the ability to program whether or not the system should boot once power returns after a power loss event. If the policy is to not boot, the system remains in an S5 state (unless previously in S4). There are only three possible events that will wake the system after a power failure:

1. **PWRBTN#:** PWRBTN# is always enabled as a wake event. When RSMRST# is low (G3 state), the PWRBTN_STS bit is reset. When the PCH exits G3 after power returns (RSMRST# goes high), the PWRBTN# signal is already high (because V_{CC}-standby goes high before RSMRST# goes high) and the PWRBTN_STS bit is 0.
2. **RTC Alarm:** The RTC_EN bit is in the RTC well and is preserved after a power loss. Like PWRBTN_STS the RTC_STS bit is cleared when RSMRST# goes low.

The PCH monitors both PCH_PWROK and RSMRST# to detect for power failures. If PCH_PWROK goes low, the PWROK_FLR bit is set. If RSMRST# goes low, PWR_FLR is set.

Although PME_EN is in the RTC well, this signal cannot wake the system after a power loss. PME_EN is cleared by RTCRST#, and PME_STS is cleared by RSMRST#.

Table 9-7. Transitions Due to Power Failure

State at Power Failure	AFTERG3_EN Bit	Transition When Power Returns
S0, S1, S3	1 0	S5 S0
S4	1 0	S4 S0
S5	1 0	S5 S0
Deep Sx	1 0	Deep Sx[1] S0

Note:
1. Entry state to Deep Sx is preserved through G3 allowing resume from Deep Sx to take appropriate path (that is, return to S3, S4 or S5).

9.7.7.6 Deep Sx

To minimize power consumption while in S3/S4/S5, the PCH supports a lower power, lower featured version of these power states known as Deep Sx. In the Deep Sx state, the Suspend wells are powered off, while the Deep Sx Well remains powered. A limited set of wake events are supported by the logic located in the DSW.

The Deep Sx capability and the SUSPWRDNACK pin functionality are mutually exclusive.

9.7.7.6.1 Entry Into Deep Sx

A combination of conditions is required for entry into Deep Sx.

All of the following must be met:

a. Intel ME in M-Off AND

((ACPRESENT = 0) AND ((DPS3_EN_DC AND S3) OR (DPS4_EN_DC AND S4) OR (DPS5_EN_DC AND S5)))

Table 9-8. Supported Deep Sx Policy Configurations

Configuration	DPS3_EN_DC	DPS4_EN_DC	DPS4_EN_AC	DPS5_EN_DC	DPS5_EN_AC
1. Enabled in S5 when on battery (ACPRESENT = 0)	0	0	0	1	0
2. Enabled in S4 and S5 when on battery (ACPRESENT = 0)	0	1	0	1	0
3. Enabled in S3, S4 and S5 when on battery (ACPRESENT = 0)	1	1	0	1	0
4. Deep S3/S4/ S5 disabled	0	0	0	0	0

Note: All other configuration is RESERVED.

The PCH also performs a SUSWARN#/SUSACK# handshake to ensure the platform is ready to enter Deep Sx. The PCH asserts SUSWARN# as notification that it is about to enter Deep Sx. Before the PCH proceeds and asserts SLP_SUS#, the PCH waits for SUSACK# to assert.

9.7.7.6.2 Exit from Deep Sx

While in Deep Sx, the PCH monitors and responds to a limited set of wake events (RTC Alarm, Power Button and WAKE#). Upon sensing an enabled Deep Sx wake event, the PCH brings up the suspend well by de-asserting SLP_SUS#.

Table 9-9. Deep Sx Wake Events

Event	Enable
RTC Alarm	RTC_DS_WAKE_DIS (RCBA+3318h:Bit 21)
Power Button	Always enabled
PCIe* WAKE# pin	PCIEXP_WAK_DIS
Wake Alarm Device	WADT_EN

ACPRESENT has some behaviors that are different from the other Deep Sx wake events. If the Intel ME has enabled ACPRESENT as a wake event then it behaves just like any other Intel ME Deep Sx wake event. However, even if ACPRESENT wakes are not enabled, if the host policies indicate that Deep Sx is only supported when on battery, then ACPRESENT going high will cause the PCH to exit Deep Sx. In this case, the suspend wells gets powered up and the platform remains in S3/M-Off, S4/M-Off or S5/M-Off. If ACPRESENT subsequently drops (before any host or Intel ME wake events are detected), the PCH will re-enter Deep Sx.

9.7.8 Event Input Signals and Their Usage

The PCH has various input signals that trigger specific events. This section describes those signals and how they should be used.

9.7.8.1 PWRBTN# (Power Button)

The PCH PWRBTN# signal operates as a "Fixed Power Button" as described in the *Advanced Configuration and Power Interface Specification.* The PWRBTN# signal has a 16 ms debounce on the input. The state transition descriptions are included in Table 9-10.

The transitions start as soon as the PWRBTN# is pressed (but after the debounce logic if PM_CFG.PB_DB_MODE='0'), and does not depend on when the Power Button is released. If PM_CFG.PB_DB_MODE='1', the transitions start as soon as the PWRBTN# is pressed, even before passing through the debounce logic. Subsequent PWRBTN# edges are ignored until the debounce period has expired. A power button override will force a transition to S5, even if PCH_PWROK is not active.

During the time that any SLP_* signal is stretched for an enabled minimum assertion width, the host wake-up is held off. As a result, it is possible that the user will press and continue to hold the Power Button waiting for the system to wake. Unfortunately, a 4 second press of the Power Button is defined as an unconditional power down, resulting in the opposite behavior that the user was intending. Therefore the Power Button Override Timer will be extended to 9-10 seconds while the SLP_* stretching timers are in progress. Once the stretching timers have expired, the Power Button will awake the system. If the user continues to press Power Button for the remainder of the 9-10 seconds it will result in the override condition to S5. Extension of the Power Button Override timer is only enforced following graceful sleep entry and during host partition resets with power cycle or power down. The timer is not extended immediately following power restoration after a global reset, G3 or Deep Sx.

Table 9-10. Transitions Due to Power Button

Present State	Event	Transition/Action	Comment
S0/Cx	PWRBTN# goes low	SMI or SCI generated (depending on SCI_EN, PWRBTN_EN and GLB_SMI_EN)	Software typically initiates a Sleep state
S3 – S5, Deep Sx	PWRBTN# goes low	Wake Event. Transitions to S0 state	Standard wakeup ***Note:*** Could be impacted by SLP_* min assertion.
G3	PWRBTN# pressed	None	No effect since no power Not latched nor detected ***Note:*** During G3 exit, PWRBTN# must be asserted at least until SLP_SUS# de-asserts to be registered by PCH as a valid wake event.
S0 – S4	PWRBTN# held low for at least 4 consecutive seconds	Unconditional transition to S5 state and if Deep Sx is enabled and conditions are met per Section 9.7.7.6, the system will then transition to Deep Sx.	No dependence on processor or any other subsystem

Power Button Override Function

If PWRBTN# is observed active for at least four consecutive seconds (always sampled after the output from debounce logic), the PCH should unconditionally transition to the G2/S5 state or Deep Sx, regardless of present state (S0 – S4), even if the PCH_PWROK is not active. In this case, the transition to the G2/S5 state or Deep Sx does not depend on any particular response from the processor, nor any similar dependency from any other subsystem.

The PWRBTN# status is readable to check if the button is currently being pressed or has been released. If PM_CFG.PB_DB_MODE='0', the status is taken after the debounce. If PM_CFG.PB_DB_MODE='1', the status is taken before the debounce. In either case, the status is readable using the PWRBTN_LVL bit.

Note: The 4-second PWRBTN# assertion should only be used if a system lock-up has occurred.

Sleep Button

The *Advanced Configuration and Power Interface Specification* defines an optional Sleep button. It differs from the power button in that it only is a request to go from S0 to S1–S4 (not S5). Also, in an S5 state, the Power Button can wake the system, but the Sleep Button cannot.

Although the PCH does not include a specific signal designated as a Sleep Button, one of the GPIO signals can be used to create a "Control Method" Sleep Button. See the *Advanced Configuration and Power Interface Specification* for implementation details.

9.7.8.2 PME# (PCI Power Management Event)

The PME# signal comes from a PCI Express device to request that the system be restarted. The PME# signal can generate an SMI#, SCI, or optionally a wake event. The event occurs when the PME# signal goes from high to low. No event is caused when it goes from low to high.

There is also an internal PME_B0 bit. This is separate from the external PME# signal and can cause the same effect.

9.7.8.3 SYS_RESET# Signal

When the SYS_RESET# pin is detected as active after the 16 ms debounce logic, the PCH attempts to perform a "graceful" reset by entering a host partition reset entry sequence.

Once the reset is asserted, it remains asserted for 5 to 6 ms regardless of whether the SYS_RESET# input remains asserted or not. It cannot occur again until SYS_RESET# has been detected inactive after the debounce logic, and the system is back to a full S0 state with PLTRST# inactive.

Note: If bit 3 of the CF9h I/O register is set then SYS_RESET# will result in a full power-cycle reset.

Note: It is not recommended to use the PCH_PWROK pin for a reset button as it triggers a global power cycle reset.

Note: SYS_RESET# is in the primary power well but it only affects the system when PCH_PWROK is high.

9.7.8.4 THERMTRIP# Signal

If THERMTRIP# goes active, the processor is indicating an overheat condition, and the PCH immediately transitions to an S5 state, driving SLP_S3#, SLP_S4#, SLP_S5# low, and setting the GEN_PMCON_2.PTS bit. The transition looks like a power button override.

When a THERMTRIP# event occurs, the PCH will power down immediately without following the normal S0 -> S5 path. The PCH will immediately drive SLP_S3#, SLP_S4#, and SLP_S5# low within 1 us after sampling THERMTRIP# active.

If the processor is running extremely hot and is heating up, it is possible (although very unlikely) that components around it, such as the PCH, are no longer executing cycles properly. Therefore, if THERMTRIP# goes active, and the PCH is relying on state machine logic to perform the power down, the state machine may not be working, and the system will not power down.

The PCH provides filtering for short low glitches on the THERMTRIP# signal in order to prevent erroneous system shut downs from noise. Glitches shorter than 25 nsec are ignored.

PCH must only honor the THERMTRIP# pin while it is being driven to a valid state by the processor. The THERMTRIP# Valid Point ='0', implies PCH will start monitoring THERMTRIP# at PLTRST# de-assertion (default). The THERMTRIP# Valid Point ='1', implies PCH will start monitoring THERMTRIP# at PROCPWRGD assertion. Regardless of the setting, PCH must stop monitoring THERMTRIP# at PROCPWRGD de-assertion.

Note: A thermal trip event will clear the PWRBTN_STS bit.

9.7.8.5 Sx_Exit_Holdoff# Signal

When S3/S4/S5 is entered and SLP_A# is asserted, Sx_Exit_Holdoff# can be asserted by a platform component to delay reset to S0. SLP_A# deassertion is an indication of the intent to resume to S0, but this will be delayed as long as Sx_Exit_Holdoff# is asserted. Sx_Exit_Holdoff# is ignored outside of an S3/S4/S5 entry sequence with

SLP_A# asserted. With the deassertion of RSMRST# (either from G3->S0 or DeepSX -> S0), this pin is a GPIO input and must be programmed by BIOS to operate as Sx_Exit_Holdoff#. When SLP_A# is asserted (or it is deasserted bu Sx_Exit_Holdoff is asserted), the PCH will not access the SPI Flash. How a platform uses this signal is platform specific. This signal is only sampled when the eSPI/LPC bus is used as an LPC.

9.7.8.6 PE_RST#

PE_RST# is a new signal on the PCH. The new Intel QAT and 10/1 GbE functions operate as standalone functions, and in End Point Only mode, the core sPCH functions are disabled but Intel QAT and 10/1GbE can still operate sending data back and forth on the PCIe uplink. Because of this, they have their own reset pin, PE_RST#. On a board, the resets driven by the PCH would be looped back into the PE_RST# pin. In the event of the PCH being used in EPO mode, then the appropriate slot reset signal will be driven onto PE_RST#.

9.7.8.7 Multi-PCH Reset Synchronization

The Intel® C620 Series Chipset PCH now incorporates a hand shake mechanism for systems with multiple PCHs. There would be one Legacy PCH (LPCH) and there can be multiple non-legacy PCHs (NLPCH). The LPCH will always be the PCH that coordinates boot, the SX entry and reset sequencing. The NLPCH will be reactive to sequencing state indications from the LPCH. The DO_RST_OUT# pins from all the NLPCH will go to the DO_RST_IN# pin of the LPCH. The DO_RST_OUT# of the LPCH will go to the DO_RST_IN# pins of the NLPCHs. Finally, RST_DONE are outputs from the NLPCH and go into the RST_DONE of the LPCH. When in the S0 state, each PCH recognizes DO_RST_IN# assertion as a warm reset. It is always the LPCH that initiates this. If a NLPCH PCH needs a reset to occur, it will drive DO_RST_OUT# to the LPCH, which will in turn drive its DO_RST_OUT# to the DO_RST_IN# of the NLPCHs.

9.7.9 ALT Access Mode

Before entering a low power state, several registers from powered down parts may need to be saved. In the majority of cases, this is not an issue, as registers have read and write paths. However, several of the ISA compatible registers are either read only or write only. To get data out of write-only registers, and to restore data into read-only registers, the PCH implements an ALT access mode.

If the ALT access mode is entered and exited after reading the registers of the PCH timer (8254), the timer starts counting faster (13.5 ms). The following steps listed below can cause problems:

1. BIOS enters ALT access mode for reading the PCH timer related registers.
2. BIOS exits ALT access mode.
3. BIOS continues through the execution of other needed steps and passes control to the operating system.

After getting control in step #3, if the operating system does not reprogram the system timer again, the timer ticks may be happening faster than expected.

Operating systems reprogram the system timer and therefore do not encounter this problem.

For other operating systems, the BIOS should restore the timer back to 54.6 ms before passing control to the operating system. If the BIOS is entering ALT access mode before entering the suspend state it is not necessary to restore the timer contents after the exit from ALT access mode.

9.7.9.1 Write Only Registers with Read Paths in ALT Access Mode

The registers described in Table 9-11 have read paths in ALT access mode. The access number field in the table indicates which register will be returned per access to that port.

Table 9-11. Write Only Registers with Read Paths in ALT Access Mode

Restore Data				Restore Data			
I/O Addr	# of Rds	Access	Data	I/O Addr	# of Rds	Access	Data
20h	12	1	PIC ICW2 of Master controller	40h	7	1	Timer Counter 0 status, bits [5:0]
		2	PIC ICW3 of Master controller			2	Timer Counter 0 base count low byte
		3	PIC ICW4 of Master controller			3	Timer Counter 0 base count high byte
		4	PIC OCW1 of Master controller[1]			6	Timer Counter 2 base count low byte
		5	PIC OCW2 of Master controller			7	Timer Counter 2 base count high byte
		6	PIC OCW3 of Master controller	42h	1		Timer Counter 2 status, bits [5:0]
		7	PIC ICW2 of Slave controller	70h	1		Bit 7 = NMI Enable, Bits [6:0] = RTC Address
		8	PIC ICW3 of Slave controller	70h	1		Bit 7 = Read value is '0'. Bits [6:0] = RTC Address
		9	PIC ICW4 of Slave controller				
		10	PIC OCW1 of Slave controller[1]				
		11	PIC OCW2 of Slave controller				
		12	PIC OCW3 of Slave controller				

Notes:
1. The OCW1 register must be read before entering ALT access mode.
2. Bits 5, 3, 1, and 0 return 0.

9.7.9.2 PIC Reserved Bits

Many bits within the PIC are reserved, and must have certain values written in order for the PIC to operate properly. Therefore, there is no need to return these values in ALT access mode. When reading PIC registers from 20h and A0h, the reserved bits shall return the values listed in Table 9-12.

Table 9-12. PIC Reserved Bits Return Values

PIC Reserved Bits	Value Returned
ICW2(2:0)	000
ICW4(7:5)	000
ICW4(3:2)	00
ICW4(0)	0
OCW2(4:3)	00
OCW3(7)	0
OCW3(5)	Reflects bit 6
OCW3(4:3)	01

9.7.9.3 Read Only Registers with Write Paths in ALT Access Mode

The registers described in Table 9-13 have write paths to them in ALT access mode. Software restores these values after returning from a powered down state. These registers must be handled special by software. When in normal mode, writing to the base address/count register also writes to the current address/count register. Therefore, the base address/count must be written first, then the part is put into ALT access mode and the current address/count register is written.

Table 9-13. Register Write Accesses in ALT Access Mode

I/O Address	Register Write Value
08h	DMA Status Register for Channels 0–3
D0h	DMA Status Register for Channels 4–7

9.7.10 System Power Supplies, Planes, and Signals

9.7.10.1 Power Plane Control with SLP_S3#, SLP_S4#, SLP_S5#, SLP_A# and SLP_GbE#

The SLP_S3# output signal can be used to cut power to the system core supply, since it only goes active for the Suspend-to-RAM state (typically mapped to ACPI S3). Power must be maintained to the PCH primary well, and to any other circuits that need to generate Wake signals from the Suspend-to-RAM state. During S3 (Suspend-to-RAM) all signals attached to powered down planes will be tri-stated or driven low, unless they are pulled using a pull-up resistor.

Cutting power to the system core supply may be done using the power supply or by external FETs on the motherboard.

The SLP_S4# or SLP_S5# output signal can be used to cut power to the system core supply, as well as power to the system memory, since the context of the system is saved on the disk. Cutting power to the memory may be done using the power supply, or by external FETs on the motherboard.

The SLP_S4# output signal is used to remove power to additional subsystems that are powered during SLP_S3#.

SLP_S5# output signal can be used to cut power to the system core supply, as well as power to the system memory, since the context of the system is saved on the disk. Cutting power to the memory may be done using the power supply, or by external FETs on the motherboard.

SLP_A# output signal can be used to cut power to the Intel ME and SPI flash on a platform that supports the M3 state (for example, certain power policies in Intel AMT).

SLP_GbE# output signal can be used to cut power to the external Intel 82579 GbE PHY device.

9.7.10.2 SLP_S4# and Suspend-to-RAM Sequencing

The system memory suspend voltage regulator is controlled by the glue logic. The SLP_S4# signal should be used to remove power to system memory rather than the SLP_S5# signal. The SLP_S4# logic in the PCH provides a mechanism to fully cycle the power to the DRAM and/or detect if the power is not cycled for a minimum time.

Note: To use the minimum DRAM power-down feature that is enabled by the SLP_S4# Assertion Stretch Enable bit (D31:F0:A4h Bit 3), the DRAM power must be controlled by the SLP_S4# signal.

9.7.10.3 PCH_PWROK Signal

When asserted, PCH_PWROK is an indication to the PCH that its core well power rails are powered and stable. PCH_PWROK can be driven asynchronously. When PCH_PWROK is low, the PCH asynchronously asserts PLTRST#. PCH_PWROK must not glitch, even if RSMRST# is low.

It is required that the power associated with the PCIe have been valid for 99 ms prior to PCH_PWROK assertion in order to comply with the 100 ms PCIe* 2.0 specification on PLTRST# de-assertion.

Note: SYS_RESET# is recommended for implementing the system reset button. This saves external logic that is needed if the PCH_PWROK input is used. Additionally, it allows for better handling of the SMBus and processor resets and avoids improperly reporting power failures.

9.7.10.4 SLP_GBE# Pin Behavior

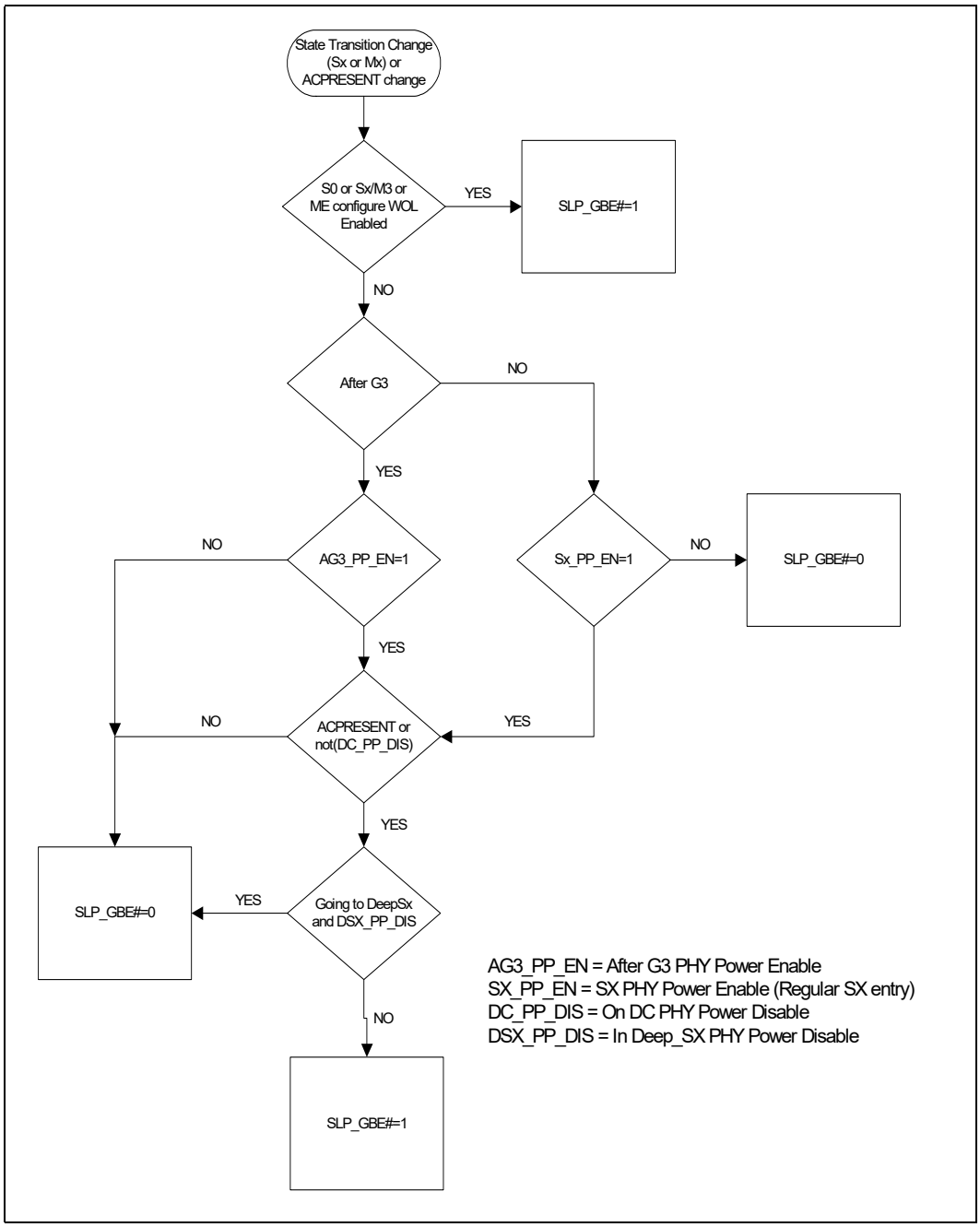
The PCH controls the voltage rails into the external GBE PHY using the SLP_GBE# pin.

- The GBE PHY is always powered when the host and ME systems are running.
  - SLP_GBE#='1' whenever SLP_S3#='1' or SLP_A#='1'.
- If the GBE PHY is required by the Intel ME in Sx/M-Off or Deep Sx, the Intel ME must configure SLP_GBE#='1' irrespective of the power source and the destination power state. The Intel ME must be powered at least once after G3 to configure this.
- If the GBE PHY is required after a G3 transition, the host BIOS must set AG3_PP_EN (B0:D31:F0:A2h bit 12).
- If the GBE PHY is required in Sx/M-Off, the host BIOS must set SX_PP_EN (B0:D31:F0:A2h bit 11).
- If the GBE PHY is required in Deep Sx, the host BIOS must keep DSX_PP_DIS (B0:D31:F0:A2h bit 13) cleared.
- If the GBE PHY is not required if the source of power is battery, the host BIOS must set DC_PP_DIS (B0:D31:F0:A2h bit 14).

Note: The Intel ME configuration of SLP_GBE# in Sx/M-Off and Deep Sx is dependent on the Intel ME power policy configuration.

The flow chart below shows how a decision is made to drive SLP_GBE# every time its policy needs to be evaluated.

Figure 9-1. Conceptual Diagram of SLP_GbE#



9.7.10.5 SUSPWRDNACK/SUSWARN#/GPP_A13 Steady State Pin Behavior

Table 9-14 summarizes SUSPWRDNACK/SUSWARN#/GPP_A13 pin behavior.

Table 9-14. SUSPWRDNACK/SUSWARN#/GPP_A13 Pin Behavior

Pin	Deep Sx (Supported /Not-Supported)	GPP_A13 Input/Output (Determine by GP_IO_SEL bit)	Pin Value in S0	Pin Value in Sx/M-Off	Pin Value in Sx/M3	Pin Value in Deep Sx
SUSPWRDNACK	Not Supported	Native	0	Depends on Intel® ME power package and power source (Note 1)	0	Off
SUSWARN#	Supported	Native	1	1 (Note 2)	1	Off
GPP_A13	Don't Care	IN	High-Z	High-Z	High-Z	Off
	Don't Care	OUT	Depends on GPP_A13 output data value	Depends on GPP_A13 output data value	Depends on GPP_A13 output data value	Off

Notes:
1. PCH will drive SPDA pin based on ME power policy configuration.
2. If entering Deep Sx, pin will assert and become undriven ("Off") when suspend well drops upon Deep Sx entry.

Table 9-15. SUSPWRDNACK During Reset

Reset Type (Note 1)	SPDA Value
Straight to S5	PCH initially drive '0' and then drive per the Intel ME power policy configuration.

9.7.10.6 RTCRST# and SRTCRST#

RTCRST# is used to reset PCH registers in the RTC well to their default value. If a jumper is used on this pin, it should only be pulled low when system is in the G3 state and then replaced to the default jumper position. Upon booting, BIOS should recognize that RTCRST# was asserted and clear internal PCH registers accordingly. It is imperative that this signal not be pulled low in the S0 to S5 states.

SRTCRST# is used to reset portions of the Intel Management Engine and should not be connected to a jumper or button on the platform. The only time this signal gets asserted (driven low in combination with RTCRST#) should be when the coin cell battery is removed or not installed and the platform is in the G3 state. Pulling this signal low independently (without RTCRST# also being driven low) may cause the platform to enter an indeterminate state. Similar to RTCRST#, it is imperative that SRTCRST# not be pulled low in the S0 to S5 states.

9.7.11 Legacy Power Management Theory of Operation

Instead of relying on ACPI software, legacy power management uses BIOS and various hardware mechanisms. The scheme relies on the concept of detecting when individual subsystems are idle, detecting when the whole system is idle, and detecting when accesses are attempted to idle subsystems.

However, the operating system is assumed to be at least APM enabled. Without APM calls, there is no quick way to know when the system is idle between keystrokes. The PCH does not support burst modes.

9.7.12 Reset Behavior

When a reset is triggered, the PCH will send a warning message to the processor to allow the processor to attempt to complete any outstanding memory cycles and put memory into a safe state before the platform is reset. When the processor is ready, it will send an acknowledge message to the PCH. Then a reset_warn/reset_ack handshake occurs with the processor. Once the message is received the PCH asserts PLTRST#.

If the PCH does not receive an acknowledge message from the processor within 4-5 seconds after sending the reset_warn message, it will trigger a global reset which will drive PLTRST# low.

A PLTRST# asserted is called a Host Reset or Host Partition Reset. Depending on the trigger, a host reset may also result in power cycling see Table 9-16 for details. If a host reset is triggered and the PCH times out before receiving an acknowledge message from the processor, a Global Reset with power-cycle will occur.

A reset in which the host and Intel ME partitions of the platform are reset is called a Global Reset. During a Global Reset, all PCH functionality is reset except RTC Power Well backed information and Suspend well status, configuration, and functional logic for controlling and reporting the reset. Intel ME and Host power back up after the power-cycle period.

Straight to S5 is another reset type where all power wells that are controlled by the SLP_S3#, SLP_S4#, and SLP_A# pins, as well as SLP_S5# and SLP_GBE# (if pins are not configured as GPIOs), are turned off. All PCH functionality is reset except RTC Power Well backed information and suspend well status, configuration, and functional logic for controlling and reporting the reset. The host stays there until a valid wake event occurs.

A new functionality that is in the Intel® C620 Series Chipset is the function called Demoted Warm Reset (DWR). In this mode of operation, global resets that would cause power cycles are demoted to a warm reset which does not. This feature has been added to cover the condition where some action would cause a power cycle, and that action could cause error register information to be lost. Because the condition that caused the global reset is possibly not cleared without a power cycle, it is possible that the system may not return to a function state if DWR is enabled. DWR is enabled via the GBL2HOST_EN register.

Table 9-16 shows the various reset triggers.

Table 9-16. Causes of Host and Global Resets (Sheet 1 of 3)

Trigger	Host Reset Without Power Cycle[1]	Host Reset With Power Cycle[2]	Global Reset With Power Cycle[3]	Straight to S5[6] (Host Stays There)
Global reset occurs while Global2Host is enabled.	Yes	No	No (note 8)	
Write of 0Eh to CF9h (RST_CNT Register)	No	Yes	No (Note 4)	
Write of 06h to CF9h (RST_CNT Register)	Yes	No	No (Note 4)	
SYS_RESET_N Asserted and CF9h (RST_CNT Register) Bit 3 = 0	Yes	No	No (Note 4)	

Table 9-16. Causes of Host and Global Resets (Sheet 2 of 3)

Trigger	Host Reset Without Power Cycle[1]	Host Reset With Power Cycle[2]	Global Reset With Power Cycle[3]	Straight to S5[6] (Host Stays There)
PE_RST_N in End Point Only mode	No	Yes	No	
DO_RESET_IN_N asserts and CF9 Bit 3 =0	Yes	No	No	
DO_RESET_IN_N asserts and CF9 Bit 3 =1	No	Yes	No (Note 4)	
SYS_RESET# Asserted and CF9h (RST_CNT Register) Bit 3 = 1	No	Yes	No (Note 4)	
SMBus Slave Message received for Reset with Power-Cycle	No	Yes	No (Note 4)	
SMBus Slave Message received for Reset without Power-Cycle	Yes	No	No (Note 4)	
SMBus Slave Message received for unconditional Power Down	No	No	No	Yes
TCO Watchdog Timer reaches zero two times	Yes	No	No (Note 4)	
Power Failure: PCH_PWROK signal goes inactive in S0/S1 or DSW_PWROK drops	No	No	Yes	
SYS_PWROK Failure: SYS_PWROK signal goes inactive in S0/S1	No	No	Yes	
Processor Thermal Trip (THERMTRIP#) causes transition to S5 and reset asserts	No	No	No	Yes
PCH internal thermal sensors signals a catastrophic temperature condition	No	No	No	Yes
Power Button 4 second override causes transition to S5 and reset asserts	No	No	No	Yes
Special shutdown cycle from processor causes CF9h-like PLTRST# and CF9h (RST_CNT Register) Bit 3 = 1	No	Yes	No (Note 4)	
Special shutdown cycle from processor causes CF9h-like PLTRST# and CF9h (RST_CNT Register) Bit 3 = 0	Yes	No	No (Note 4)	
Intel® Management Engine Triggered Host Reset without Power-Cycle	Yes	No	No (Note 4)	
Intel® Management Engine Triggered Host Reset with Power-Cycle	No	Yes	No (Note 4)	
Intel® Management Engine Triggered Power Button Override	No	No	No	Yes
Intel® Management Engine Watchdog Timer Timeout	No	No	No	Yes
Intel® Management Engine Triggered Global Reset	No	No	Yes	
Intel® Management Engine Triggered Host Reset with power down (host stays there)	No	Yes (Note 5)	No (Note 4)	
IE Triggered Host Reset without Power-Cycle	Yes	No	No (Note 4)	
IE Triggered Host Reset with Power-Cycle	No	Yes	No (Note 4)	
IE Triggered Power Button Override	No	No	No	Yes
IE Triggered Global Reset	No	No	Yes	
IE Triggered Host Reset with power down (host stays there)	No	Yes (Note 5)	No (Note 4)	
IE Watchdog Timer Timeout	No	No	No	Yes
IE Hardware Uncorrectable Error	No	No	No	Yes

Table 9-16. Causes of Host and Global Resets (Sheet 3 of 3)

Trigger	Host Reset Without Power Cycle[1]	Host Reset With Power Cycle[2]	Global Reset With Power Cycle[3]	Straight to S5[6] (Host Stays There)
PLTRST# Entry Timeout (Note 7)	No	No	Yes	
PROCPWRGD Stuck Low	No	No	Yes	
Power Management Watchdog Timer	No	No	No	Yes
Intel® Management Engine Hardware Uncorrectable Error	No	No	No	Yes

Notes:
1. The PCH drops this type of reset request if received while the system is in S3/S4/S5.
2. PCH does not drop this type of reset request if received while system is in a software-entered S3/S4/S5 state. However, the PCH will perform the reset without executing the RESET_WARN protocol in these states.
3. The PCH does not send warning message to processor, reset occurs without delay.
4. Trigger will result in global reset with power-cycle if the acknowledge message is not received by the PCH.
5. The PCH waits for enabled wake event to complete reset.
6. Upon entry to S5, if Deep Sx is enabled and conditions are met per Section 9.7.7.6, the system will transition to Deep Sx.
7. PLTRST# Entry Timeout is automatically initiated if the hardware detects that the PLTRST# sequence has not been completed within four seconds of being started.
8. Qualified global reset occurs while global resets are converted to warm resets

9.7.13 ADR

The Asynchronous DRAM Refresh (ADR) feature provides a mechanism to enable preservation of key data in NVDIMM system memory, in the event of AC power supply failure. ADR entry forces a flush of data from specific ADR protected "write buffers" inside processor. Any write data not in the buffers conceptually referred to as the "ADR protected buffers" is lost on power-failure.

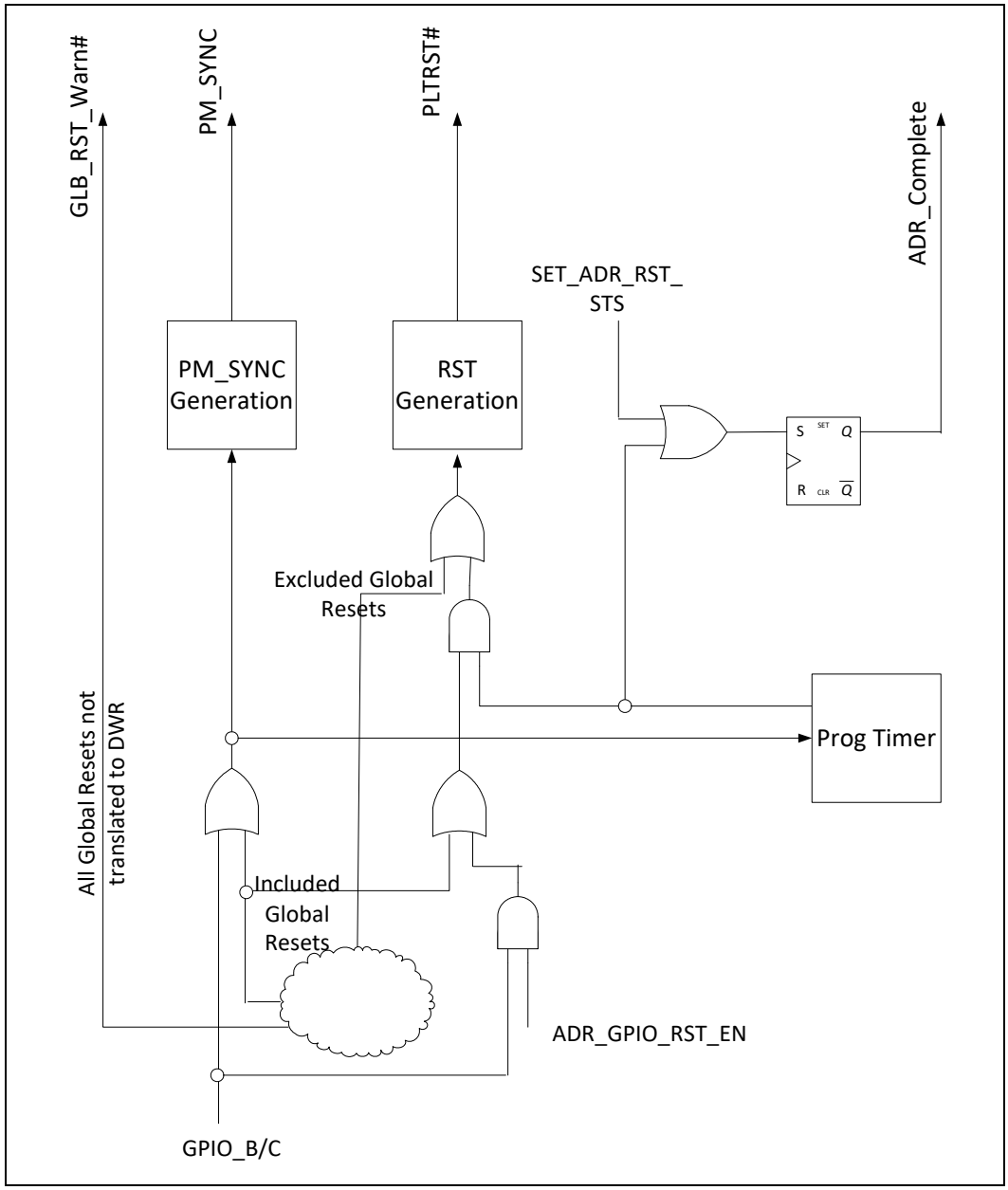
The objective of ADR is to provide an early warning to the CPU of an impending reset or power failure so that it can save critical information while the reset is in progress. For any Host partition reset or S3/4/5 entry, the CPU already receives this early warning via the Reset_Warn or Go_S3/4/5 DMI message. However, prior to this feature the CPU did not receive any warning for a global reset. This feature adds that warning for global reset sources that are considered to be "safe" to delay. A separate enable bit for each global reset source is provided in the ADR_EN register. The method used to provide this early warning to the CPU involves the PM_SYNC pin. The CPU will monitor a particular PM_SYNC state for this indication. For power failure source of global reset, it would be impossible for the PCH to use PM_SYNC to warn the CPU of a PCH_PWROK or RSMRST# power failure, since PM_SYNC control is immediately lost in those scenarios. However, platforms supporting ADR receive an early warning from the power supply indicating that AC power has been lost and the platform voltage rails will soon be dropping. This indication will be connected to one of 4 GPIOs capable of driving PM_SYNC. So, before PCH_PWROK or RSMRST# drops that GPIO will drop and trigger the warning to the CPU over PM_SYNC.

The GPIO that triggers and ADR cycle is not an edge triggered GPIO. It is level sensitive. As long as the GPIO is active, then the ADR cycle is in progress and messages will be sent up the PM_SYNC line to the CPU. If the signal on the GPIO is removed, then the logic sees the ADR request as being removed. Therefore, once the GPIO goes active to trigger an ADR, it must keep it in that state until either 1) the platform shuts down by dropping the power good signals, or 2) PLTRST# goes low, signaling the platform is now in reset.

When the timer for ADR expires, the signal ADR_COMPLETE goes active. This bit will stay active until BIOS clears it.

The picture below is a conceptual drawing of how ADR functions.

Figure 9-2. ADR Flow



9.8 Power Management Configuration Registers Summary

The power management registers are distributed within the PCI Device 31: Function 2 space, with dedicated I/O and memory-mapped spaces. Bits not explicitly defined in each register are assumed to be reserved. Writes to reserved bits must retain their previous values. Other than a read/modify/write, software should not attempt to use the value read from a reserved bit, as it may not be consistently 1 or 0.

Table 9-17. Summary of Power Management Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	PCI Identifier (PCIID)—Offset 0h	8086h
4h	7h	Device Status & Command (PCISTS_PCICMD)—Offset 4h	4h
8h	Bh	Class Code & Revision ID (PCICC_PCIRID)—Offset 8h	5800000h
Ch	Fh	Header Type & Master Latency Timer (PCIHTYPE_PCIMLT)—Offset Ch	800000h
10h	13h	32-bit Power Management Data Base Address Register (PM_DATA_BAR)—Offset 10h	0h
2Ch	2Fh	Subsystem Identifiers (PCISID)—Offset 2Ch	0h
40h	43h	ACPI Base Address (ABASE)—Offset 40h	1h
44h	47h	ACPI Control (ACTL)—Offset 44h	0h
48h	4Bh	PM Base Address (PWRMBASE)—Offset 48h	0h
A0h	A3h	General PM Configuration A (GEN_PMCON_A)—Offset A0h	A0000000h
A4h	A7h	General PM Configuration B (GEN_PMCON_B)—Offset A4h	4006h
A8h	ABh	BM_BREAK_EN and Cx State Configuration Register (BM_CX_CNF)—Offset A8h	0h
ACh	AFh	Extended Test Mode Register 3 (ETR3)—Offset ACh	0h

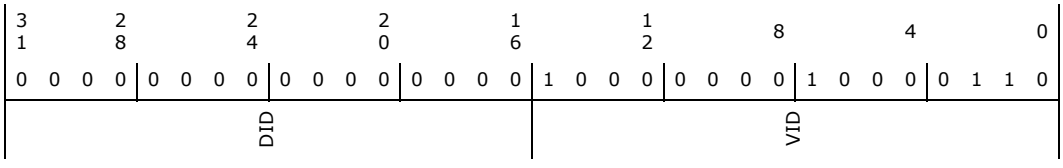
9.8.1 PCI Identifier (PCIID)—Offset 0h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 8086h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0h RO/V	**Device Identification (DID):** These bits are controlled as follows:
15:0	8086h RO	**Vendor Identification (VID):** This is a 16-bit value assigned to Intel. Intel VID=8086h.

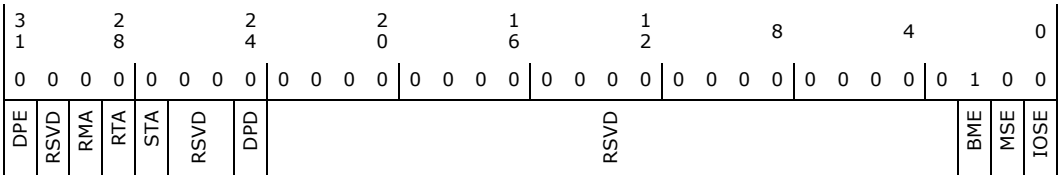
9.8.2 Device Status and Command (PCISTS_PCICMD)—Offset 4h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 4h



Bit Range	Default & Access	Field Name (ID): Description
31	0h RW/1C/V	**Detected Parity Error (DPE):** Set when the bridge detects a parity error on the internal backbone. This bit is reset by PLTRST# assertion.
30	0h RO	Reserved
29	0h RW/1C/V	**Received Master Abort (RMA):** Set when the bridge receives a completion with unsupported request status from the backbone. This bit is reset by PLTRST# assertion.
28	0h RW/1C/V	**Received Target Abort (RTA):** Set when the bridge receives a completion with completer abort status from the backbone. This bit is reset by PLTRST# assertion.
27	0h RW/1C/V	**Signaled Target Abort (STA):** Set when the bridge generates a completion packet with target abort status on the backbone. This bit is reset by PLTRST# assertion.
26:25	0h RO	Reserved
24	0h RW/1C/V	**Data Parity Error Detected (DPD):** Set when the bridge receives a completion packet from the backbone from a previous request, and detects a parity error. This bit is reset by PLTRST# assertion.
23:3	0h RO	Reserved
2	1h RW	**Bus Master Enable (BME):** Bus master Enable does not apply to messages sent out by PMC. This bit is reset by PLTRST# assertion.
1	0h RW	**Memory Space Enable (MSE):** Controls a device's response to Memory Space accesses. This bit controls whether the host to PMC MMIO BAR is enabled or not. This bit is reset by PLTRST# assertion.
0	0h RO	**I/O Space Enable (IOSE):** PMC does not define any IO BARs in the standard PCI header BAR offsets.

9.8.3 Class Code and Revision ID (PCICC_PCIRID)—Offset 8h

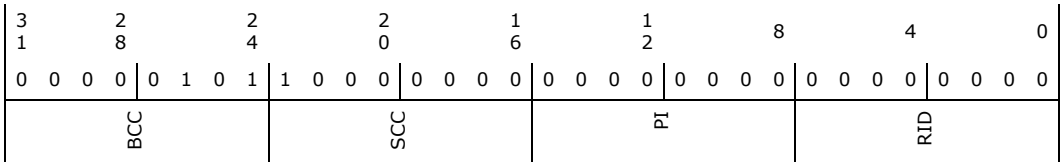
Writing to btis [7:0] of this register controls what is reported in the all of the RID fields of the corresponding registers in the component. The value written does not get directly loaded in this register. However, the value is checked to see which value to report.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 5800000h



Bit Range	Default & Access	Field Name (ID): Description
31:24	5h RO	**Base Class Code (BCC):** Indicates a memory controller device class.
23:16	80h RO	**Sub-Class Code (SCC):** Indicates an unspecified other memory controller.
15:8	0h RO	**Programming Interface (PI):** No programming interface.
7:0	0h RW/V	**Revision ID (RID):** Indicates the part revision. This field is reset by PLTRST# assertion.

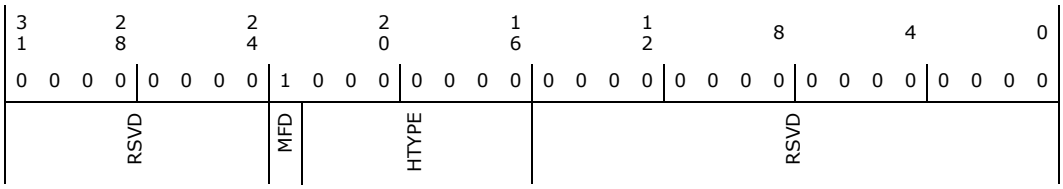
9.8.4 Header Type and Master Latency Timer (PCIHTYPE_PCIMLT)—Offset Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 800000h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	1h RO	**Multi-Function Device (MFD):** Indicates that this is part of a multi-function device.
22:16	0h RO	**Header Type (HTYPE):** Indicates a generic device header.
15:0	0h RO	Reserved

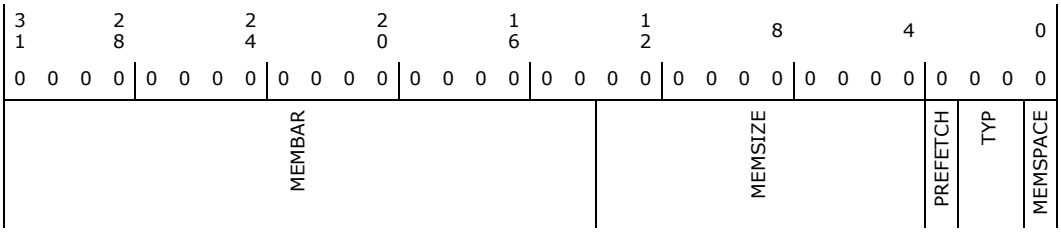
9.8.5 32-bit Power Management Data Base Address Register (PM_DATA_BAR)—Offset 10h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RW	**Memory BAR (MEMBAR):** Software programs this register with the base address of the device's memory region. This field is reset by PLTRST# assertion.
13:4	0h RO	**Memory Size (MEMSIZE):** Hardwired to 0 to indicate 16 KB of memory space
3	0h RO	**Prefetchable (PREFETCH):** A device can mark a range as prefetchable if there are no side effects on reads, the device returns all bytes on reads regardless of the byte enables. Hardwired to 0 to indicate the device's memory space as notprefetchable.
2:1	0h RO	**Type (TYP):** Hardwired to 0 to indicate that Base register is 32-bits wide and mapping can be done anywhere in the 32-bit Memory Space.
0	0h RO	**Memory Space Indicator (MEMSPACE):** Hardwired to 0 to identify a Memory BAR.

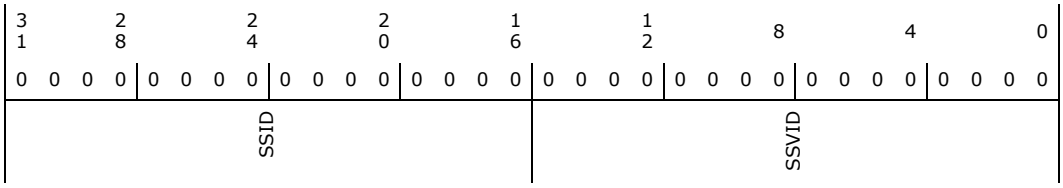
9.8.6 Subsystem Identifiers (PCISID)—Offset 2Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0h RW/O	**Subsystem ID (SSID):** Written by BIOS. Not used by hardware. This field is reset by PLTRST# assertion.
15:0	0h RW/O	**Subsystem Vendor ID (SSVID):** Written by BIOS. Not used by hardware. This field is reset by PLTRST# assertion.

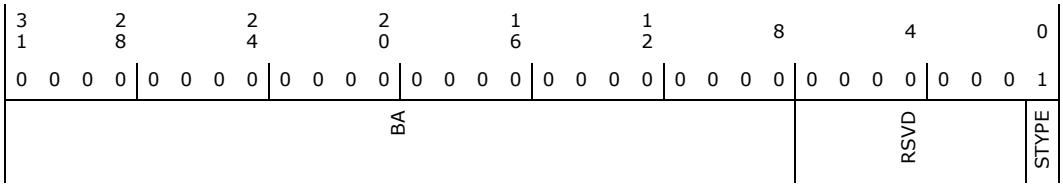
9.8.7 ACPI Base Address (ABASE)—Offset 40h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 1h



Bit Range	Default & Access	Field Name (ID): Description
31:8	0h RW	**Base Address (BA):** Provides the 256 bytes of I/O space for ACPI logic. This field is reset by PLTRST# assertion.
7:1	0h RO	Reserved.
0	1h RO	**Space Type (STYPE):** Always 1 to indicate I/O space.

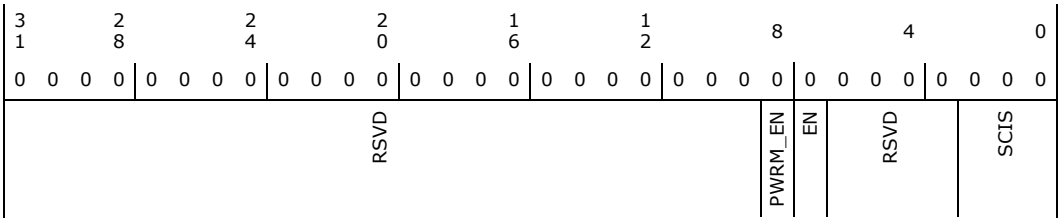
9.8.8 ACPI Control (ACTL)—Offset 44h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:9	0h RO	Reserved
8	0h RW	**PWRM Enable (PWRM_EN):** When set, decode of the memory range pointed by PWRMBASE is enabled. This bit is reset by PLTRST# assertion.
7	0h RW	**ACPI Enable (EN):** When set, decode of the I/O range pointed to by the ACPI base register is enabled and the ACPI power management function is enabled. This bit is reset by PLTRST# assertion.
6:3	0h RO	Reserved
2:0	0h RW	**SCI IRQ Select (SCIS):** Specifies on which IRQ the SCI will internally appear. If not using the APIC, the SCI must be routed to IRQ[9-11], and that interrupt is not sharable with the SERIRQ stream, but is shareable with other PCI interrupts. If using the APIC, the SCI can also be mapped to IRQ20-23, and can be shared with other interrupts. Bits - SCI Map ---- - ------- 000 - IRQ9 001 - IRQ10 010 - IRQ11 011 - Reserved 100 - IRQ20 (only if APIC is enabled) 101 - IRQ21 (only if APIC is enabled) 110 - IRQ22 (only if APIC is enabled) 111 - IRQ23 (only if APIC is enabled) When the interrupt is mapped to APIC interrupts 9, 10 or 11, the APIC should be programmed for active-high reception. When the interrupt is mapped to APIC interrupts 20 through 23, the APIC should be programmed for active-low reception. This field is reset by PLTRST# assertion.

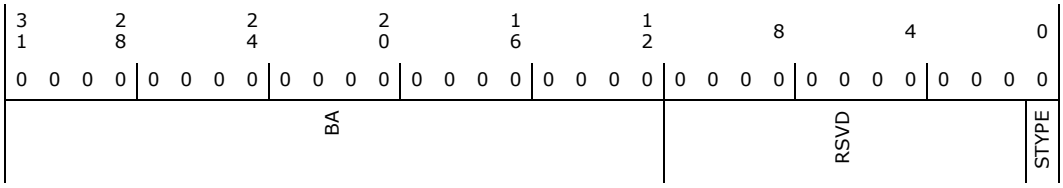
9.8.9 PM Base Address (PWRMBASE)—Offset 48h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:12	0h RW	**Base Address (BA):** Provides the 4 KB of memory space for Power Management registers. This field is reset by PLTRST# assertion.
11:1	0h RO	Reserved.
0	0h RO	**Space Type (STYPE):** Always 0 to indicate memory space.

9.8.10 General PM Configuration A (GEN_PMCON_A)—Offset A0h

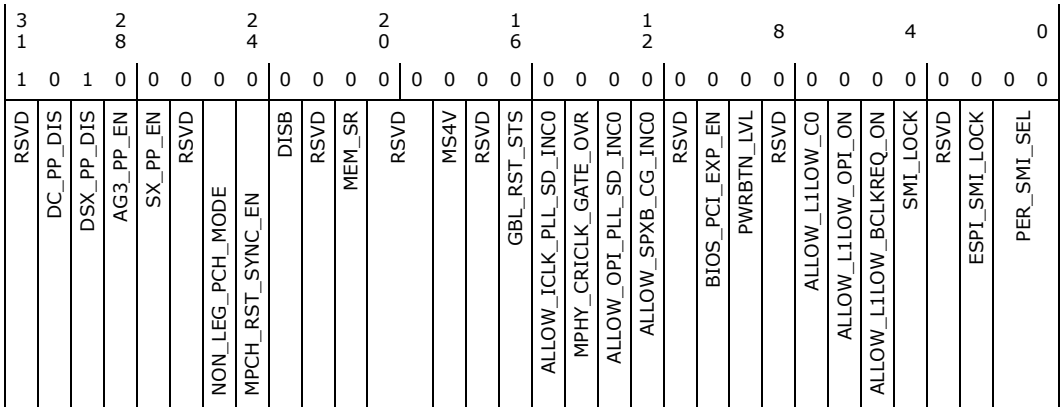
Usage ACPI, Legacy

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 2

Default: A0000000h



Bit Range	Default & Access	Field Name (ID): Description
31	1b RO	**Reserved**
30	0b RW	**DC PHY Power Disable (DC_PP_DIS):** This bit determines the Host software contribution to whether the LAN PHY remains powered in Sx/MOFF or Deep Sx while on battery. This bit is reset by RTCRST# assertion.
29	1b RW	**Deep Sx PHY Power Disable (DSX_PP_DIS):** This bit determines the Host software contribution to whether the LAN PHY remains powered in Deep Sx. If this bit is cleared, for the PHY to be powered in Deep Sx state, SX_PP_EN must be set to '1'. This bit is reset by RTCRST# assertion.
28	0b RW	**After G3 PHY Power Enable (AG3_PP_EN):** This bit determines the Host software contribution to whether the LAN PHY is powered up after exiting G3 (to either Sx/MOFF or Deep Sx). This bit is reset by RTCRST# assertion.
27	0b RW	**Sx PHY Power Enable (SX_PP_EN):** This bit determines the Host software contribution to whether the LAN PHY remains powered in an Sx/MOFF state that was entered from S0 (rather than from G3). This bit is reset by DSW_PWROK de-assertion.
26	0b RO	**Reserved**
25	0b RW/L	**Non Legacy PCH Mode (NON_LEG_PCH_MODE):** When this bit is set to 1b, the PCH is in non-legacy PCH Mode
24	0b RW/L	**MultiPCH Reset Sync Enable (MPCH_RST_SYNC_EN):** When this bit is set to 1b the PCH will synchronize the resets between a legacy and non-legacy PCH.
23	0h RW	**DRAM Initialization Scratchpad Bit (DISB):** This bit does not effect hardware functionality in any way. It is provided as a scratchpad bit that is maintained through main power well resets and CF9h-initiated resets. BIOS is expected to set this bit prior to starting the DRAM initialization sequence and to clear this bit after completing the DRAM initialization sequence. BIOS can detect that a DRAM initialization sequence was interrupted by a reset by reading this bit during the boot sequence. If the bit is 1, then the DRAM initialization was interrupted. This bit is reset by RSMRST# assertion.
22	0h RO	Reserved
21	0h RO/V	**Memory Placed in Self-Refresh (MEM_SR):** This bit will be set to '1' if DRAM should have remained powered and held in Self-Refresh through the last power state transition (i.e. the last time the system left S0). The scenarios where this should be the case are: - successful S3 entry / exit - successful Host partition reset without power cycle These scenarios both involve a handshake between the PCH and the Processor. The acknowledge from the Processor back to the PCH is assumed to imply that memory was successfully placed into Self-Refresh (the PCH has no way to verify whether that actually occurred). This bit will be cleared whenever the PCH begins a transition out of S0. ***Note:*** This bit is meaningless when recovering from a Demoted Warm Reset (DWR)
20:19	0h RO	Reserved
18	0h RW/1C/V	**Minimum SLP_S4# Assertion Width Violation Status (MS4V):** Hardware sets this bit when the SLP_S4# assertion width is less than the time programmed in the SLP_S4# Minimum Assertion Width field (D31.F0.A4h.5:4). The PCH begins the timer when SLP_S4# pin (including ME override logic) is asserted during S4/S5 entry, or when the RSMRST# input is deasserted during Primary well power-up. ***Notes:*** 1. The status bit is cleared by software writing a 1 to the bit. 2. This bit is functional regardless of the value in the "SLP_S4# Assertion Stretch Enable" and the "Disable SLP_X Stretching After SUS Power Failure" bits. 3. This bit is reset by the assertion of the RSMRST# pin, but can be set in some cases before the default value is readable.
17	0h RO	Reserved
16	0h RW/1C/V	**Global Reset Status (GBL_RST_STS):** This bit is set after a global reset (not G3 or Deep Sx) occurs. See the GEN_PMCON_B.HOST_RST_STS bit for potential usage models. The status bit is cleared by software writing a 1 to the bit.

Bit Range	Default & Access	Field Name (ID): Description
15	0h RW	**Allow iCLK PLL Shutdown in C0 (ALLOW_ICLK_PLL_SD_INC0):** 0 = PMC allows iCLK PLL Shutdown only when the CPU is in a non-C0 (Cx) state. (Default) 1 = iCLK PLL can be shut down when the CPU is in C0 or Cx state assuming all other gating conditions are satisfied. This bit is reset by PLTRST# assertion.
14	0h RW	**Dynamic mPHY CRI Clock Gate Override (MPHY_CRICLK_GATE_OVR):** 0 = Enable CRI clock gating in the modPHY. (Default) 1 = Disable CRI clock gating in the modPHY This bit is reset by PLTRST# assertion.
13	0h RW	**Allow OPI PLL Shutdown in C0 (ALLOW_OPI_PLL_SD_INC0):** 0 = PMC allows OPI PLL Shutdown only when the CPU is in a non-C0 (Cx)state. (Default) 1 = OPI PLL can be shut down when the CPU is in C0 or Cx state assuming all other gating conditions are satisfied. ***Note:*** The signal driven by this configuration bit is connected to both OPI and USB3/PXP PLLs to support PLL shutdown when backbone clock is sourced from the USB3/PXP PLL (a backup mode). This bit is reset by PLTRST# assertion.
12	0h RW	**Allow SPXB Clock Gating in C0 (ALLOW_SPXB_CG_INC0):** 0 = PMC allows SBXB backbone clock gating only when the CPU is in a non-C0 state (Cx). (Default) 1 = SPXB backbone clock can be gated when the CPU is in C0 or Cx state assuming all other gating conditions are satisfied. This bit is reset by PLTRST# assertion.
11	0h RO	Reserved
10	0h RW	**BIOS PCI Express Enable (BIOS_PCI_EXP_EN):** This bit acts as a global enable for the SCI associated with the PCI express ports. 0 = The various PCI Express* ports and processor CANNOT cause PCI_EXP_STS bit to go active. (Default) 1 = The various PCI Express* ports and processor CAN cause PCI_EXP_STS bit to go active. This bit is reset by PLTRST# assertion.
9	0h RO/V	**Power Button Level (PWRBTN_LVL):** This read-only bit indicates the current state of the PWRBTN# signal. 1= High, 0 = Low. The value reflected in this bit is dependent upon PM_CFG1.PB_DB_MODE. The PB_DB_MODE bit's value causes the following behavior: - '0': PWRBTN_LVL is taken from the debounced PWRBTN# pin value that is seen at the output of a 16ms debouncer. - '1': PWRBTN_LVL is taken from the raw PWRBTN# pin (before the debouncer).
8	0h RO	Reserved
7	0h RW	**Allow L1.LOW Entry During C0 (ALLOW_L1LOW_C0):** 0 = PMC only allows L1.LOW entry if the CPU is in a non-C0 state (Cx). (Default) 1 = PMC allows L1.LOW entry in C0 or Cx state. This bit is reset by PLTRST# assertion.
6	0h RW	**Allow L1.LOW Entry with OPI Voltage On (ALLOW_L1LOW_OPI_ON):** 0 = PMC only allows L1.LOW entry if the OPI voltage is off. (Default) 1 = PMC allows L1.LOW entry regardless of whether the OPI voltage is on/off. This bit is reset by PLTRST# assertion.
5	0h RW	**Allow L1.LOW Entry with CPU BCLK REQ Asserted (ALLOW_L1LOW_BCLKREQ_ON):** 0 = PMC only allows L1.LOW entry if the CPU's BCLK request is de-asserted.(Default) 1 = PMC allows L1.LOW entry regardless of whether the CPU's BCLK request is asserted/de-asserted. This bit is reset by PLTRST# assertion.
4	0h RW/L	**SMI Lock (SMI_LOCK):** When this bit is set, writes to the GLB_SMI_EN bit will have no effect. Once the SMI_LOCK bit is set, writes of '0' to SMI_LOCK bit will have no effect (i.e., once set, this bit can only be cleared by PLTRST# assertion).

Bit Range	Default & Access	Field Name (ID): Description
3	0b RW/L	**ESPI SMI LOCK (ESPI_SMI_LOCK):** When this bit is set to 1b, writes to the ESPI_SMI_EN bit will have not effect. Once the ESPI_SMI_LOCK bit is set, writes of 0b to it will have no effect. Only a PLTRST# will reset this to 0b.
2	0b RO	Reserved
1:0	0h RW	**Period SMI Select (PER_SMI_SEL):** Software sets these bits to control the rate at which the periodic SMI# is generated: 00 = 64 seconds (Default), 01 = 32 seconds, 10 = 16 seconds, 11 = 8 seconds Tolerance for the timer is ± 1 second. This field is reset by PLTRST# assertion.

9.8.11 General PM Configuration B (GEN_PMCON_B)—Offset A4h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 4006h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0
RSVD													SLPSX_STR_POL_LOCK	ACPI_BASE_LOCK	PM_DATA_BAR_DIS	PME_B0_S5_DIS	SUS_PWR_FLR	WOL_EN_OVRD	DIS_SLP_X_STRCH_SUS_UP	SLP_S3_MIN_ASST_WDTH		HOST_RST_STS	RSVD	SWSMI_RATESEL		S4MAW		S4ASE	RTC_PWR_STS	PWR_FLR	AG3E

Bit Range	Default & Access	Field Name (ID): Description
31:19	0h RO	Reserved
18	0h RW/L	**SLP_Sx# Stretching Policy Lock-Down (SLPSX_STR_POL_LOCK):** When set to 1, this bit locks down the following fields: (Those bits become read-only.) - GEN_PMCON_3.DIS_SLP_X_STRCH_SUSPF - GEN_PMCON_3.SLP_S3_MIN_ASST_WDTH - GEN_PMCON_3.S4MAW - GEN_PMCON_3.S4ASE - PM_CFG.SLP_A_MIN_ASST_WDTH - PM_CFG.SLP_GBE_MIN_ASST_WDTH - PM_CFG.PWR_CYC_DUR Writes of '0' to this bit are always ignored. This bit becomes locked when a value of '1' is written to it. Once locked, the only way to clear this bit is to perform a platform reset (PLTRST# assertion).
17	0h RW/L	**ACPI Base Lock (ACPI_BASE_LOCK):** When set to 1, this bit locks down the ACPI Base Address Register (ABASE) at offset 40h. The Base Address Field becomes read-only. Writes of '0' to this bit are always ignored. This bit becomes locked when a value of '1' is written to it. Once locked, the only way to clear this bit is to perform a platform reset (PLTRST# assertion).

Bit Range	Default & Access	Field Name (ID): Description
16	0h RW/L	**Power Management Data BAR Disable (PM_DATA_BAR_DIS):** When set to 1, this bit disables all accesses to the MMIO range pointed to by PM_DATA_BAR. This does not affect the BAR value itself, which can still be changed after this bit is set. But once the bit is set to '1', the PMC will drop writes to the data region pointed to by PM_DATA_BAR. And reads will return 0. Writes of '0' to this bit are always ignored. This bit becomes locked when a value of '1' is written to it. Once locked, the only way to clear this bit is to perform a platform reset (PLTRST# assertion).
15	0h RW	**PME B0 S5 Disable (PME_B0_S5_DIS):** When set to '1', this bit blocks wake events from PME_B0_STS in S5, regardless of the state of PME_B0_EN. When cleared (default), wake events from PME_B0_STS are allowed in S5 if PME_B0_EN = '1'. Wakes from power states other than S5 are not affected by this policy bit. The net effect of setting PME_B0_S5_DIS = '1' is described by the truth table below: Y = Wake; N = Don't wake; B0 = PME_B0_EN; OV = WOL Enable Override B0/OV | S3/S4 | S5 00 | N | N 01 | N | Y (LAN only) 11 | Y (all PME B0 sources) | Y (LAN only) 10 | Y (all PME B0 sources) | N This bit is cleared by the RTCRST# assertion.
14	1h RW/1C	**SUS Well Power Failure (SUS_PWR_FLR):** This bit is set to '1' whenever Primary well power is lost, as indicated by RSMRST# assertion. Software writes a 1 to this bit to clear it. This bit is in the Primary well, and defaults to '1' based on RSMRST# assertion (not cleared by any type of reset).
13	0h RW	**WOL Enable Override (WOL_EN_OVRD):** 0 = WoL polices are determined by OS-visible bits. 1 = WoL is enabled enabled to wake the system from S5 regardless of the value in the PME_B0_EN bit in the GPE0_EN register. This allows the system BIOS to enable WoL regardless of the policies selected through the OS. This bit is maintained in the RTC power well, therefore permitting WOL following a surprise power failure even in cases in which the system may have been running in S0 without the PME Enables set. (Note that the LAN NVRAM configuration must support WOL after SUS power loss.) This bit has no effect on wakes from S1, S3, or S4. This bit is cleared by the RTCRST# assertion.
12	0h RW/L	**Disable SLP_X Stretching After SUS Well Power Up (DIS_SLP_X_STRCH_SUS_UP):** 0 = SLP_* stretching will be performed after SUS power failure as enabled in various other fields. SLP_* stretch timers start on SUS well power up (the PCH has no ability to count stretch time while the SUS well is powered down). 1 = All SLP_* pin stretching is disabled when powering up after a SUS well power loss. ***Notes:*** This policy bit has a different effect on SLP_SUS# stretching than on the other SLP_* pins, since SLP_SUS# is the control signal for one of the scenarios where SUS well power is lost (Deep Sx). The effect of setting this bit to '1' on: - SLP_S3#, SLP_S4#, SLP_A# and SLP_GBE# stretching: disabled after any SUS power loss - SLP_SUS# stretching: disabled after G3, but no impact on Deep Sx This field is not writable when the SLP_Sx# Stretching Policy Lock-Down bit is set. This bit is cleared by the RTCRST# assertion.
11:10	0h RW/L	**SLP_S3# Minimum Assertion Width (SLP_S3_MIN_ASST_WDTH):** This 2-bit value indicates the minimum assertion width of the SLP_S3# signal to guarantee that the Main power supplies have been fully power-cycled. This value may be modified per platform depending on power supply capacitance, board capacitance, power failure detection circuits, etc. Settings are: 00: 60 usec 01: 1 ms 10: 50 ms 11: 2 sec This field is not writable when the SLP_Sx# Stretching Policy Lock- Down bit is set. This bit is cleared by the RSMRST# pin.
9	0h RW/1C/V	**Host Reset Status (HOST_RST_STS):** This bit is set by hardware when a host partition reset (not a global reset, Deep Sx, or G3) occurs. This bit is an optional tool to help BIOS determine when a host partition reset might have collided with a wake from a valid sleep state. A possible usage model would be to consult and then write a '1' to clear this bit during the boot flow before determining what action to take based on reading PM1_STS.WAK_STS = '1'. If HOST_RST_STS = '1' and/or GEN_PMCON_A.GBL_RST_STS = '1', the cold reset boot path could be followed rather than the resume path, regardless of the setting of WAK_STS. This bit does not affect PCH operation in any way, and can therefore be left set if BIOS chooses not to use it.

Bit Range	Default & Access	Field Name (ID): Description
8	0h RO	Reserved
7:6	0h RW	**SWSMI Rate Select (SWSMI_RATESEL):** This 2-bit value indicates when the SWSMI timer will time out. Valid values are: 00 [=] 1.5 ms ± 0.6 ms 01 [=] 16 ms ± 4 ms 10 [=] 32 ms ± 4 ms 11 [=] 64 ms ± 4 ms These bits are not cleared by any type of reset except RTCRST# assertion.
5:4	0h RW/L	**SLP_S4# Minimum Assertion Width (S4MAW):** This 2-bit value indicates the minimum assertion width of the SLP_S4# signal to guarantee that the DRAMs have been safely power-cycled. This value may be modified per platform depending on DRAM types, power supply capacitance, etc. Valid values are: 11: 1 second 10: 2 seconds 01: 3 seconds 00: 4 seconds This value is used in two ways: 1. If the SLP_S4# assertion width is ever shorter than this time, a status bit (D31.F0.A2h.2) is set for BIOS to read when S0 is entered 2. If enabled by bit 3 in this register, the hardware will prevent the SLP_S4# signal from deasserting within this minimum time period after asserting. Note that the logic that measures this time is in the Primary well. Therefore, when leaving a G3 or Deep Sx state, the minimum time is measured from the deassertion of the internal Primary well reset (unless the "Disable_SLP_X Stretching After SUS Power Failure" bit is set). This field is not writable when the SLP_Sx# Stretching Policy Lock-Down bit is set. RTCRST# forces this field to the conservative default state (00b).
3	0h RW/L	**SLP_S4# Assertion Stretch Enable (S4ASE):** 0 = The minimum assertion time for SLP_S4# 1 = The SLP_S4# pin (which includes the ME override logic) will minimally assert for the time specified in bits 5:4 of this register. This bit is not writable when the SLP_Sx# Stretching Policy Lock-Down bit is set. This bit is cleared by RTCRST# assertion.
2	1h RW	**RTC Power Status (RTC_PWR_STS):** Intel PCH will set this bit to '1' when RTCRST# indicates a weak or missing battery. The bit will remain set until the software clears it by writing a '0' back to this bit position. This bit is not cleared by any type of reset.
1	1h RW/1C	**Power Failure (PWR_FLR):** This bit is in the DSW well and defaults to '1' based on DSW_PWROK assertion (not cleared by any type of reset). 0 = Indicates that the trickle current has not failed since the last time the bit was cleared. 1 = Indicates that the trickle current (from the main battery or trickle supply) was removed or failed. Software writes a '1' to this bit to clear it.
0	0h RW	**AFTERG3_EN (AG3E):** Determines what state to go to when power is reapplied after a power failure (G3 state). 0 = System will return to an S0 state (boot) after power is re-applied. 1 = System will return to the S5 state (except if it was in S4, in which case it will return to S4-like state). In the S5 state, the only enabled wake-up event is the Power Button or any enabled wake event that was preserved through the power failure. This bit is in the RTC well and is only cleared by RTCRST# assertion.

9.8.12 BM_BREAK_EN and Cx State Configuration Register (BM_CX_CNF)—Offset A8h

Usage ACPI, Legacy. Power well Primary. This register is used to enable BM break events and C-state related modes. Note that these C-state settings have no effect if the C-states are not entered.

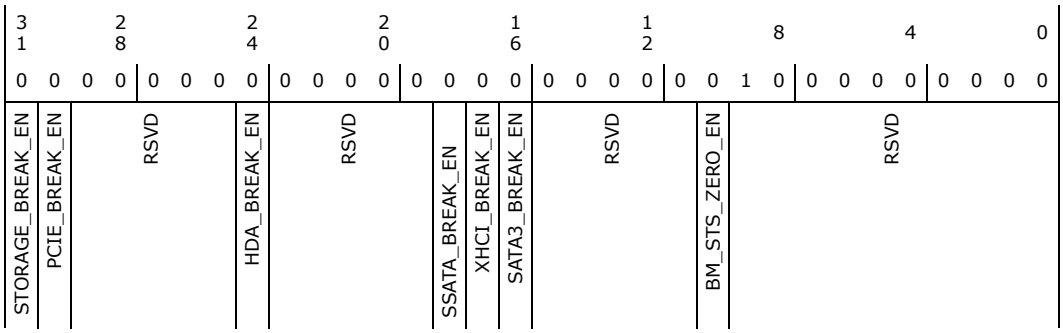
Access Method

Type: CFG Register (Size: 32 bits)

Device: 31

Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31	0h RW	**Storage Break Enable (STORAGE_BREAK_EN):** When this bit is a 1, Serial ATA traffic will cause BM_STS to be set, even if the BM_STS_ZERO_EN bit is set. This bit is reset by PLTRST# assertion.
30	0h RW	**PCI Express Break Enable (PCIE_BREAK_EN):** When this bit is a 1, PCI Express traffic will cause BM_STS to be set, even if the BM_STS_ZERO_EN bit is set. This bit is reset by PLTRST# assertion.
29:25	0h RO	Reserved
24	0h RW	**HD Audio Break Enable (HDA_BREAK_EN):** When this bit is a 1, HD Audio traffic will cause BM_STS to be set, even if the BM_STS_ZERO_EN bit is set. This bit is reset by PLTRST# assertion.
23:18	0h RO	Reserved
18	0b RW	**SSATA Break Enable (SSATA_BREAK_EN):** When this bit is set to 1b, SSATA traffic will cause BM_STS to be set, even if the BM_STS_ZERO_EN bit is set.
17	0h RW	**XHCI Break Enable (XHCI_BREAK_EN):** When this bit is a 1, XHCI traffic will cause BM_STS to be set, even if the BM_STS_ZERO_EN bit is set. This bit is reset by PLTRST# assertion.
16	0h RW	**SATA3 Break Enable (SATA3_BREAK_EN):** When this bit is a 1, SATA3 traffic will cause BM_STS to be set, even if the BM_STS_ZERO_EN bit is set. This bit is reset by PLTRST# assertion.
15:11	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
10	0h RW	**Bus Master Status Zero Enable (BM_STS_ZERO_EN):** When this bit is a 1, PCH will not set the BM_STS if there is bus master activity from PCI, PCI Express and internal bus masters. ***Note:*** If the BM_STS bit is already set when the BM_STS_ZERO_EN bit is set, the BM_STS bit will remain set. Software will still need to clear the BM_STS bit. ***Note:*** BM_STS_ZERO_EN does not apply to PHOLD (LPC DMA or LPC bus master activity). A separate policy bit (PHOLD_BM_STS_BLOCK) determines whether PHOLD will set BM_STS. This bit is reset by PLTRST# assertion.
9	1b RO	Reserved
8:0	0h RO	Reserved

9.8.13 Extended Test Mode Register 3 (ETR3)—Offset ACh

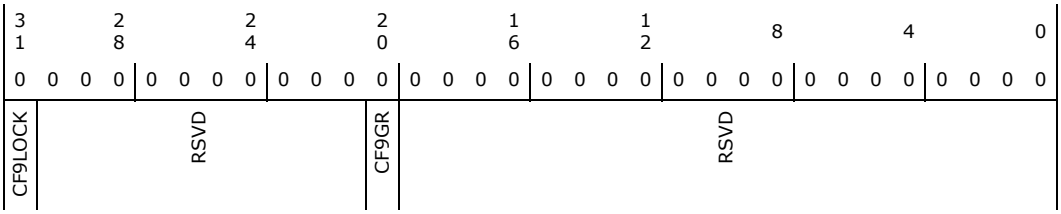
This register resides in the primary well. All bits except bit[23:16] are reset by host_deep_rst_b. Bit[23:16] are reset by pri_pwrgood_rst_b only.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31	0h RW/V/L	**CF9h Lockdown (CF9LOCK):** 0 = CF9h Global Reset bit R/W. 1 = CF9h Global Reset bit RO. When set, this bit becomes RO and is reset by a CF9h reset or RSMRST# assertion (other reset types are not applicable). In manufacturing/debug environments this bit should be left as default '0'. In all other environments, BIOS must program this bit to '1'.
30:21	0h RO	Reserved.
20	0h RW/L	**CF9h Global Reset (CF9GR):** 0 = A CF9h write of 6h or Eh will only reset the Host partition. 1 = A CF9h write of 6h or Eh will cause a Global Reset of both the Host and the ME partitions. It is recommended that BIOS should set this bit early on in the boot sequence, and then clear it and set the CF9LOCK bit prior to loading the OS in both an ME Enabled and a ME Disabled system. This register is locked by the CF9 Lockdown (CF9LOCK) bit. This register is not reset by a CF9h reset. It is reset by RSMRST# assertion.
19:0	0h RO	Reserved.

9.9 PMC I/O Based Registers Summary

The below table shows registers associated with ACPI and Legacy power management support. These register locations are all offsets from ACPI base address defined in PCI Device 31: Function 2 (ABASE), and can be moved to any 256-byte aligned I/O location. In order to access these registers, the ACPI Enable bit in ACPI Control Register (B0:D31:F2 offset 44h) must be set. The registers are defined to support the ACPI 4.0a specification and generally use the same bit names.

Note: All reserved bits and registers will always return 0 when read and will have no effect when written.

Table 9-18. Summary of PMC I/O Based Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	Power Management 1 Enables and Status (PM1_EN_STS)—Offset 0h	0h
4h	7h	Power Management 1 Control (PM1_CNT)—Offset 4h	0h
8h	Bh	Power Management 1 Timer (PM1_TMR)—Offset 8h	0h
30h	33h	SMI Control and Enable (SMI_EN)—Offset 30h	2h
34h	37h	SMI Status Register (SMI_STS)—Offset 34h	0h
40h	43h	General Purpose Event Control (GPE_CTRL)—Offset 40h	0h
44h	47h	Device Activity Status Register (DEVACT_STS)—Offset 44h	0h
50h	53h	PM2a Control Block (PM2A_CNT_BLK)—Offset 50h	0h
80h	83h	General Purpose Event 0 Status [31:0] (GPE0_STS_31_0) - Offset 80h	0h
84h	87h	General Purpose Event 0 Status [63:32] (GPE0_STS_63_32) - Offset 84h	0h
88h	8Bh	General Purpose Event 0 Status [95:64] (GPE0_STS_95_64) - Offset 88h	0h
8Ch	8Fh	General Purpose Event 0 Status [127:96] (GPE0_STS_127_96) - Offset 8Ch	0h
90h	93h	General Purpose Event 0 Enable [31:0] (GPE0_EN_31_0)—Offset 90h	0h
94h	97h	General Purpose Event 0 Enable [63:32] (GPE0_EN_31_0)—Offset 94h	0h
98h	9Ah	General Purpose Event 0 Enable [95:64] (GPE0_EN_31_0)—Offset 98h	0h
9Ch	9Fh	General Purpose Event 0 Enable [127:96] (GPE0_EN_127_96)—Offset 9Ch	0h

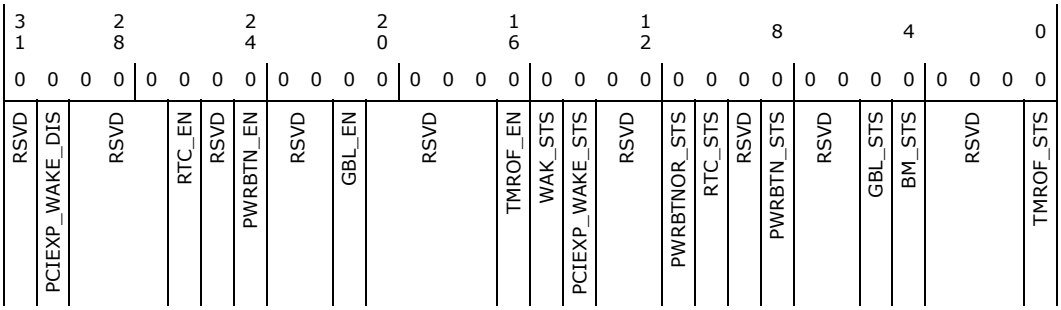
9.9.1 Power Management 1 Enables and Status (PM1_EN_STS)—Offset 0h

Access Method

Type: IO Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31	0h RO	Reserved
30	0h RW	**PCI Express Wake Disable (PCIEXP_WAKE_DIS):** This bit disables the inputs to the PCIEXP_WAKE_STS bit in the PM1 Status register from waking the system. Modification of this bit has no impact on the value of the PCIEXP_WAKE_STS bit. This bit is reset by DSW_PWROK de-assertion.
29:27	0h RO	Reserved
26	0h RW/V	**RTC Alarm Enable (RTC_EN):** This is the RTC alarm enable bit. It works in conjunction with the SCI_EN bit: RTC_EN SCI_EN Effect when RTC_STS is set 0 X No SMI# or SCI. If system was in S3-S5, no wake even occurs. 1 0 SMI#. If system was in S3-S5, then a wake event occurs before the SMI#. 1 1 SCI. If system was in S3-S5, then a wake event occurs before the SCI. ***Note:*** This bit is in the RTC well and is reset by RTCRST# assertion, to allow an RTC event to wake after a power failure.
25	0h RO	Reserved
24	0h RW/V	**Power Button Enable (PWRBTN_EN):** This bit is the power button enable. It works in conjunction with the SCI_EN bit: PWRBTN_EN SCI_EN Effect when PWRBTN_STS is set 0 X No SMI# or SCI. 1 0 SMI#. 1 1 SCI. ***Note:*** PWRBTN_EN has no effect on the PWRBTN_STS bit being set by the assertion of the power button. The Power Button is always enabled as a Wake event.
23:22	0h RO	Reserved
21	0h RW	**Global Enable (GBL_EN):** Global enable bit. When both the GBL_EN and the GBL_STS are set, PCH generates an SCI. This bit is reset by PLTRST# assertion.
20:17	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
16	0h RW	**Timer Overflow Interrupt Enable (TMROF_EN):** This is the timer overflow interrupt enable bit. It works in conjunction with the SCI_EN bit: TMROF_EN SCI_EN Effect when TMROF_STS is set 0 X No SMI# or SCI. 1 0 SMI#. 1 1 SCI. This bit is reset by PLTRST# assertion.
15	0h RW/1C/V	**Wake Status (WAK_STS):** This bit is set when the system is in one of the Sleep states (via the SLP_EN bit) and an enabled Intel PCH Wake event occurs. Upon setting this bit, the Intel PCH will transition the system to the ON state. This bit can only be set by hardware and can only be cleared by writing a one to this bit position. This bit is not affected by hard resets caused by a CF9 write, but is reset by RSMRST#. If a power failure occurs (such as removed batteries) without the SLP_EN bit set, the WAK_STS bit will not be set when the power returns if the AFTER_G3 bit is 0. If the AFTER_G3 bit is 1, then the WAK_STS bit will be set after waking from a power failure. If necessary, the BIOS can clear the WAK_STS bit in this case.
14	0h RW/1C/V	**PCI Express Wake Status (PCIEXP_WAKE_STS):** This bit is set by hardware to indicate that the system woke due to a PCI Express wakeup event. This event can be caused by the PCI Express WAKE# pin being active, or one or more of the PCI Express ports being in beacon state, or receipt of a PCI Express PME message at root port. This bit should only be set when one of these events causes the system to transition from a non-S0 system power state to the S0 system power state. This bit is set independent of the PCIEXP_WAKE_DIS bit. Software writes a 1 to clear this bit. If WAKE# pin is still active during the write or one or more PCI Express ports is in the beacon state or PME message received indication is not cleared in the root port, then the bit will remain active (i.e. all inputs to this bit are level sensitive). ***Note:*** This bit does not itself cause a wake event or prevent entry to a sleeping state. Thus if the bit is 1 and the system is put into a sleeping state, the system will not automatically wake.
13:12	0h RO	Reserved
11	0h RW/1C/V	**Power Button Override (PWRBTNOR_STS):** This bit is set any time a Power Button Override Event occurs (i.e. the power button is pressed for at least 4 consecutive seconds), the corresponding bit is received in the SMBus slave message, the ME-Initiated Power Button Override bit is set, the ME-Initiated Host Reset with Power Down is set, or due to an internal thermal sensor catastrophic condition. These events cause an unconditional transition to the S5 state. The BIOS or SCI handler clears this bit by writing a 1 to it. This bit is not affected by hard resets via CF9h writes, and is not reset by RSMRST#. Thus, this bit is on RTC well and is preserved through power failures (reset by RTCRST#). Note that this bit is still asserted when the global SCI_EN is set to '1' then an SCI will be generated.
10	0h RW/1C/V	**RTC Status (RTC_STS):** This bit is set when the RTC generates an alarm (assertion of the IRQ8# signal), and is not affected by any other enable bit. See RTC_EN for the effect when RTC_STS goes active. This bit is only set by hardware and can only be reset by writing a one to this bit position. This bit is not affected by hard resets caused by a CF9 write, but is reset by DSW_PWROK de-assertion.
9	0h RO	Reserved
8	0h RW/1C/V	**Power Button Status (PWRBTN_STS):** This bit is set when the PWRBTN# signal is asserted (low), independent of any other enable bit. See PWRBTN_EN for the effect when PWRBTN_STS goes active. PWRBTN_STS is always a wake event. This bit is only set by hardware and can be cleared by software writing a one to this bit position. This bit is not affected by hard resets caused by a CF9 write, but is reset by DSW_PWROK de-assertion. If the PWRBTN# signal is held low for more than 4 seconds, the Intel PCH clears the PWRBTN_STS bit, sets the PWRBTNOR_STS bit, the system transitions to the S5 state, and only PWRBTN# is enabled as a wake event. If PWRBTN_STS bit is cleared by software while the PWRBTN# pin is still held low, this will not cause the PWRBTN_STS bit to be set. The PWRBTN# signal must go inactive and active again to set the PWRBTN_STS bit. Note that the SMBus Unconditional Powerdown message, the CPU Thermal Trip and the Internal Thermal Sensors' Catastrophic Condition result in behavior matching the Powerbutton Override, which includes clearing this bit.
7:6	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
5	0h RW/1C/V	**GBL Status (GBL_STS):** This bit is set when an SCI is generated due to the BIOS wanting the attention of the SCI handler. BIOS has a corresponding bit, BIOS_RLS, which will cause an SCI and set this bit. The SCI handler should then clear this bit by writing a 1 to it. This bit will not cause wake events or SMI#. This bit is not effected by SCI_EN. ***Note:*** GBL_STS being set will cause an SCI, even if the SCI_EN bit is not set. Software must take great care not to set the BIOS_RLS bit (which causes GBL_STS to be set) if the SCI handler is not in place. This bit is reset by PLTRST# assertion.
4	0h RW/1C/V	**Bus Master Status (BM_STS):** This bit is set to 1 by the Intel PCH when a PCH-visible bus master requests access to memory or the BM_BUSY# signal is active. This bit is cleared by the Processor writing a 1 to this bit position. This bit will not cause a wake event, SCI, or SMI. This bit is reset by PLTRST# assertion.
3:1	0h RO	Reserved
0	0h RW/1C/V	**Timer Overflow Status (TMROF_STS):** This is the timer overflow status bit. This bit gets set anytime bit 22 of the 24 bit timer goes low (bits are counted from 0 to 23). This will occur every 2.3435 seconds. See TMROF_EN for the effect when TMROF_STS goes active. Software clears this bit by writing a 1 to it. This bit is reset by PLTRST# assertion.

9.9.2 Power Management 1 Control (PM1_CNT)—Offset 4h

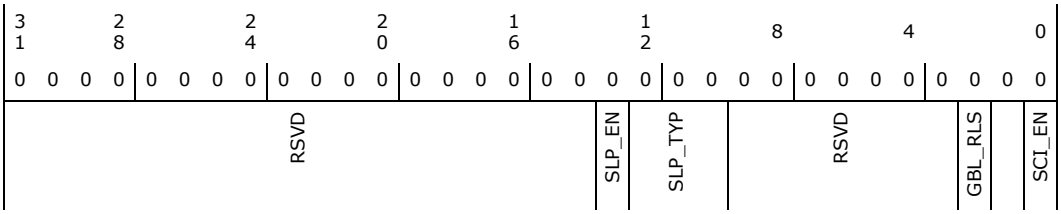
Lockable: No Usage: ACPI or Legacy Power Well: Bits 0-9, 13-31: Primary, Bits 10-12: RTC

Access Method

Type: IO Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13	0h WO	**Sleep Enable (SLP_EN):** This is a write-only bit and reads to it always return a zero. Setting this bit causes the system to sequence into the Sleep state defined by the SLP_TYP field. This bit is reset by PLTRST# assertion.
12:10	0h RW	**Sleep Type (SLP_TYP):** This 3-bit field defines the type of Sleep the system should enter when the SLP_EN bit is set to 1. Bits Mode Typical Mapping 000 ON S0 001 Reserved 010 Reserved 011 Reserved 100 Reserved 101 Suspend-To-RAM S3 110 Suspend-To-Disk S4 111 Soft Off S5 These bits are reset by RTCRST# only.
9:3	0h RO	Reserved
2	0h WO	**Global Release (GBL_RLS):** This bit always reads as 0. ACPI software writes a '1' to this bit to raise an event to the BIOS. BIOS software has corresponding enable and status bits to control its ability to receive ACPI events.
1	0h RW	**Scratchpad**
0	0h RW	**SCI Enable (SCI_EN):** Selects the SCI interrupt or the SMI# for various events. 0 = These events will generate an SMI#. 1 = These events will generate an SCI. This bit is reset by PLTRST# assertion.

9.9.3 Power Management 1 Timer (PM1_TMR)—Offset 8h

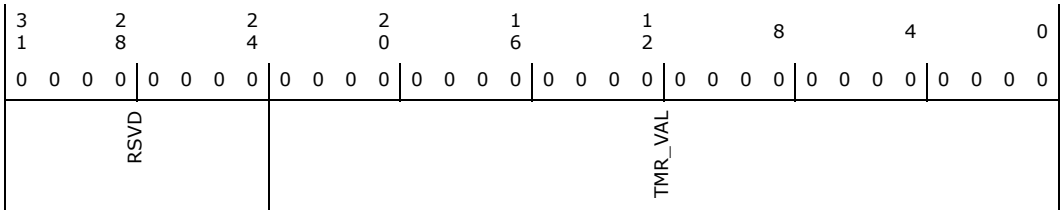
Lockable: No Usage: ACPI Power Well: Primary

Access Method

Type: IO Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23:0	0h RO/V	**Timer Value (TMR_VAL):** This read-only field returns the running count of the PM timer. This counter runs off a 3.579545 MHz clock (derived from 14.31818 MHz divided by 4). It is reset (to 0) during a PCI reset, and then continues counting as long as the system is in the S0 state. Anytime bit 22 of the timer goes HIGH to LOW (bits referenced from 0 to 23), the TMROF_STS bit is set. The High-to-Low transition will occur every 2.3435 seconds. Writes to this register have no effect.

9.9.4 SMI Control and Enable (SMI_EN)—Offset 30h

Lockable: No
Usage: ACPI or Legacy
Power Well: Primary

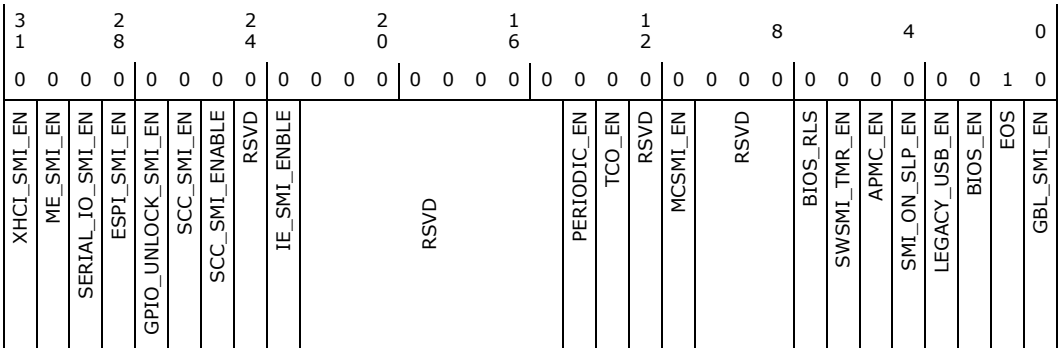
Note: This register is symmetrical to the SMI Status Register.

Access Method

Type: IO Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 2h



Bit Range	Default & Access	Field Name (ID): Description
31	0h RW	**XHCI SMI Enable (XHCI_SMI_EN):** Software sets this bit to enable XHCI SMI events. This bit is reset by PLTRST# assertion.
30	0h RW	**ME SMI Enable (ME_SMI_EN):** Software sets this bit to enable ME SMI# events. This bit is reset by PLTRST# assertion.
29	0h RW	**Intel Serial I/O SMI Enable (SERIAL_IO_SMI_EN):** Software sets this bit to enable Intel Serial I/O SMI events. This bit is reset by PLTRST# assertion.
28	0h RW	**eSPI SMI Enable (ESPI_SMI_EN):** Software sets this bit to enable eSPI SMI events. This bit is reset by PLTRST# assertion.
27	0h RW/1S	**GPIO Unlock SMI Enable (GPIO_UNLOCK_SMI_EN):** Setting this bit will cause the Intel PCH to generate an SMI# when the GPIO_UNLOCK_SMI_STS bit is set in the SMI_STS register. Once written to '1', this bit can only be cleared by PLTRST# assertion.
26	0h RW	**SCC SMI Enable (SCC_SMI_EN):** Software sets this bit to enable SCC SMI events. This bit is reset by PLTRST# assertion.
25	0b RW	**SCC SMI Enable (SCC_SMI_EN)**: Software sets this bit to enable SCC SMI events
24	0b RO	**Reserved**
23	0b RW	**IE SMI Enable (IE_SMI_EN)**: Software sets this bit to enable IE SMI events.
22:15	0h RO	**Reserved**

Bit Range	Default & Access	Field Name (ID): Description
14	0h RW	**Periodic Enable (PERIODIC_EN):** Setting this bit will cause the Intel PCH to generate an SMI# when the PERIODIC_STS bit is set in the SMI_STS register. This bit is reset by PLTRST# assertion.
13	0h RW/L	**TCO Enable (TCO_EN):** 1 = Enables the TCO logic to generate SMI#. 0 = Disables TCO logic from generating an SMI#. If the NMI2SMI_EN bit is set, then SMIs that are caused by NMIs (i.e. rerouted) will not be gated by the TCO_EN bit. Even if the TCO_EN bit is 0, the NMIs will still be routed to cause the SMI#. ***Note:*** This bit can not be written once the TCO_LOCK bit (at offset 08h of TCO I/O Space) is set. This prevents unauthorized software from disabling the generation of TCO-based SMIs. This bit is reset by PLTRST# assertion.
12	0h RO	Reserved
11	0h RW	**Microcontroller SMI Enable (MCSMI_EN):** Software sets this bit to 1 to enables Intel PCH to trap access to the microcontroller range (62h or 66h). A 'trapped' cycles will be claimed by Intel PCH, but not forwarded to LPC. An SMI# will also be generated. This bit is reset by PLTRST# assertion.
10:8	0h RO	Reserved
7	0h WO	**BIOS Release (BIOS_RLS):** Enables the generation of an SCI interrupt for ACPI software when a '1' is written to this bit position by BIOS software. This bit always reads a '0'. ***Note:*** GBL_STS being set will cause an SCI, even if the SCI_EN bit is not set. Software must take great care not to set the BIOS_RLS bit (which causes GBL_STS to be set) if the SCI handler is not in place. This bit is reset by PLTRST# assertion.
6	0h RW	**Software SMI Timer Enable (SWSMI_TMR_EN):** Software sets this bit to a '1' to start the Software SMI# Timer. When the timer expires (depending on the SWSMI_RATE_SEL bits), it will generate an SMI# and set the SWSMI_TMR_STS bit. The SWSMI_TMR_EN bit will remain at 1 until software sets it back to 0. Clearing the SWSMI_TMR_EN bit before the timer expires will reset the timer and the SMI# will not be generated. The default for this bit is 0. This bit is reset by PLTRST# assertion.
5	0h RW	**APMC Enable (APMC_EN):** If set, this enables writes to the APM_CNT register to cause an SMI#. This bit is reset by PLTRST# assertion.
4	0h RW	**SMI On Sleep Enable (SMI_ON_SLP_EN):** If this bit is set, the Intel PCH will generate an SMI# when a write access attempts to set the SLP_EN bit (in the PM1_CNT register). Furthermore, the Intel PCH will not put the system to a sleep state. It is expected that the SMI# handler will turn off the SMI_ON_SLP_EN bit before actually setting the SLP_EN bit. This bit is reset by PLTRST# assertion.
3	0h RW	**Legacy USB Enable (LEGACY_USB_EN):** Enables legacy USB circuit to cause SMI#. This bit is reset by PLTRST# assertion.
2	0h RW	**BIOS Enable (BIOS_EN):** Enables the generation of SMI# when ACPI software writes a 1 to the GBL_RLS bit. Note that if the BIOS_STS bit, which gets set when software writes a 1 to GBL_RLS bit, is already a 1 at the time that BIOS_EN becomes 1, an SMI# will be generated when BIOS_EN gets set. This bit is reset by PLTRST# assertion.
1	1h RW/1S/V	**End of SMI (EOS):** This bit controls the arbitration of the SMI signal to the processor. This bit must be set in order for Intel PCH to assert SMI# low to the processor after SMI# has been asserted previously. Once Intel ICH asserts SMI# low, the EOS bit is automatically cleared. In the SMI handler, the CPU should clear all pending SMIs (by servicing them and then clearing their respective status bits), set the EOS bit, and exit SMM. This will allow the SMI arbiter to reassert SMI upon detection of an SMI event and the setting of a SMI status bit. The SMI# signal will go inactive for 4 PCI clocks. This bit is reset by PLTRST# assertion.
0	0h RW/L	**Global SMI Enable (GBL_SMI_EN):** 0 = No SMI# will be generated by PCH. 1 = Enables the generation of SMI# in the system upon any enabled SMI event. ***Note:*** When the SMI_LOCK bit is set, this bit cannot be changed. This bit is reset by PLTRST# assertion.

9.9.5 SMI Status Register (SMI_STS)—Offset 34h

Lockable: No
Usage: ACPI or Legacy
Power Well: Primary

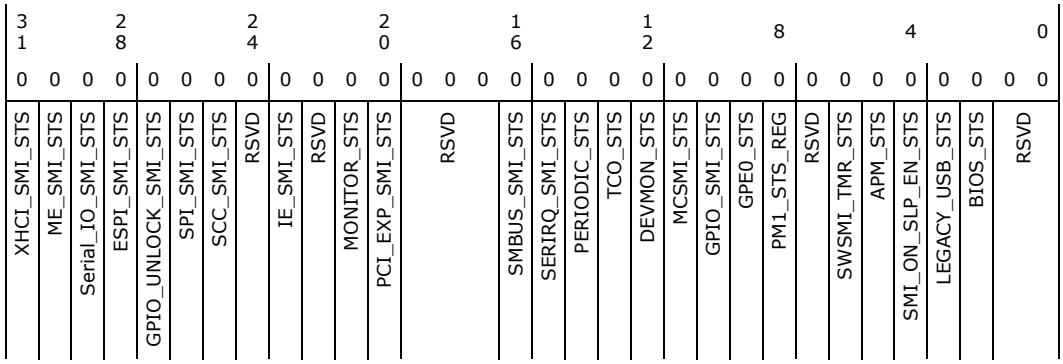
Note: If the corresponding _EN bit is set when the _STS bit is set, the Intel PCH will cause an SMI# (except bits 8-10, which don't cause SMI#).

Access Method

Type: IO Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31	0h RO/V	**XHCI SMI Status (XHCI_SMI_STS):** This bit will be set when any USB3 (XHCI) Host Controller is requesting an SMI.
30	0h RO/V	**ME SMI Status (ME_SMI_STS):** This bit will be set when Intel ME is requesting an SMI#.
29	0h RW/1C/V	**Intel Serial I/O SMI Status (Serial_IO_SMI_STS):** This bit gets set when Intel Serial I/O agent is requesting SMI #. This bit is set by hardware and cleared by software writing a 1 to this bit position.
28	0h RO/V	**eSPI SMI Status (ESPI_SMI_STS):** This bit is set if an eSPI agent is requesting an SMI#. This bit is set by hardware and cleared by the BMC sending an SMI deassertion VW.
27	0h RW/1C/V	**GPIO Unlock SMI Status (GPIO_UNLOCK_SMI_STS):** This bit will be set of the GPIO registers lockdown logic is requesting an SMI#. Writing a '1' to this bit position clears this bit to '0'. This bit is reset by PLTRST# assertion.
26	0h RO/V	**SPI SMI Status (SPI_SMI_STS):** This bit will be set when the SPI logic is requesting an SMI#. This bit is read only because the sticky status and enable bits associated with this function are located in the SPI registers.
25	0h RW/1C/V	**SCC SMI Status (SCC_SMI_STS):** This bit gets set when SCC agent is requesting SMI#.This bit is set by hardware and cleared by software writing a 1 to this bit position
24	0b RO	**Reserved**
23	0b RO/V	**IE SMI Status (IE_SMT_STS):** This bit will be set if IE is requesting an SMI#
22	0b RO	**Reserved**

Bit Range	Default & Access	Field Name (ID): Description
21	0h RO/V	**Monitor Status (MONITOR_STS):** This bit will be set if the Trap/SMI logic has caused the SMI. This will occur when the CPU or a bus master accesses an assigned register (or a sequence of accesses).
20	0h RO/V	**PCI_EXP_SMI Status (PCI_EXP_SMI_STS):** 1- PCI Express SMI event occurred. This could be due to a PCI Express PME event or Hot Plug Event.
19:17	0h RO	Reserved.
16	0h RW/1C/V	**SMBus SMI Status (SMBUS_SMI_STS):** 0 = This bit is set from the 64 KHz clock domain used by the SMBus. Software must wait at least 15.63 microseconds after initial assertion of this bit before clearing it. This bit is sticky and is cleared by writing a 1 to this bit position. 1 = Indicates that the SMI# was caused by: 1.The SMBus Slave receiving a message that an SMI# should be caused, or 2. The SMBALERT# signal goes active and the SMB_SMI_EN bit is set and the SMBALERT_DIS bit is cleared, or 3. The SMBus Slave receiving a HOST_NOTIFY message and the HOST_NOTIFY_INTREN and the SMB_SMI_EN bits are set, or 4. The SMBus Slave receiving a "SMI in S0" message. This bit is reset by PLTRST# assertion.
15	0h RO/V	**SERIRQ_SMI Status (SERIRQ_SMI_STS):** 1 = Indicates the SMI# was caused by the SERIRQ decoder. 0 = SMI# not caused by SERIRQ decoder. ***Note:*** This bit is not sticky. Writes to this bit will have no effect.
14	0h RW/1C/V	**Periodic Status (PERIODIC_STS):** This bit will be set at the rate determined by the PER_SMI_SEL bits. If the PERIODIC_EN bit is also set, the Intel PCH will generate an SMI#. This bit is cleared by writing a 1 to this bit position. This bit is reset by PLTRST# assertion.
13	0h RW/1C/V	**TCO Status (TCO_STS):** 0 = SMI not caused by TCO logic. 1 = Indicates SMI was caused by the TCO logic. ***Note:*** Will not cause wake event. This bit is cleared by writing a 1 to this bit position. This bit is reset by PLTRST# assertion.
12	0h RO/V	**DEVMON Status (DEVMON_STS):** This read-only bit is set when bit 0 in the DEVTRAP_STS register is set. It is not sticky, so writes to this bit will have no effect.
11	0h RW/1C/V	**Microcontroller SMI Status (MCSMI_STS):** This bit is set if there is an access to the power management microcontroller range (62h or 66h). If this bit is set, and the MCSMI_EN bit is also set, the Intel PCH will generate an SMI#. This bit is set by hardware and cleared by software writing a 1 to its bit position. This bit is reset by PLTRST# assertion.
10	0h RO/V	**GPIO SMI Status (GPIO_SMI_STS):** This bit will be a 1 if any GPIO that is enabled to trigger SMI is asserted. GPIOs that are not routed to cause an SMI will have no effect on this bit. This bit is NOT sticky. Writes to this bit will have no effect. ***Note:*** See the GPIO chapter for the individual GPIO SMI status, enable, and routing bit definitions.
9	0h RO/V	**GPE0 Status (GPE0_STS):** There are several status/enable bit pairs in GPE0_STS/EN_127_96 that are capable of triggering SMI#s. This bit is a logical OR of all of those pairs (i.e. this bit is asserted whenever at least one of those pairs has both the status and enable bit asserted). This bit is NOT sticky. Writes to this bit will have no effect. ***Note:*** The setting of this bit does not cause the SMI#. The following bit pairs are included in this logical OR: - GPE0_STS/EN_127_96 [18, 17, 16, 13, 11, 10, 8, 2]
8	0h RO/V	**PM1 Status Register (PM1_STS_REG):** This is an OR of the bits (except for bits 5 and 4) in the ACPI PM1 Status Reg. Not sticky. Writes to this bit have no effect. ***Note:*** The setting of this bit does not cause the SMI#.
7	0h RO	Reserved.
6	0h RW/1C/V	**Software SMI Timer Status (SWSMI_TMR_STS):** This bit will be set to 1 by the hardware when the Software SMI# Timer expires. This bit will remain 1 until the software writes a 1 to this bit. This bit is reset by PLTRST# assertion.
5	0h RW/1C/V	**APM Status (APM_STS):** SMI# was generated by a write access to the APM control register and if the APMC_EN bit is set. This bit is cleared by writing a one to its bit position. This bit is reset by PLTRST# assertion.

Bit Range	Default & Access	Field Name (ID): Description
4	0h RW/1C/V	**SMI_ON_SLP_EN Status (SMI_ON_SLP_EN_STS):** This bit will be set by the Intel PCH when a write access attempts to set the SLP_EN bit. This bit is cleared by writing a 1 to this bit position. This bit is reset by PLTRST# assertion.
3	0h RO/V	**Legacy USB Status (LEGACY_USB_STS):** This non-sticky read-only bit is a logical OR of each of the SMI status bits in the USB Legacy Keybd Register ANDed with the corresponding enable bits. This bit will not be active if the enable bits are not set. Writes to this bit will have no effect.
2	0h RW/1C/V	**BIOS Status (BIOS_STS):** This bit gets set by hardware when a 1 is written by software to the GBL_RLS bit. When both BIOS_EN and the BIOS_STS bit are set, an SMI# will be generated. The BIOS_STS bit is cleared when software writes a 1 to this bit position. This bit is reset by PLTRST# assertion.
1:0	0h RO	Reserved

9.9.6 General Purpose Event Control (GPE_CTRL)—Offset 40h

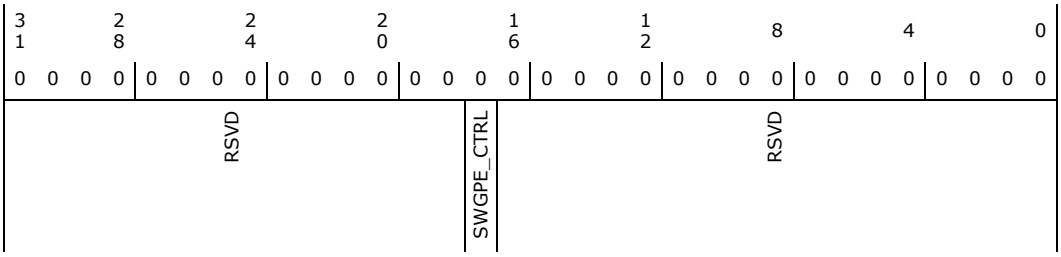
Lockable: No Usage: ACPI or Legacy Power Well:

Access Method

Type: IO Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:18	0h RO	Reserved.
17	0h RW/V	**Software GPE Control (SWGPE_CTRL):** This bit allows software to control the assertion of SWGPE_STS bit. This bit is used by hardware as the level input signal for the SWGPE_STS bit in the GPE0_STS register. When SWGPE_CTRL is 1, SWGPE_STS will be set to 1, and writes to SWGPE_STS with a value of 1 to clear SWGPE_STS will result in SWGPE_STS being set back to 1 by hardware. When SWGPE_CTRL is 0, writes to SWGPE_STS with a value of 1 will clear SWGPE_STS to 0. This bit is reset by RSMRST# assertion.
16:0	0h RO	Reserved.

9.9.7 Device Activity Status Register (DEVACT_STS)—Offset 44h

Each bit indicates if an access has occurred to the corresponding device's trap range, or for bits [6:9], if the corresponding PCI interrupt is active. This register is used in conjunction with the Periodic SMI# timer to detect any system activity for legacy power management. The periodic SMI# timer indicates if it is the right time to read the DEVACT_STS register.

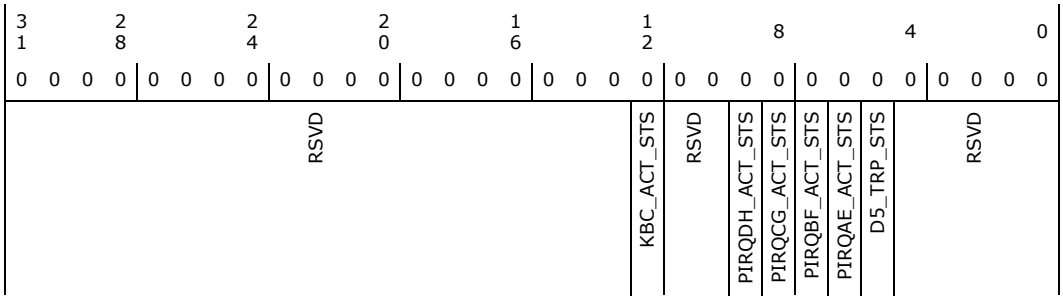
Note: Software clears bits that are set in this register by writing a 1 to the bit position.

Access Method

Type: IO Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:13	0h RO	Reserved.
12	0h RW/1C/V	**Keyboard Controller Device Activity (KBC_ACT_STS):** KBC (60/64h). 0 = Indicates that there has been no access to this device I/O range. 1 = This device I/O range has been accessed. Clear this bit by writing a 1 to the bit location. This bit is reset by PLTRST# assertion.
11:10	0h RO	Reserved
9	0h RW/1C/V	**PIRQDH Activity Status (PIRQDH_ACT_STS):** PIRQ[D or H] 0 = The corresponding PCI interrupts have not been active. 1 = At least one of the corresponding PCI interrupts have been active. Clear this bit by writing a 1 to the bit location. This bit is cleared by PLTRST# assertion.
8	0h RW/1C/V	**PIRQCG Activity Status (PIRQCG_ACT_STS):** PIRQ[C or G] 0 = The corresponding PCI interrupts have not been active. 1 = At least one of the corresponding PCI interrupts have been active. Clear this bit by writing a 1 to the bit location. This bit is cleared by PLTRST# assertion.
7	0h RW/1C/V	**PIRQBF Activity Status (PIRQBF_ACT_STS):** PIRQ[B or F] 0 = The corresponding PCI interrupts have not been active. 1 = At least one of the corresponding PCI interrupts have been active. Clear this bit by writing a 1 to the bit location. This bit is cleared by PLTRST# assertion.

Bit Range	Default & Access	Field Name (ID): Description
6	0h RW/1C/V	**PIRQAE Activity Status (PIRQAE_ACT_STS):** PIRQ[A or E] 0 = The corresponding PCI interrupts have not been active. 1 = At least one of the corresponding PCI interrupts have been active. Clear this bit by writing a 1 to the bit location. This bit is cleared by PLTRST# assertion.
5	0h RW/1C/V	**D5 Trap Status (D5_TRP_STS):** 0 = The corresponding I/O have not been accessed. 1 = The following are accessed (as determined by the I/O ranges in the LPC decoder and even if the LPC forwarding is not enabled): SP1, SP2, PP, FDC. Clear this bit by writing a 1 to the bit location. This bit is cleared by PLTRST# assertion.
4:0	0h RO	Reserved

9.9.8 PM2a Control Block (PM2A_CNT_BLK)—Offset 50h

Lockable: No Usage: ACPI or Legacy Power Well: Primary

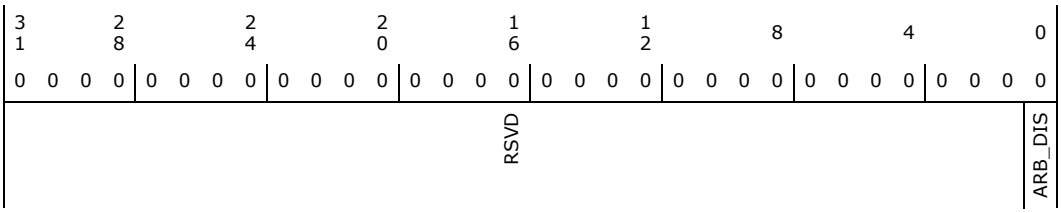
Note: The BIOS must describe this register as 1 byte wide to the OS.

Access Method

Type: IO Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:1	0h RO	Reserved
0	0h RW	**Arbiter Disable (ARB_DIS):** This bit is a scratchpad bit for legacy software compatibility. This bit is reset by PLTRST# assertion.

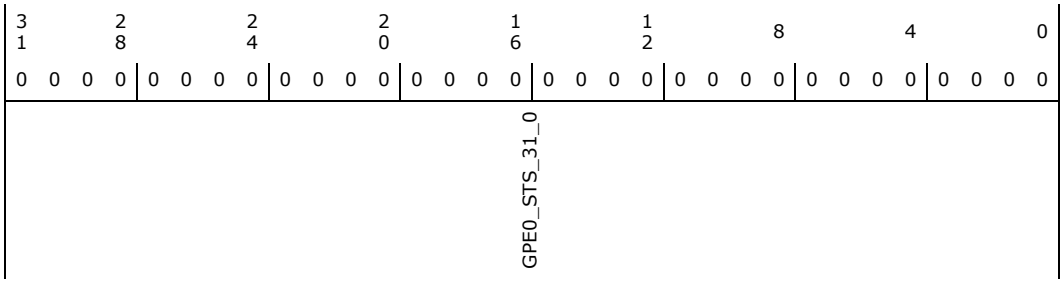
9.9.9 General Purpose Event 0 Status [31:0] (GPE0_STS_31_0) - Offset 80h

Access Method

Type: IO Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0h RW/1C/V	General Purpose Event 0 Status [31:0] (GPE0_STS_31_0): These bits are set any time the corresponding GPIO is setup as an input and the corresponding GPIO signal is high (or low if the corresponding RXINV bit is set). If the corresponding enable bit is set in the GPE0_EN_31_0 register, then when the GPE0_STS_31_0 is set: • If the system is in an S3-S5 state, the event will also wake up the system. • If the system is in the S0 state (or upon waking back to S0) an SCI will be generated, depending on the GPIROURTSCI bit for the corresponding GPI. These bits are sticky bit and are cleared by writing a "1" back to the bit position. ***Note:*** The GPP/GPD group mapped to this register is configured via GPIO_CFG.DW0 and MISCCFG.DW0 Both GPIO_CFG.DW0 and MISCCFG.DW0 must be programmed to the same state. The GPIO_CFG register is in Chapter 9, "Power Management," and the MISCCFG is in Chapter 18, "General Purpose Input and Output."

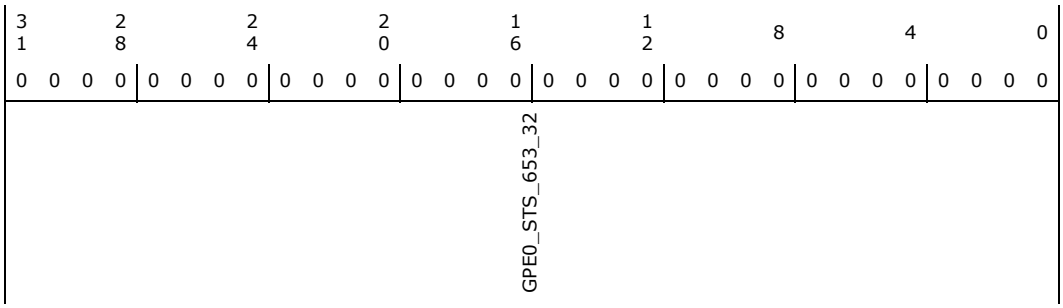
9.9.10 General Purpose Event 0 Status [63:32] (GPE0_STS_63_32) - Offset 84h

Access Method

Type: IO Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0h RW/1C/V	General Purpose Event 0 Status [63:32] (GPE0_STS_63_32): These bits are set any time the corresponding GPIO is setup as an input and the corresponding GPIO signal is high (or low if the corresponding RXINV bit is set). If the corresponding enable bit is set in the GPE0_EN_31_0 register, then when the GPE0_STS_31_0 is set: • If the system is in an S3-S5 state, the event will also wake up the system. • If the system is in the S0 state (or upon waking back to S0) an SCI will be generated, depending on the GPIROURTSCI bit for the corresponding GPI. These bits are sticky bit and are cleared by writing a "1" back to the bit position. ***Note:*** The GPP/GPD group mapped to this register is configured via GPIO_CFG.DW0 and MISCCFG.DW0 Both GPIO_CFG.DW0 and MISCCFG.DW0 must be programmed to the same state. The GPIO_CFG register is in Chapter 9, "Power Management," and the MISCCFG is in Chapter 18, "General Purpose Input and Output."

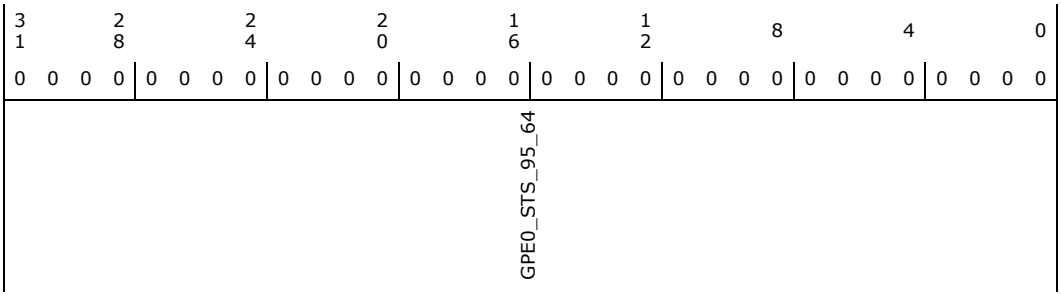
9.9.11 General Purpose Event 0 Status [95:64] (GPE0_STS_95_64) - Offset 88h

Access Method

Type: IO Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0h RW/1C/V	General Purpose Event 0 Status [95:64] (GPE0_STS_95_64): These bits are set any time the corresponding GPIO is setup as an input and the corresponding GPIO signal is high (or low if the corresponding RXINV bit is set). If the corresponding enable bit is set in the GPE0_EN_31_0 register, then when the GPE0_STS_31_0 is set: • If the system is in an S3-S5 state, the event will also wake up the system. • If the system is in the S0 state (or upon waking back to S0) an SCI will be generated, depending on the GPIROURTSCI bit for the corresponding GPI. These bits are sticky bit and are cleared by writing a "1" back to the bit position. ***Note:*** The GPP/GPD group mapped to this register is configured via GPIO_CFG.DW0 and MISCCFG.DW0 Both GPIO_CFG.DW0 and MISCCFG.DW0 must be programmed to the same state. The GPIO_CFG register is in Chapter 9, "Power Management," and the MISCCFG is in Chapter 18, "General Purpose Input and Output."

9.9.12 General Purpose Event 0 Status [127:96] (GPE0_STS_127_96)—Offset 8Ch

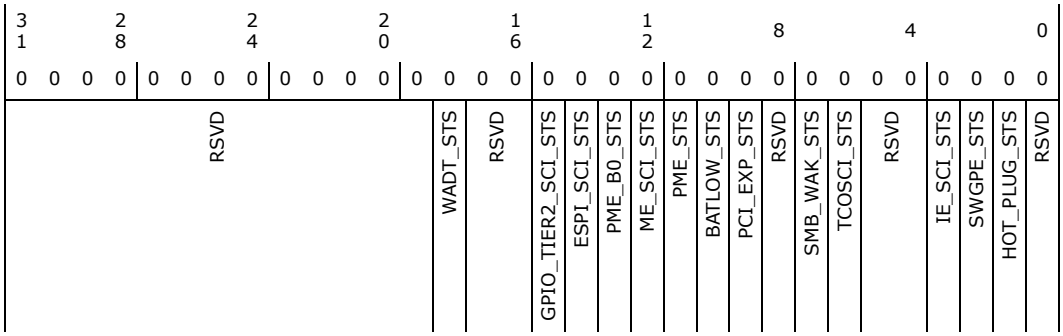
Note: This register is symmetrical to the General Purpose Event 0 Enable [127:96] Register. Unless indicated otherwise below, if the corresponding _EN bit is set, then when the STS bit get set, the Intel PCH will generate a Wake Event. Once back in an S0 state (or if already in an S0 state when the event occurs), the Intel PCH will also generate an SCI if the SCI_EN bit is set, or an SMI# if the SCI_EN bit is not set and GBL_SMI_EN is set.
The GPE0_STS bits [95:0] are claimed by the GPIO register block.

Access Method

Type: IO Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:19	0h RO	Reserved
18	0h RW/1C/V	**Wake Alarm Device Timer Status (WADT_STS):** This bit is set whenever the any of the wake alarm device timers signal a timer expiration. This bit is reset by RSMRST# assertion.
17:16	0h RO	Reserved
15	0h RW/1C/V	**GPIO Tier2 SCI Status (GPIO_TIER2_SCI_STS):** This bit is a logical OR of sci_wake from tier 2 GPIO's.
14	0h RW/1C/V	**eSPI SCI Status (ESPI_SCI_STS):** This bit will be set when an agent attached to eSPI is requesting an SCI. ***Note:*** This source is not able to cause a wake event. This bit is reset by RSMRST# assertion.

Bit Range	Default & Access	Field Name (ID): Description
13	0h RW/1C/V	**Power Management Event Bus 0 Status (PME_B0_STS):** This bit will be set to 1 by the Intel PCH when any internal device with PCI Power Management capabilities on bus 0 asserts the equivalent of the PME# signal. Additionally, if the PME_B0_EN and SCI_EN bits are set, and the system is in an S0 state, then the setting of the PME_B0_STS bit will generate an SCI (or SMI# if SCI_EN is not set). If the PME_B0_EN bit is set, and the system is in an S3-S4 state (or S5 state due to SLP_TYP and SLP_EN), then the setting of the PME_B0_STS bit will generate a wake event. If the system is in an S5 state due to power button override, then the PME_B0_STS bit will not cause a wake event or SCI. This bit is cleared by a software write of '1'. Internal devices which can set this bit: - Integrated LAN (Legacy 1 GbE, not the 10/1 GbE LAN - HD Audio/Audio DSP - SATA - EHCI ('USB2') - XHCI ('USB3') - ME Maskable Host Wake This bit is reset by RSMRST# assertion.
12	0h RW/1C/V	**ME SCI Status (ME_SCI_STS):** This bit will be set when Intel ME is requesting an SCI. Software must clear the ME source of the SCI before clearing this bit. ***Note:*** This source is not able to cause a wake event. This bit is reset by RSMRST# assertion.
11	0h RW/1C/V	**Power Management Event Status (PME_STS):** This bit will be set to 1 by hardware when the PME# signal goes active. Additionally, if the PME_EN and SCI_EN bits are set, and system is in an S0 state, then the setting of the PME_STS bit will generate an SCI (or SMI# if SCI_EN is not set). If the PME_EN bit is set, and the system is in an S3-S4 state (or S5 state due to SLP_TYP and SLP_EN), then the setting of the PME_STS bit will generate a wake event. If the system is in an S5 state due to power button override, then PME_STS will not cause a wake event or SCI. This bit is cleared by writing a 1 to this bit position or RSMRST# assertion.
10	0h RW/1C/V	**RSVD**
9	0h RW/1C/V	**PCI Express Status (PCI_EXP_STS):** This bit will be set to 1 by hardware to indicate that: - The PME event message was received on one or more of the PCI-Express Ports - An Assert PMEGPE message received from the MCH via DMI. ***Note:*** The PCI WAKE# pin and the PCI Express Beacons have no impact on this bit. Software attempts to clear this bit by writing a 1 to this bit position. If the PCI_EXP_STS bit went active due to an Assert PMEGPE message, then a Deassert PMEGPE message must be received prior to the software write in order for the bit to be cleared. If the bit is not cleared and the corresponding PCI_EXP_EN bit is set, the leveltriggered SCI will remain active. Note that a race condition exists where the PCI Express device sends another PME message because the PCI Express device was not serviced within the time when it must resend the message. This may result in a spurious interrupt, and this is comprehended and approved by the PCI Express specification. The window for this race condition is approximately 95-105 milliseconds. This bit is reset by RSMRST# assertion.
8	0h RW/1C/V	**RSVD**
7	0h RW/1C/V	**SMBus Wake Status (SMB_WAK_STS):** This bit is set to 1 by the hardware to indicate that the wake event was caused by the PCH's SMBus logic. This could be due to either the SM Bus slave unit receiving a message or the SMBALERT# signal going active. ***Notes:*** 1. The SMBus controller will independently cause an SMI# so this bit does not need to do so (unlike the other bits in this register). 2. This bit is set by the SMBus slave command 01h (Wake/SMI#) even when the system is in the S0 state. Therefore, to avoid an instant wake on subsequent transitions to sleep states, software must clear this bit after each reception of the Wake/SMI# command or just prior to entering the sleep state. 3. The SMBALERT_STS bit (D31:F3:I/O Offset 00h:bit 5) should be cleared by software before clearing this bit. This bit is reset by RSMRST# assertion.
6	0h RW/1C/V	**TCOSCI Status (TCOSCI_STS):** This bit will be set to 1 by hardware when the TCO logic or Thermal Sensor logic causes an SCI. This bit can be reset by writing a one to this bit position or by RSMRST# assertion.
5:4	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
3	RW/1C/V	**IE SCI Status (IE_SCI_STS):** This bit will be set when IE is requesting an SCI. Software must clear the IE source of the SCI before clearing this bit. This bit will not cause a wake event.
2	0h RW/1C/V	**Software GPE Status (SWGPE_STS):** The SWGPE_CTRL bit (bit 1 of GPE_CTRL reg) acts as a level input to this bit. This bit is reset by RSMRST# assertion.
1	0h RW/1C/V	**Hot Plug Status (HOT_PLUG_STS):** Enables PCH to cause an SCI when the HOT_PLUG_STS bit is set. This is used to allow the PCI Express ports to cause an SCI due to hot-plug events. The following events cause HOT_PLUG_STS bit to set - Assert GPE message received from any of the PCIE ports in PCH - Assert HPGPE message received from any of the PCIE ports in PCH - Assert GPE message received downstream from MCH - Assert HPGPE message received downstream from MCH. This bit is reset by RSMRST# assertion.
0	0h RO	Reserved

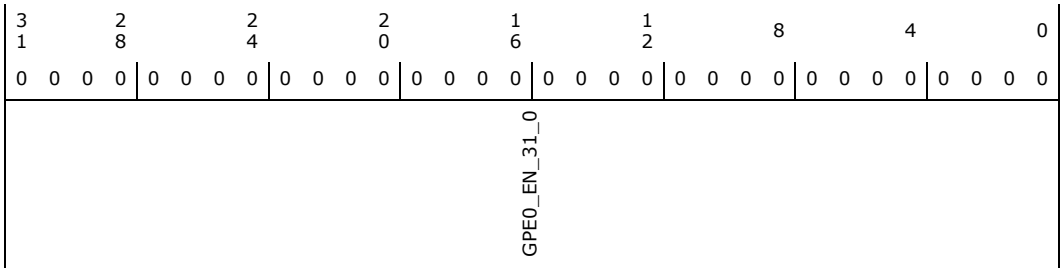
9.9.13 General Purpose Event 0 Enable[31:0] (GPE0_EN_31_0) - Offset 90h

Access Method

Type: IO Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0h RW	**General Purpose Event 0 Enable [31:0] (GPE0_EN_31_0)**: These bits enable the corresponding GPE0_STS[31:0] bits being set to cause an SCI and/or wake event. ***Note:*** The GPP/GPD group mapped to this register is configured via GPIO_CFG.DW2 and MISCCFG.DW2 Both GPIO_CFG.DW2 and MISCCFG.DW2 must be programmed to the same state. The GPIO_CFG register is in Chapter 9, "Power Management," and the MISCCFG is in Chapter 18, "General Purpose Input and Output."

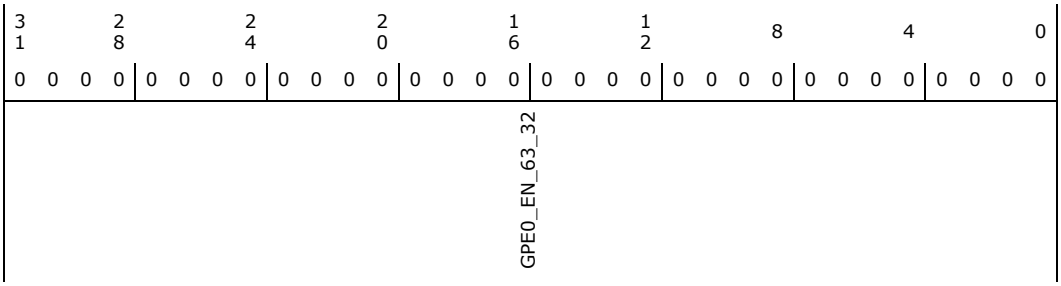
9.9.14 General Purpose Event 0 Enable[63_32] (GPE0_EN_63_32) - Offset 94h

Access Method

Type: IO Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0h RW	**General Purpose Event 0 Enable [63:32] (GPE0_EN_63_32)**: These bits enable the corresponding GPE0_STS[31:0] bits being set to cause an SCI and/or wake event. ***Note:*** The GPP/GPD group mapped to this register is configured via GPIO_CFG.DW2 and MISCCFG.DW2 Both GPIO_CFG.DW2 and MISCCFG.DW2 must be programmed to the same state. The GPIO_CFG register is in Chapter 9, "Power Management," and the MISCCFG is in Chapter 18, "General Purpose Input and Output."

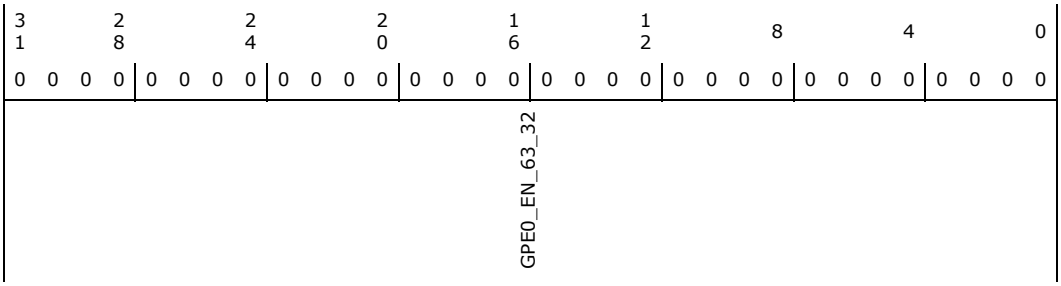
9.9.15 General Purpose Event 0 Enable[95_64] (GPE0_EN_95_64) - Offset 98h

Access Method

Type: IO Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0h RW	**General Purpose Event 0 Enable [95:64] (GPE0_EN_95_64)**: These bits enable the corresponding GPE0_STS[31:0] bits being set to cause an SCI and/or wake event. ***Note:*** The GPP/GPD group mapped to this register is configured via GPIO_CFG.DW2 and MISCCFG.DW2 Both GPIO_CFG.DW2 and MISCCFG.DW2 must be programmed to the same state. The GPIO_CFG register is in Chapter 9, "Power Management," and the MISCCFG is in Chapter 18, "General Purpose Input and Output."

9.9.16 General Purpose Event 0 Enable [127:96] (GPE0_EN_127_96)—Offset 9Ch

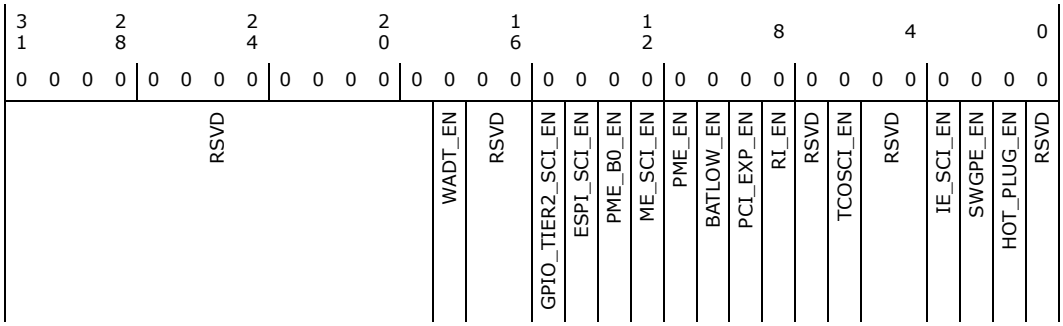
Note: This register is symmetrical to the General Purpose Event 0 Status [127:96] Register. The GPE0_STS bits [95:0] are claimed by the GPIO register block.

Access Method

Type: IO Register
(Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:19	0h RO	Reserved
18	0h RW	**Wake Alarm Device Timer Enable (WADT_EN):** Used to enable the setting of the WADT_STS bit to generate Wake/SMI#/SCI. This bit is reset by DSW_PWROK de-assertion.
17:16	0h RO	Reserved
15	0h RW/V	**GPIO Tier2 SCI EN (GPIO_TIER2_SCI_EN):** Used to enable the setting of GPIO_TIER2_SCI_STS to generate wake/SCI#.
14	0h RW/V	**eSPI SCI Enable (ESPI_SCI_EN):** Used to enable the setting of the ESPI_SCI_STS bit to generate a SCI.
13	0h RW/V	**PME_B0 Enable (PME_B0_EN):** Enables the setting of the PME_B0_STS bit to generate a wake event and/or an SCI or SMI#. This bit is reset by RTCRST# assertion.
12	0h RW/V	**ME SCI Enable (ME_SCI_EN):** Used to enable the setting of the ME_SCI_STS bit to generate a SCI.

Bit Range	Default & Access	Field Name (ID): Description
11	0h RW/V	**Power Management Event Enable (PME_EN):** Enables the setting of the PME_STS to generate a wake event and/or an SCI. This bit is reset by RTCRST# assertion.
10	0h RW/V	**Reserved**
9	0h RW/V	**PCI Express Enable (PCI_EXP_EN):** Enables PCH to cause an SCI when PCI_EXP_STS bit is set. This is used to allow the PCI Express ports, including the link to the MCH, to cause an SCI due to wake/PME events.
8	0h RW/V	**RI Enable (RI_EN):** When RI_EN and RI_STS are both set, a Wake event will occur. If RI_EN is not set, then when RI_STS is set, no Wake event will occur. This bit is reset by RTCRST# assertion.
7	0h RO	Reserved
6	0h RW/V	**TCOSCI Enable (TCOSCI_EN):** When TCOSCI_EN and TCOSCI_STS are both set, an SCI will be generated. This bit is reset by RSMRST# assertion.
5:4	0h RO	Reserved
3	0b RW/V	**IE SCI Enable (IE_SCI_EN)**. When set to 1b, enables IE_SCI_STS bit to generate an SCI.
2	0h RW/V	**Software GPE Enable (SWGPE_EN):** This bit, when set to 1, enables the SW GPE function. If SWGPE_CTRL is written to a 1, hardware will set SWGPE_STS (acts as a level input) If SWGPE_STS, SWGPE_EN, and SCI_EN are all 1's, an SCI will be generated If SWGPE_STS = 1, SWGPE_EN = 1, SCI_EN = 0, and GBL_SMI_EN = 1 then an SMI# will be generated
1	0h RW/V	**Hot Plug Enable (HOT_PLUG_EN):** Enables PCH to cause an SCI when the HOT_PLUG_STS bit is set. This is used to allow the PCI Express ports to cause an SCI due to hot-plug events. The following events cause HOT_PLUG_STS bit to set: - Assert GPE message received from any of the PCIE ports in PCH - Assert HPGPE message received from any of the PCIE ports in PCH - Assert GPE message received downstream from MCH - Assert HPGPE message received downstream from MCH
0	0h RO	Reserved

9.10 PMC Memory Mapped Registers Summary

Table 9-19. Summary of PMC Memory Mapped Registers (Sheet 1 of 3)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	Wake Alarm Device Timer: AC (WADT_AC)—Offset 0h	FFFFFFFFh
4h	7h	Wake Alarm Device Timer: DC (WADT_DC)—Offset 4h	FFFFFFFFh
8h	Bh	Wake Alarm Device Expired Timer: AC (WADT_EXP_AC)—Offset 8h	FFFFFFFFh
Ch	Fh	Wake Alarm Device Expired Timer: DC (WADT_EXP_DC)—Offset Ch	FFFFFFFFh
10h	13h	Power and Reset Status (PRSTS)—Offset 10h	0h
14h	17h	Chipset Initialization Register 14 (CIR14)—Offset 14h	0h
18h	1Bh	Power Management Configuration Reg 1 (PM_CFG)—Offset 18h	20h
1Ch	1Fh	PCH Power Management Status (PCH_PM_STS) - Offset 1Ch	0h
20h	23h	Chipset Initialization Register 20(CIR20)—Offset 20	0h
24h	27h	PCH Power Management Status (PCH_PM_STS2)—Offset 24h	0h
28h	2Bh	S3 Power Gating Policies (S3_PWRGATE_POL)—Offset 28h	0h
2Ch	2Fh	S4 Power Gating Policies (S4_PWRGATE_POL)—Offset 2Ch	0h

Table 9-19. Summary of PMC Memory Mapped Registers (Sheet 2 of 3)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
30h	33h	S5 Power Gating Policies (S5_PWRGATE_POL)—Offset 30h	0h
34h	37h	DeepSx Configuration (DSX_CFG)—Offset 34h	0h
3Ch	3Fh	Power Management Configuration Reg 2 (PM_CFG2)—Offset 3Ch	0h
40h	43h	Chipset Initialization Register 40 (CIR40)—Offset 40h	0h
44h	47h	Chipset Initialization Register 44 (CIR44)—Offset 44h	0h
48h	4Bh	Chipset Initialization Register 48 (CIR48)—Offset 48h	0h
4Ch	4Fh	Chipset Initialization Register 4C (CIR4C)—Offset 4Ch	0h
50h	53h	Chipset Initialization Register 50 (CIR50)—Offset 50h	0h
54h	57h	Chipset Initialization Register 54 (CIR54)—Offset 54h	0h
58h	5Bh	Chipset Initialization Register 58 (CIR58)—Offset 58h	0h
60h	63h	Chipset Initialization Register 60 (CIR60)—Offset 60h	0h
68h	6Bh	Chipset Initialization Register 68 (CIR68)—Offset 68h	0h
78h	7Bh	Chipset Initialization Register 78 (CIR78)—Offset 78h	0h
7Ch	7Fh	Chipset Initialization Register 7C (CIR7C)—Offset 7Ch	0h
80h	83h	Chipset Initialization Register 80 (CIR80)—Offset 80h	0h
84h	87h	Chipset Initialization Register 84 (CIR84)—Offset 84h	0h
88h	8Bh	Chipset Initialization Register 88 (CIR88)—Offset 88h	0h
8Ch	8Fh	Chipset Initialization Register 8C (CIR8C)—Offset 8Ch	0h
90h	93h	Chipset Initialization Register 90 (CIR90)—Offset 90h	0h
98h	9Bh	Chipset Initialization Register 98 (CIR98)—Offset 98h	0h
A0h	A3h	Chipset Initialization Register A0 (CIRA0)—Offset A0h	0h
A4h	A7h	Chipset Initialization Register A4 (CIRA4)—Offset A4h	0h
A8h	ABh	Chipset Initialization Register A8 (CIRA8)—Offset A8h	0h
ACh	AFh	Chipset Initialization Register AC (CIRAC)—Offset ACh	0h
B0h	B3h	Chipset Initialization Register B0 (CIRB0)—Offset B0h	0h
B4h	B7h	Chipset Initialization Register B4 (CIRB4)—Offset B4h	0h
C0h	C3h	Chipset Initialization Register C0 (CIRC0)—Offset C0h	0h
C4h	C7h	Chipset Initialization Register C4 (CIRC4)—Offset C4h	0h
C8h	CBh	PM_SYNC Miscellaneous Config (PM_SYNC_MISC_CFG)—Offset C8h	0h
D0h	D3h	Chipset Initialization Register D0 (CIRD0)—Offset D0h	0h
D4h	D7h	Chipset Initialization Register D4 (CIRD4)—Offset D4h	0h
DCh	DFh	Chipset Initialization Register DC (CIRDC)—Offset DCh	0h
E0h	E3h	Chipset Initialization Register E0 (CIRE0)—Offset E0h	0h
E4h	E7h	Chipset Initialization Register E4 (CIRE4)—Offset E4h	0h
E8h	EBh	Chipset Initialization Register E8 (CIRE8)—Offset E8h	0h
F0h	F3h	ADR Enable (ADR_EN) - Offset F0h	
FCh	FFh	ACPI Timer Control (ACPI_TMR_CTL) - Offset FCH	0h
10C	10fh	Global to Host reset enable (GBL2HOST_EN) - Offset 10Ch	
110h	113h	Last TSC Alarm Value[31:0] (TSC_ALARM_LO)—Offset 110h	0h
114h	117h	Last TSC Alarm Value[63:32] (TSC_ALARM_HI)—Offset 114h	0h
120h	123h	GPIO Configuration (GPIO_CFG)—Offset 120h	432h

Table 9-19. Summary of PMC Memory Mapped Registers (Sheet 3 of 3)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
124h	127h	Global Reset Causes Register 0 (GBLRST_CAUSE0) - Offset 124h	0h
128h	12Bh	Global Reset Causes Register 1 (GBLRST_CAUSE1) - Offset 128h	0h
12Ch	12Fh	Host Partition Reset Causes (HPR_CAUSE0) - Offset 12Ch	0h
200h	203h	(MODPHY_PM_CFG1)—Offset 200h	FFFFh
204h	207h	MODPHY Power Management Configuration 2 (MODPHY_PM_CFG2)—Offset 204h	5000000h
208h	20Bh	MODPHY Power Management Configuration 3 (MODPHY_PM_CFG3)—Offset 208h	0h
20Ch	20Fh	MODPHY Power Management Configuration 4 (MODPHY_PM_CFG4)—Offset 20Ch	0h
324h	327h	Chipset Initialization Register 324 (CIR324)—Offset 324h	0h
328h	32Bh	Chipset Initialization Register 328 (CIR328)—Offset 328h	0h
32Ch	32Fh	Chipset Initialization Register 32C (CIR32C)—Offset 32Ch	0h
5D0h	5D3h	Host SW PG Control Register 1 (HSWPGCR1) —Offset 5D0h	0h

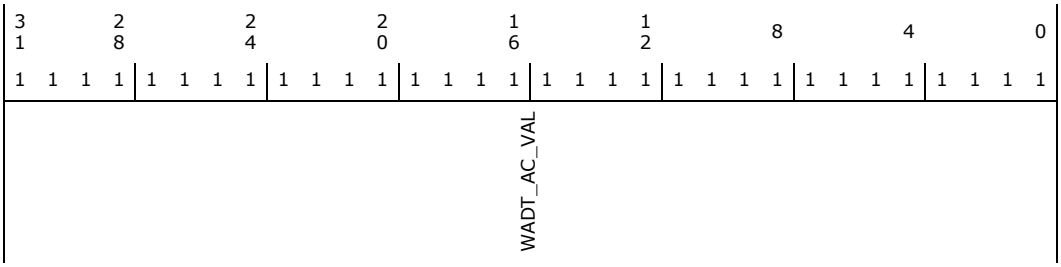
9.10.1 Wake Alarm Device Timer: AC (WADT_AC)—Offset 0h

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31
Function: 2

Default: FFFFFFFFh



Bit Range	Default & Access	Field Name (ID): Description
31:0	FFFFFFFFh RW/V	**Wake Alarm Device Timer Value for AC Mode (WADT_AC_VAL):** This field contains the 32-bit wake alarm device timer value (granularity 1s) for AC power. The timer begins decrementing when written to a value other than FFFFFFFFh (regardless of the power source when the write occurs). Upon counting down to 0: If on AC power, GPE0b_STS.WADT_STS will be set. This status bit being set will generate a host wake if GPE0b_EN.WADT_EN is 1. If the power source is DC at this time, the status bit is not set. However, if AC power subsequently returns to the platform, the AC Expired Timer begins running. See the WADT_EXP_AC register for details. The timer returns to its default value of FFFFFFFFh. This bit is reset by DSW_PWROK de-assertion.

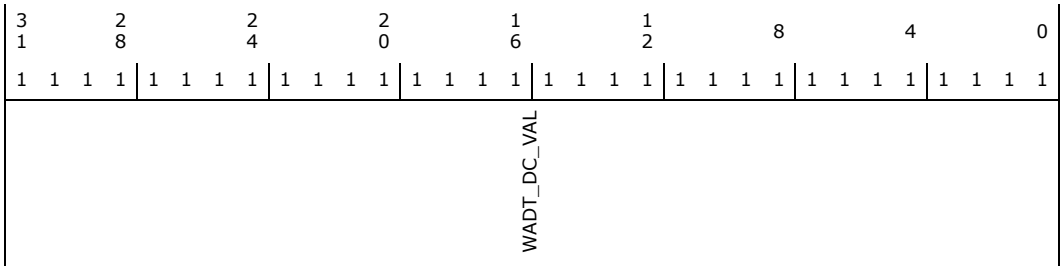
9.10.2 Wake Alarm Device Timer: DC (WADT_DC)—Offset 4h

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31
Function: 2

Default: FFFFFFFFh



Bit Range	Default & Access	Field Name (ID): Description
31:0	FFFFFFFFh RW/V	**Wake Alarm Device Timer Value for DC Mode (WADT_DC_VAL):** This field contains the 32-bit wake alarm device timer value (granularity 1s) for DC power. The timer begins decrementing when written to a value other than FFFFFFFFh (regardless of the power source when the write occurs). Upon counting down to 0: If on DC power, GPE0b_STS.WADT_STS will be set. This status bit being set will generate a host wake if GPE0b_EN.WADT_EN is 1. If the power source is AC at this time, the status bit is not set. However, if DC power subsequently returns to the platform, the DC Expired Timer begins running. See the WADT_EXP_DC register for details. The timer returns to its default value of FFFFFFFFh. This bit is reset by DSW_PWROK de-assertion.

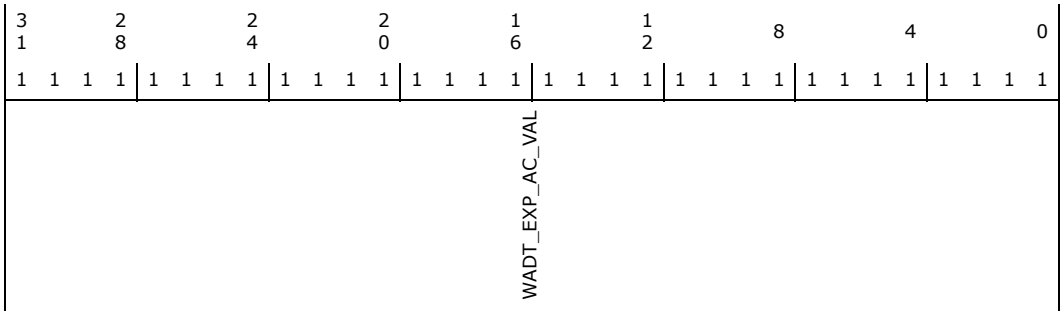
9.10.3 Wake Alarm Device Expired Timer: AC (WADT_EXP_AC)—Offset 8h

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31
Function: 2

Default: FFFFFFFFh



Bit Range	Default & Access	Field Name (ID): Description
31:0	FFFFFFFFh RW/V	**Wake Alarm Device Expired Timer Value for AC Mode (WADT_EXP_AC_VAL):** This field contains the 32-bit wake alarm device [quote]Expired Timer[quote] value (granularity 1s) for AC power. The timer begins decrementing after switching from DC to AC power, in the case where the WADT_AC timer has already expired while the platform was on DC power. This timer only decrements while operating on AC power. So if the power source switches back to DC power, the timer will stop (but not reset). When AC power returns, the timer will again begin decrementing. ***Note:*** This timer will only begin decrementing under the conditions described above if this field has been configured for something other than its default value of FFFFFFFFh (i.e. FFFFFFFFh = disabled). Upon expiration of this timer: If on AC power, GPE0b_STS.WADT_STS will be set. This status bit being set will generate a host wake if GPE0b_EN.WADT_EN is 1. BOTH the AC and DC Expired Timers return to their defaults value of FFFFFFFFh. This bit is reset by DSW_PWROK de-assertion.

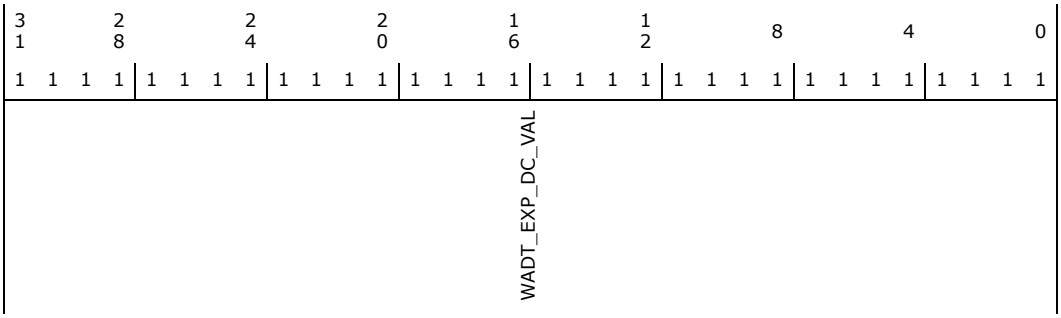
9.10.4 Wake Alarm Device Expired Timer: DC (WADT_EXP_DC)—Offset Ch

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31
Function: 2

Default: FFFFFFFFh



Bit Range	Default & Access	Field Name (ID): Description
31:0	FFFFFFFFh RW/V	**Wake Alarm Device Expired Timer Value for DC Mode (WADT_EXP_DC_VAL):** This field contains the 32-bit wake alarm device [quote]Expired Timer[quote] value (granularity 1s) for DC power. The timer begins decrementing after switching from AC to DC power, in the case where the WADT_DC timer has already expired while the platform was on AC power. This timer only decrements while operating on DC power. So if the power source switches back to AC power, the timer will stop (but not reset). When DC power returns, the timer will again begin decrementing. ***Note:*** This timer will only begin decrementing under the conditions described above if this field has been configured for something other than its default value of FFFFFFFFh (i.e. FFFFFFFFh = disabled). Upon expiration of this timer: If on DC power, GPE0b_STS.WADT_STS will be set. This status bit being set will generate a host wake if GPE0b_EN.WADT_EN is 1. BOTH the AC and DC Expired Timers return to their defaults value of FFFFFFFFh. This bit is reset by DSW_PWROK de-assertion.

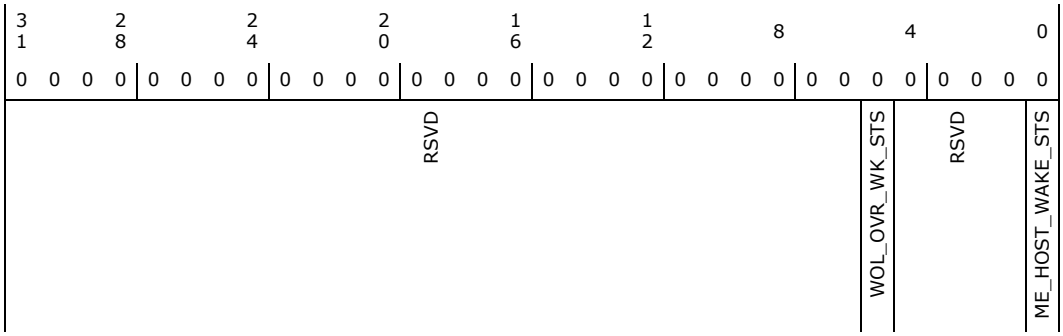
9.10.5 Power and Reset Status (PRSTS)—Offset 10h

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:6	0h RO	Reserved
5	0h RW/1C/V	**Wake On LAN Override Wake Status (WOL_OVR_WK_STS):** This bit gets set when integrated LAN Signals a Power Management Event AND the system is in S5. BIOS can read this status bit to determine this wake source. Software clears this bit by writing a 1 to it.
4:1	0h RO	Reserved
0	0h RW/1C/V	**ME_HOST_WAKE_STS (ME_HOST_WAKE_STS):** This bit is set when the ME generates a non-maskable wake event and is not affected by any other enable bit. When this bit is set, the host power management logic wakes to S0.

9.10.6 Chipset Initialization Register 14 (CIR14)—Offset 14h

BIOS may write to the register.

9.10.7 Power Management Configuration Reg 1 (PM_CFG)—Offset 18h

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31
Function: 2

Default: 20h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
RSVD				Register Lock	RSVD					RTC_DSX_WAKE_DIS	RSVD	SLP_SUS_MIN_ASST_WDTH		SLP_A_MIN_ASST_WDTH		SLP_GBE_MIN_ASST_WDTH		RSVD			PB_DB_MODE	PWR_CYC_DUR		RSVD		COCS	RSVD			TIMING T581	

Bit Range	Default & Access	Field Name (ID): Description
31:22	0h RW	Reserved
27	0b RW/L	**Register Lock**: When this bit is set, the GBL2HOST_EN register is locked down. One written set to "1", only a PLTRST# will reset the bit to o.
26:23	0h RW	**Reserved**
22	0b RW	**Reserved.** BIOS will need to program this bit to 1b. This bit is self locking. Once set to 1b it will take a system reset to clear back to 0b
21	0h RW	**RTC Wake from DeepSx Disable (RTC_DSX_WAKE_DIS):** When set, this bit disables RTC wakes from waking the system from DeepSx. This bit is reset by RTCRST# assertion.
20	0h RO	Reserved
19:18	0h RW/L	**SLP_SUS# Minimum Assertion Width (SLP_SUS_MIN_ASST_WDTH):** This 2-bit value indicates the minimum assertion width of the SLP_SUS# signal to guarantee that the SUS well power supplies have been fully power-cycled. This value may be modified per platform depending on power supply capacitance, board capacitance, power failure detection circuits, etc. Settings are: 00 = 0 ms (i.e. stretching disabled - default) 01 = 500ms 10 = 1s 11 = 4s This field is not writable when the SLP_Sx# Stretching Policy Lock- Down bit is set. This field is ignored when exiting a G3 state if the "Disable SLP_X Stretching After SUS Power Failure" bit is set. Note that unlike with all other SLP_* pin stretching, this disable bit only impacts SLP_SUS# stretching during G3 exit rather than both G3 and DeepSx exit. SLP_SUS# stretching always applies to DeepSx regardless of the disable bit. **Programming Note**: For platforms that enable DeepSx, BIOS must program SLP_SUS# stretching to be greater than or equal to the largest stretching value on any other SLP_* pin (SLP_S3#, SLP_S4#, SLP_GBE#, or SLP_A#). This bit is cleared by the RTCRST# pin.

Bit Range	Default & Access	Field Name (ID): Description
17:16	0h RW/L	**SLP_A# Minimum Assertion Width (SLP_A_MIN_ASST_WDTH):** This 2-bit value indicates the minimum assertion width of the SLP_A# signal to guarantee that the ASW power supplies have been fully power-cycled. This value may be modified per platform depending on power supply capacitance, board capacitance, power failure detection circuits, etc. Settings are: 00 = 0 ms (i.e. stretching disabled - default) 01 = 4 s 10 = 98 ms 11 = 2 s This field is not writable when the SLP_Sx# Stretching Policy Lock- Down bit is set. This field is ignored when exiting a G3 or Deep Sx state if the "Disable SLP_X Stretching After SUS Power Failure" bit is set. This bit is cleared by the RTCRST# pin.
15:14	0h RW/L	**SLP_GBE# Minimum Assertion Width (SLP_GBE_MIN_ASST_WDTH):** This 2-bit value indicates the minimum assertion width of the SLP_GBE# signal to guarantee that the power to the PHY has been fully power-cycled. This value may be modified per platform depending on power supply, capacitance, board capacitance, power failure detection circuits, etc. This field is not writable when the SLP_Sx# Stretching Policy Lock Down bit is set to 1. This bit is reset by RTCRST# assertion.
13:11	0b RW	**After G3 Last State Enable (AG3_LS_EN):** When PM_CFG.AG3E is '0', AG3_LS_EN determines whether the PCH will consider the platform's previous state when determining whether to power-up after G3. Encodings: 0 = (default): PCH power-up policies after G3 do not depend on the platform's state when the G3 occurred. 1 = PCH power-up policies after G3 depend on the platform's state when the G3 occurred. — If the power failure occurred while the platform was in S0 or while in the process of going to S0, the platform will power-up to S0 upon exiting G3. If the power failure occurred while the platform was in Sx or while in the process of going to Sx, the platform will stay in S4/S5 upon exiting G3. ***Note:*** This bit applies only when GEN_PMCON_3.AG3E is '0'. If AG3E is '1', the platform will always stay in S4/S5 after G3 regardless of the value of AG3_LS_EN. ***Note:*** The destination state is considered to be S0 during both host wake and throughout host partition resets with/without power cycle. Reset with RTCRST#
12	0b RW	**After Type 8 Global Reset Last State Enable (A8GR_LS_EN):** AGR_LS_EN determines whether the PCH will consider the platform's previous state when determining whether to power-up after non-thermal and non-explicitly requested type 8 global resets. See the "Reset Sources" table for a list of impacted reset types. Encodings: 0 = (default): PCH power-up policies after a global reset do not depend on the platform's state when the reset occurred. 1 = PCH power-up policies after a global reset depend on the platform's state when the reset occurred. — If the global reset occurred while the platform was in S0 or while in the process of going to S0, the platform will power-up to S0 after the reset. — If the global reset occurred while the platform was in Sx or while in the process of going to Sx, the platform will stay in S5 upon exiting G3. ***Note:*** he destination state is considered to be S0 during both host wake and throughout host partition resets with/without power cycle. This bit is reset by RTCRST#
11	0b RW	**Global Reset Three Strike Counter Enable (GR_TSC_EN):** When set, GR_TSC_EN will cause the PMC to keep the platform in S5 after the third consecutive type 7 global reset occurs during the boot flow. The three strike counter is reset in the following situations: • The system reaches S0. A type 8 global reset occurs, including after the three strike counter causes the system to stay in S5. • RSMRST# asserts (due to DeepSx entry or a G3 event occurring). This bit is reset by RTCRST#

Bit Range	Default & Access	Field Name (ID): Description
10	0h RW	**Power Button Debounce Mode (PB_DB_MODE):** This bit controls when interrupts (SMI#, SCI) are generated in response to assertion of the PWRBTN# pin. This bit's values cause the following behavior: - '0': The 16ms debounce period applies to all usages of the PWRBTN# pin (legacy behavior). - '1': When a falling edge occurs on the PWRBTN# pin, an interrupt is generated and the 16ms debounce timer starts. Subsequent interrupts are masked while the debounce timer is running. Power button override logic always samples the post-debounce version of the pin. This bit is reset by RTCRST# assertion.
9:8	0h RW/L	**Reset Power Cycle Duration (PWR_CYC_DUR):** The value in this register determines the minimum time a platform will stay in reset (SLP_S3#, SLP_S4#, SLP_S5# asserted and also SLP_A# and SLP_GBE# asserted if applicable) during a host partition reset with power cycle or a global reset. The duration programmed in this register takes precedence over the applicable SLP_# stretch timers in these reset scenarios. This field is not writable when the SLP_Sx# Stretching Policy Lock-Down bit is set. Note that the duration programmed in this register should never be smaller than the stretch duration programmed in the following registers: - GEN_PMCON_3.SLP_S3_MIN_ASST_WDTH - GEN_PMCON_3.S4MAW - PM_CFG.SLP_A_MIN_ASST_WDTH - PM_CFG.SLP_GBE_MIN_ASST_WDTH This bit is reset by RTCRST# assertion.
7:6	0h RO	Reserved
5	1h RW/V	**CPU OC Strap (COCS):** SW programs this pin with the value that should be reflected to the GPIO8_OCS pin, when the pin is in native mode. Hardware also sets this bit when the over-clocking watchdog timer expires. This bit is reset by RSMRST# assertion.
4:2	0h RO	Reserved
1:0	0h RW/V	**Timing T581:** This field configures the t581 timing in the Purley PDG involved in the power down flow (PROCPWRGD de-assertion to CLKOUT_BCLK turning OFF). Encodings (all min timings): 00: 10 us (default) 01: 100 us 10: 1 ms 11: 10 ms reset_type=host_deep_rst_b

9.10.8 PCH Power Management Status (PCH_PM_STS)—Offset 1Ch

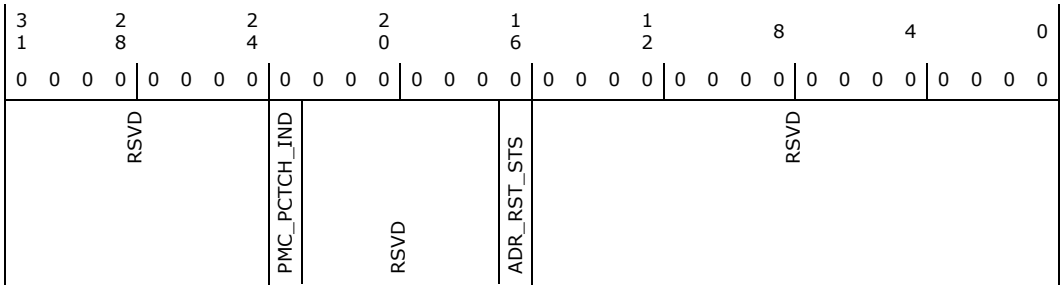
This register contains the status bit to indicate that an ADR flow occurred. All other bits are reserved.

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	**Reserved**
23	0b RO/V	**PMC Patch Load indicator (PMC_PATCH_IND)**: This bit is reset to 0b on RSMRST# going high or a global reset. The PMC sets this bit to "1" when it loads and starts using a patch. The PMC is always expected to load a PMC FW patch. BIOS should check this bit at boot and alert the user if not set to "1"
22:17	0h RO	**Reserved**
16	0b RW/1C/V	**ADR Reset Status (ADR_RST_STS):** Went set to "1", this bit indicates that the Automatic DIMM Self-Refresh (ADR) flow was executed during a prior global reset entry.
15-0	0h RO/RW/ 1C/V	**Reserved**

9.10.9 Chipset Initialization Register 20(CIR20)—Offset 20h

The BIOS may write to the register.

9.10.10 PCH Power Management Status (PCH_PM_STS2)—Offset 24h

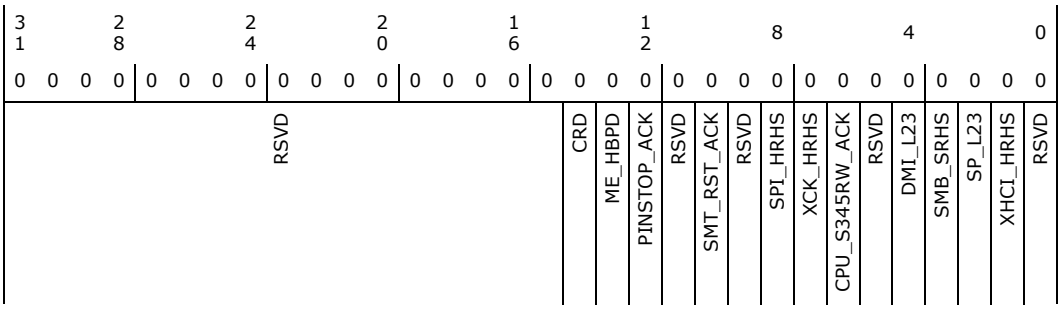
This register contains misc. fields used to record events pertaining to PCH power management. Unless otherwise indicated, all RWC bits are cleared with a write of '1' by software.

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:15	0h RO	Reserved
14	0h RW/1C/V	**CPU Reset Done Failure (CRD):** CPU Reset Done message did not arrive from the CPU.
13	0h RW/1C/V	**ME Host Boot Prep Done Failure (ME_HBPD): Intel** ME Host Boot Preparation did not complete.
12	0h RW/1C/V	**PINSTOP Acknowledge Failure (PINSTOP_ACK):** The GbE PHY did not respond to the PINSTOP message.
11	0h RO	Reserved
10	0h RW/1C/V	**SMT Reset Acknowledge Failure (SMT_RST_ACK):** One or more SMT controllers did not respond to the ME bus reset warning.
9	0h RO	Reserved
8	0h RW/1C/V	**SPI Common Prep Handshake Failure (SPI_HRHS):** The SPI controller did not complete the host partition reset/Sx entry handshake.
7	0h RW/1C/V	**XCK Common Prep Handshake Failure (XCK_HRHS):** The integrated clocking unit did not complete the host partition reset/Sx entry handshake.
6	0h RW/1C/V	**CPU S345/Reset Warn Acknowledge Failure (CPU_S345RW_ACK):** The CPU did not respond to the GO_S345 or RESET_WARN message.
5	0h RO	Reserved
4	0h RW/1C/V	**DMI L23 Entry Failure (DMI_L23):** The DMI interface did not respond to the request to entry L23.
3	0h RW/1C/V	**SMBus Host Reset Handshaking Failure (SMB_SRHS):** The host SMBus controller did not complete the host partition reset handshake.

Bit Range	Default & Access	Field Name (ID): Description
2	0h RW/1C/V	**South Port L23 Entry Failure (SP_L23):** The PCH PCI Express ports did not complete the host partition reset/Sx entry handshake.
1	0h RW/1C/V	**XHCI Common Prep Handshake Failure (XHCI_HRHS):** The XHCI controller did not complete the host partition reset/Sx entry handshake.
0	0h RO	Reserved

9.10.11 S3 Power Gating Policies (S3_PWRGATE_POL)—Offset 28h

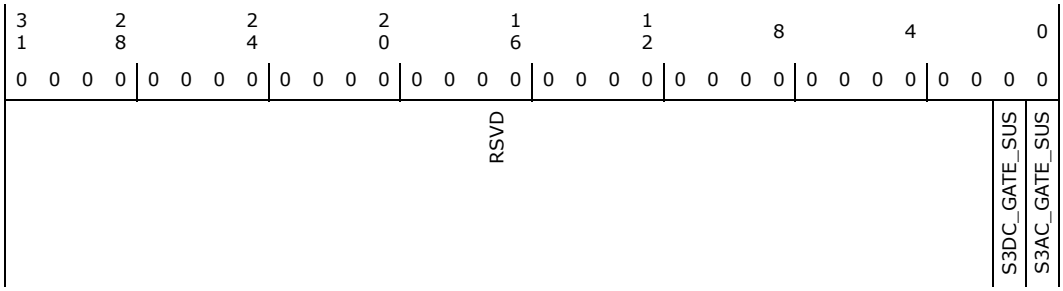
This register contains policy bits to configure various power gating options while the system is in S3. Note that setting any of these policies to "enabled" may not directly result in power gating - in some cases other HW qualifications may be dynamically applied.
This register is in the RTC power well and is reset by RTCRST# assertion.

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:2	0h RO	Reserved
1	0h RW	**S3 Power Gate Enable in DC Mode: SUS Well (S3DC_GATE_SUS):** A '1' in this bit enables power gating of the SUS well in S3 while operating on DC power (based on the AC_PRESENT pin value).
0	0h RW	**S3 Power Gate Enable in AC Mode: SUS Well (S3AC_GATE_SUS):** A '1' in this bit enables power gating of the SUS well in S3 while operating on AC power (based on the AC_PRESENT pin value).

9.10.12 S4 Power Gating Policies (S4_PWRGATE_POL)—Offset 2Ch

This register contains policy bits to configure various power gating options while the system is in S4.

Note: Setting any of these policies to "enabled" may not directly result in power gating - in some cases other HW qualifications may be dynamically applied.

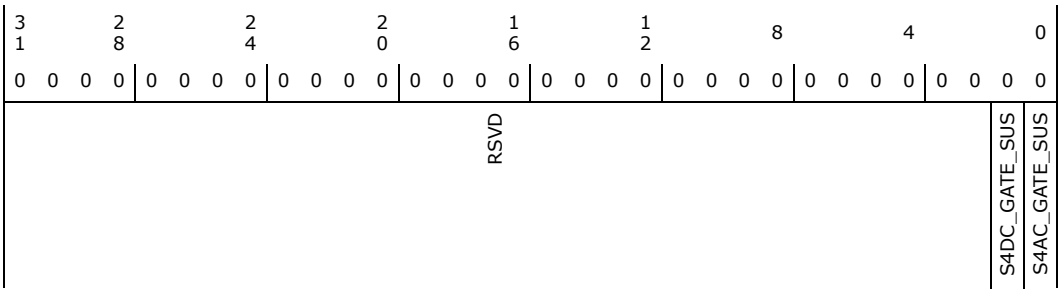
This register is in the RTC power well and is reset by RTCRST# assertion.

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:2	0h RO	Reserved
1	0h RW	**S4 Power Gate Enable in DC Mode: SUS Well (S4DC_GATE_SUS):** A '1' in this bit enables power gating of the SUS well in S4 while operating on DC power (based on the AC_PRESENT pin value).
0	0h RW	**S4 Power Gate Enable in AC Mode: SUS Well (S4AC_GATE_SUS):** A '1' in this bit enables power gating of the SUS well in S4 while operating on AC power (based on the AC_PRESENT pin value).

9.10.13 S5 Power Gating Policies (S5_PWRGATE_POL)—Offset 30h

This register contains policy bits to configure various power gating options while the system is in S5.

Note: Setting any of these policies to "enabled" may not directly result in power gating - in some cases other HW qualifications may be dynamically applied.

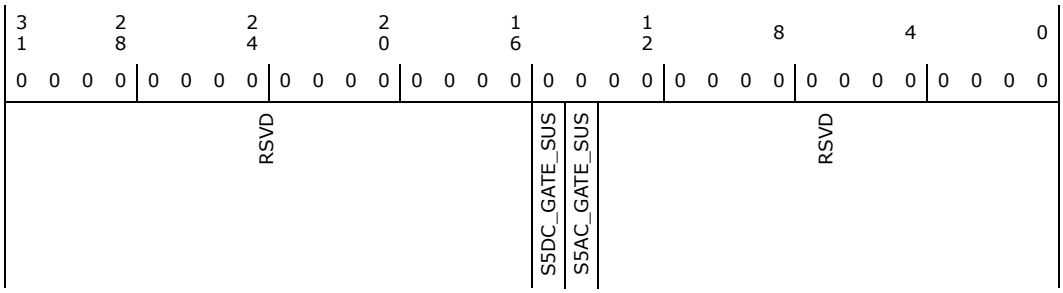
This register is in the RTC power well and is reset by RTCRST# assertion.

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0h RO	Reserved
15	0h RW	**S5 Power Gate Enable in DC Mode: SUS Well (S5DC_GATE_SUS):** A '1' in this bit enables power gating of the SUS well in S5 while operating on DC power (based on the AC_PRESENT pin value).
14	0h RW	**S5 Power Gate Enable in AC Mode: SUS Well (S5AC_GATE_SUS):** A '1' in this bit enables power gating of the SUS well in S5 while operating on AC power (based on the AC_PRESENT pin value).
13:0	0h RO	Reserved

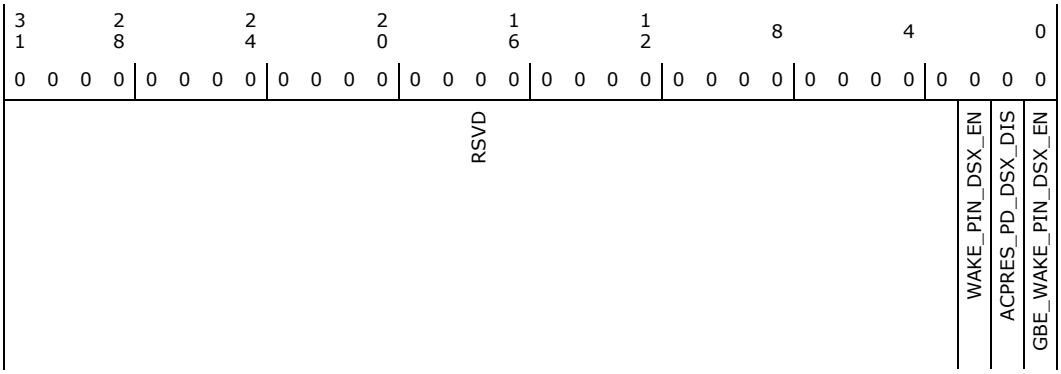
9.10.14 DeepSx Configuration (DSX_CFG)—Offset 34h

This register contains misc. fields used to configure the PCH's power management behavior.
This register is in the RTC power well and is reset by RTCRST# assertion.

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)
Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:3	0h RO	Reserved
2	0h RW	**WAKE# Pin DeepSx Enable (WAKE_PIN_DSX_EN):** When this bit is 1, the PCI Express WAKE# pin is monitored while in Deep Sx, supporting waking from Deep Sx due to assertion of this pin. In this case, the platform must externally pull up the pin to the DSW (instead of pulling up to the SUS as has historically been the case). When this bit is 0: - DeepSx enabled configurations: The PCH internal pull-down on the WAKE# pin is enabled in Deep Sx and during G3 exit and the pin is not monitored during this time. -Deep Sx disabled configurations: The PCH internal pull-down on the WAKE# pin is never enabled. ***Note:*** Deep Sx disabled configurations must leave this bit at 0.
1	0h RW	**AC_PRESENT Pin Pulldown in DeepSx Disable (ACPRES_PD_DSX_DIS):** When this bit is 1, the internal pull-down on the ACPRESENT pin is disabled. However, the pulldown is not necessarily enabled if the bit is '0'. This bit must be left at '0' for Deep Sx disabled configurations, and the pulldown is disabled for those configurations even though the bit is '0'. To support ME wakes from Deep Sx using MGPIO2, the pin is always monitored regardless of the value of this host policy bit. When this bit is '0': DeepSx enabled configurations: The PCH internal pull-down on ACPRESENT is enabled in Deep Sx and during G3 exit. Deep Sx disabled configurations: The PCH internal pull-down on ACPRESENT is always disabled.
0	0h RW	**GbE WAKE pin in DeepSx Enable (GbE_WAKE_PIN_DSX_EN):** When this bit is 1b, the GPD2_GBE_WAKE_N pin is monitored while in DeepSX, supporting waking from DeepSX due to assertion of this pin. DeepSX disabled configurations must leave this bit 0. When this bit is 0: DeepSX enabled configurations: The PCH internal pulldown on GBE_WAKE_N is enabled in DeepSX and during G3 exit, and the pin is not monitored during this time. DeepSX disabled configurations: The PCH internal pulldown is never enabled.

9.10.15 Power Management Configuration Reg 2 (PM_CFG2)—Offset 3Ch

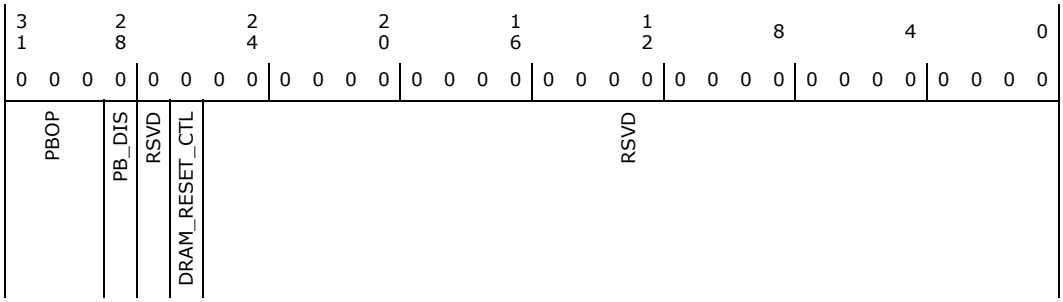
Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31

Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:29	0h RW	**Power Button Override Period (PBOP):** This field determines, while the power button remains asserted, how long the PMC will wait before initiating a global reset. Encoding: 000b - 4 seconds 001b - 6 seconds 010b - 8 seconds 011b - 10 seconds 100b - 12 seconds 101b - 14 seconds Others - Reserved This bit is reset by DSW_PWROK de-assertion.
28	0h RW	**Power Button Native Mode Disable (PB_DIS):** When this bit is '0' (default), the PMC's power button logic will act upon the input value from the GPIO unit, as normal. When this bit is set to '1', the PMC must force its internal version of the power button pin to '1'. This will result in the PMC logic constantly seeing the pin as de-asserted. This bit is reset by RTCRST# assertion.
27	0h RO	Reserved
26	0h RW/V	**DRAM_RESET# Control (DRAM_RESET_CTL):** BIOS uses this bit to control the DRAM_RESET# pin from the PCH, which is routed to the reset pin on the DRAM. Encoding: 0 = DRAM_RESET# output is asserted (driven low) 1 = DRAM_RESET# output is tri-stated. ***Note:*** This bit is cleared to '0' by HW when SLP_S4# goes low. This bit is reset by DSW_PWROK de-assertion.
25:0	0h RO	Reserved

9.10.16 Chipset Initialization Register 40 (CIR40)—Offset 40h

The BIOS may program this register.

9.10.17 Chipset Initialization Register 44 (CIR44)—Offset 44h

The BIOS may program this register.

9.10.18 Chipset Initialization Register 48 (CIR48)—Offset 48h

The BIOS may program this register.

9.10.19 Chipset Initialization Register 4C (CIR4C)—Offset 4Ch

The BIOS may program this register.

9.10.20 Chipset Initialization Register 50 (CIR50)—Offset 50h

The BIOS may program this register.

9.10.21 Chipset Initialization Register 54 (CIR54)—Offset 54h

The BIOS may program this register.

9.10.22 Chipset Initialization Register 58 (CIR58)—Offset 58h

The BIOS may program this register.

9.10.23 Chipset Initialization Register 60 (CIR60)—Offset 60h

The BIOS may program this register.

9.10.24 Chipset Initialization Register 68 (CIR68)—Offset 68h

The BIOS may program this register.

9.10.25 Chipset Initialization Register 78 (CIR78)—Offset 78h

The BIOS may program this register.

9.10.26 Chipset Initialization Register 7C (CIR7C)—Offset 7Ch

The BIOS may program this register.

9.10.27 Chipset Initialization Register 80 (CIR80)—Offset 80h

The BIOS may program this register.

9.10.28 Chipset Initialization Register 84 (CIR84)—Offset 84h

The BIOS may program this register.

9.10.29 Chipset Initialization Register 88 (CIR88)—Offset 88h

The BIOS may program this register.

9.10.30 Chipset Initialization Register 8C (CIR8C)—Offset 8Ch

The BIOS may program this register.

9.10.31 Chipset Initialization Register 90 (CIR90)—Offset 90h

The BIOS may program this register.

9.10.32 Chipset Initialization Register 98 (CIR98)—Offset 98h

The BIOS may program this register.

9.10.33 Chipset Initialization Register A0 (CIRA0)—Offset A0h

The BIOS may program this register.

9.10.34 Chipset Initialization Register A4 (CIRA4)—Offset A4h

The BIOS may program this register.

9.10.35 Chipset Initialization Register A8 (CIRA8)—Offset A8h

The BIOS may program this register.

9.10.36 Chipset Initialization Register AC (CIRAC)—Offset ACh

The BIOS may program this register.

9.10.37 Chipset Initialization Register B0 (CIRB0)—Offset B0h

The BIOS may program this register.

9.10.38 Chipset Initialization Register B4 (CIRB4)—Offset B4h

The BIOS may program this register.

9.10.39 Chipset Initialization Register C0 (CIRC0)—Offset C0h

The BIOS may program this register.

9.10.40 Chipset Initialization Register C4 (CIRC4)—Offset C4h

The BIOS may program this register.

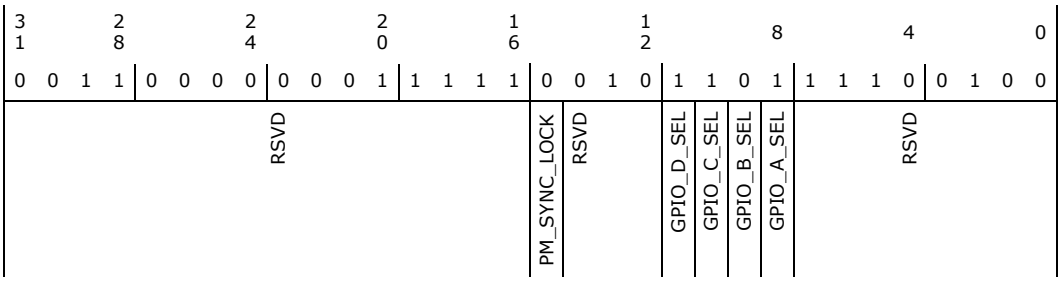
9.10.41 PM_Sync Miscellaneous Configuration (PM_SYNC_MISC_CFG)—Offset C8h

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0h RO	**Reserved**
15	0b RW/L	**PM_SYNC configuration lock (PM_SYNC_LOCK):** This bit is used to lock down the settings of several PM_SYNC related configuration bits. This bit is self locking. Writing a 1b to this will cause the bit to be locked until a PLTRST# occurs.
14:12	0h RO	**Reserved**
11	0b RW/L	**GPIO_D Pin Selection (GPIO_D_SEL)**: If this bit is 0, the GPIO_D = CPU_GP_3. If the bit is 1 then it is CPU_GP_2. This bit is not writeable once PM_SYNC_LOCK = 1.
10	0b RW/L	**GPIO_C Pin Selection (GPIO_C_SEL)**: If this bit is 0, the GPIO_C = CPU_GP_0. If the bit is 1 then it is CPU_GP_1. This bit is not writeable once PM_SYNC_LOCK = 1.
9	0b RW/L	**GPIO_B Pin Selection (GPIO_B_SEL)**: If this bit is 0, the GPIO_B = CPU_GP_2. If the bit is 1 then it is CPU_GP_0. This bit is not writeable once PM_SYNC_LOCK = 1.
8	0b RW/L	**GPIO_A Pin Selection (GPIO_A_SEL)**: If this bit is 0, the GPIO_A = CPU_GP_1. If the bit is 1 then it is CPU_GP_3. This bit is not writeable once PM_SYNC_LOCK = 1.
7:0	0h RO	Reserved

9.10.42 Chipset Initialization Register D0 (CIRD0)—Offset D0h

The BIOS may program this register.

9.10.43 Chipset Initialization Register D4 (CIRD4)—Offset D4h

The BIOS may program this register.

9.10.44 Chipset Initialization Register DC (CIRDC)—Offset DCh

The BIOS may program this register.

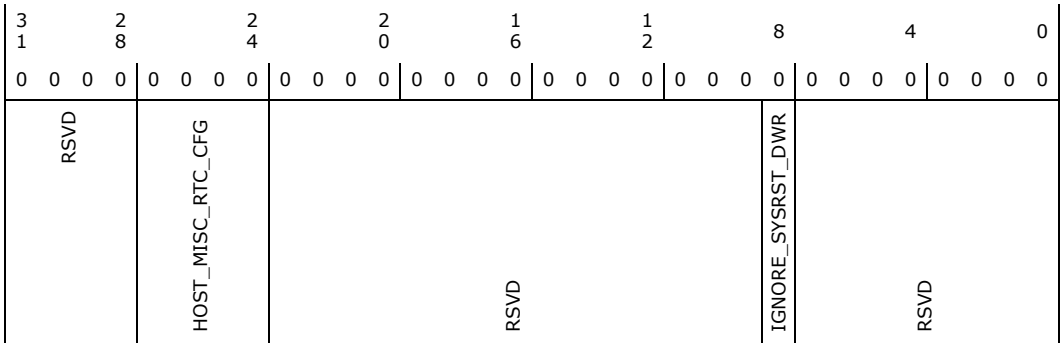
9.10.45 Chipset Initialization Register E0 (CIRE0)—Offset E0h

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:28	4h RO	**RSVD**
27:24	0h RW/L	**Host Miscellaneous RTC Config(HOST_MISC_RTC_CFG):** For those parts of the die that are able to be power gated, these bits determine how long the power management controller will wait for them to get to their default state before going into a global reset. The PMC will serially wait this amount of time for each domain. The number of power domains is silicon dependent. For the Intel® C620 Series Chipset, the total time the PMC controller will wait is approximately 20x the value. Bits 27:26 Reserved Bits 25:24 00 - 1 us 01 - 2 us 10 - 5 us 11 - 20 us
23:9	0000h RW, RO	Reserved
8	0b RW	**Ignore SYSRST on Demoted Warm Reset (IGNORE_SYSRST_DWR):** HOST/BIOS will set this bit to 1 if the SYSRST# input for reset requests it to be ignored during the Demoted Warm Reset (DWR) process.
7:0	00h RW, RO	Reserved

9.10.46 Chipset Initialization Register E4 (CIRE4)—Offset E4h

The BIOS may program this register.

9.10.47 Chipset Initialization Register E8 (CIRE8)—Offset E8h

The BIOS may program this register.

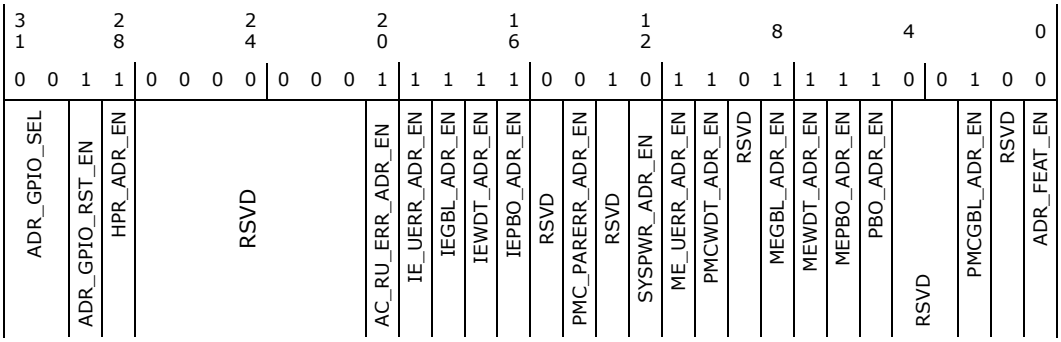
9.10.48 ADR_EN - ADR Enable - Offset F0h

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:30	00b RW	**ADR GPIO Selection (ADR_GPIO_SEL):**. This field selects which PM_SYNC GPIO state will be a source of ADR. The input may also be a source of a global reset based on ADR_GPIO_RST_EN. 00 GPIO_A 01 GPIO_B 10 GPIO_C 11 GPIO_D ***Note:*** The PCH only supports use of CPU_GP_0 as the board level trigger of ADR. Therefore, this field must be programmed to the PM_SYNC GPIO state which is assigned to the CPU_GP_0 in the PM_SYNC_MISC_CFG register (which will be GPIO_B or GPIO_C)
29	1b RW	**ADR GPIO Reset Enable (ADR_GPIO_RST_EN):** If this bit is set to '1'b, a Global Reset will be generated when the ADR timer expires after being triggered by the ADR_GPIO selected in the ADR_GPIO_SEL field and the feature is enabled both by fusing and ADR_FEATEN = 1. If this bit is cleared, the ADR GPIO may still be enabled to affect the PM_SYNC pin, but on it's own it will not trigger a global reset. ***Note:*** This enable bit is not required for this source to trigger the warning to the CPU over PM_SYNC. This bit only impacts whether or not this source will actually trigger a global reset.
28	1b RW	**Host Partition Reset ADR Enable (HPR_ADR_EN):** If this bit is to "1", PMC FW will set the ADR_RST_STS bit upon receipt of the Reset_Warn_ACK DMI message.
27:21	00h RO	**Reserved**
20	1b RW/V	**AC RU Error ADR Enable (AC_RU_ERR_ADR_EN):** If this bit is set to "1", the reset unit for QAT and 10/1GbE LAN will have it's error global reset resource included in the ADR feature.
19	1b RW/v	**IE Uncorrectable Error ADR Enable (IE_UERR_ADR_EN):** When set, an uncorrectable error source in IE that causes a global reset is included in the ADR feature
18	1b RW/V	**IE Initiated Global Reset ADR Enable (IEGBL_ADR_EN)**: When set to '1'. the IE initiated global reset source is included in the ADR enable.
17	1b RW/V	**IE FW watchdog Timer ADR Enable (IEWDT_ADR_EN)**: If set to "1", the IE firmware watchdog timer global reset source is included in the ADR feature.
16	1b RW/V	**IE Initiated Power Button Override ADR Enabled (IEPBO_ADR_EN)**: If set to "1", the IE initiated power button override global reset source is included in the ADR feature.

Bit Range	Default & Access	Field Name (ID): Description
15	0b RO	**Reserved**
14	0b RW/V	**PMC Parity Error ADR Enable (PMC_PARERR_ADR_EN).** If this bit is set to '1', the PMC RAM parity error global reset sources is included in the ADR feature.
13	1b RW/V	**Reserved**
12	0b RW/V	**SYS_PWROK Failure ADR Enabled (SYSPWR_ADR_EN).** If this bit is set to '1', the SYS_PWROK failure global reset source is included in the ADR feature
11	1b RW/V	**ME Uncorrectable Error ADR Enable (ME_UERR_ADR_EN):** If set to '1', the Intel ME uncorrectable error global reset source is included in the ADR enable
10	1b RW/V	**PMC FW Watchdog timer ADR enable (PMCWDT_ADR_EN)**; If set to '1', the PMC firmware watchdog timer global reset source is included in the ADR feature.
9	0b RO	**Reserved**
8	1b RW/V	**ME Initiated Global Reset ADR enable (MEGBL_ADR_EN)**: If set to "1", the Intel ME initiated global reset source is included in the ADR feature.
7	1b RW/V	**ME FW Watchdog Timer ADR Enable (MEWDT_ADR_EN):** If set to "1", the Intel ME FW Watchdog timer global reset source is included in the ADR feature.
6	1B RW/V	**ME Initiated Power Button Override ADR Enable (MEPBO_ADR_EN):** If set to "1", the Intel ME initiated Power Button Override global reset source is included in the ADR feature.
5	1b RW/V	**Power Button Override ADR Enable (PBO_ADR_EN**): If set to "1", the Power Button Override global reset source is included in the ADR feature.
4:3	00b RO	**Reserved**
2	1b RW/V	**PMC FW Initiated Global Reset ADR Enable (PMCGBL_ADR_EN):** If set to "1", PMC FW initiated global reset sources are included in the ADR feature. These sources are: Host Partition Reset entry timeout Any host partition reset triggered with the "Promote Host Partition Reset to Global Reset" policy bit set to "1"
1	RO	**Reserved**
0	0h RW/1V	**ADR Feature Enable (ADR_FEAT_EN) -:** If this bit is a '1' and the feature is enabled in the fusing, PCH support for Automatic DIMM Self-Refresh is enabled. The other bits in this register determine which individual reset sources will trigger an ADR. ***Note:*** This bit must not be set to '1' until PM_SYNC has been fully configured to convey the global reset warning to the CPU.

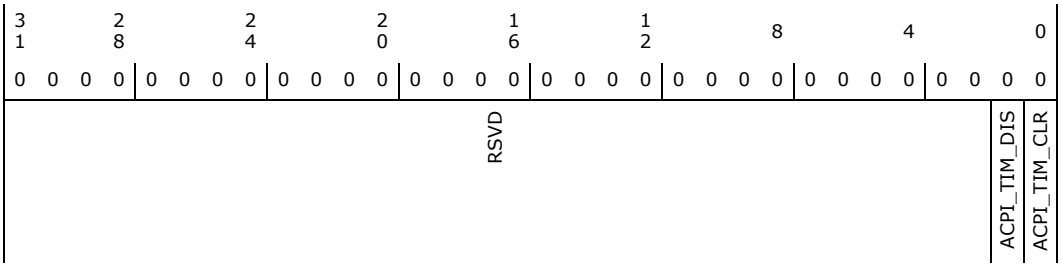
9.10.49 ACPI Timer Control (ACPI_TMR_CTL)—Offset FCh

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:2	0h RO	Reserved
1	0h RW	**ACPI Timer Disable (ACPI_TIM_DIS):** This bit determines whether the ACPI Timer is enabled to run. Note that even when enabled, the timer only runs during S0. - 0: ACPI Timer is enabled (default) - 1: ACPI Timer is disabled (halted at the current value) This bit is reset by PLTRST# assertion.
0	0h RW/1S/V	**ACPI Timer Clear (ACPI_TIM_CLR):** Writing a 1 to this bit will clear the ACPI Timer to all 0s. Hardware will automatically clear the bit back to 0 once the timer clear operation has completed. Writing a 0 to this bit has no effect. Implementation ***Note:*** The PCH must be capable of honoring this bit even while ACPI_TIM_DIS=1. This bit is reset by PLTRST# assertion.

9.10.50 Global to Host Reset Enable(GBL2HOST_EN)—Offset 10Ch

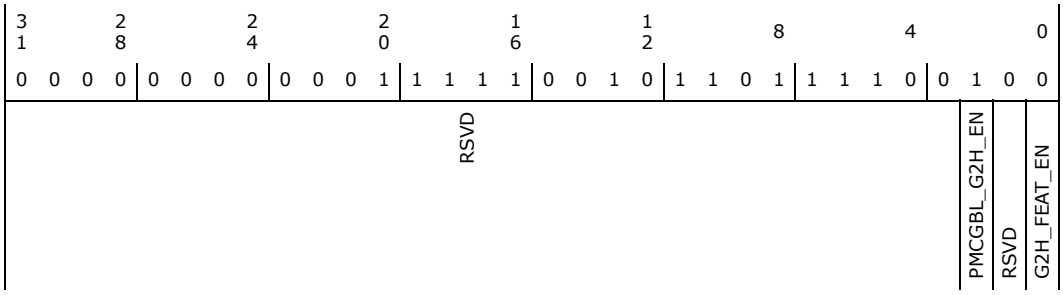
This register allows software to demote a global reset initiated by the PMC controller down to a host portion reset. This is used to keep resets that would cause a power cycle, and losing error status in the CPU, from occurring. When you enable a Global to Host reset by setting bit 0 = 1b, the only bit that can then be a "1" is bit 2. All other bits must be set to 0b.

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:3	0003E4BCh RO	Reserved. BIOS will write 0h to these registers.
2	0b RW/L	**Power Management Global Reset to Host enable**: When set to 1b and Bit0 of this register is a 1b, then the following errors are converted to a host reset. Host partition reset entry timeout
1	0b RO	Reserved.
0	0b RW/L	**G2H Feature Enable (G2H_FEAT_EN):** When set to 1b, the PCH will support conversion of global resets to warm resets. Which global resets are converted are defined by the other bits in this register.

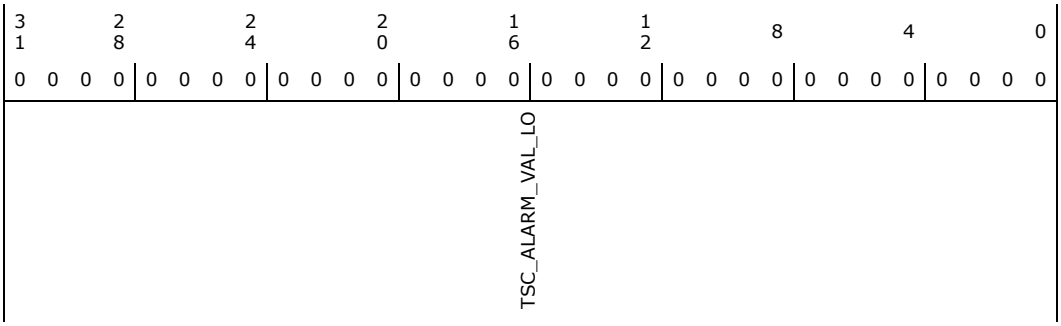
9.10.51 Last TSC Alarm Value[31:0] (TSC_ALARM_LO)—Offset 110h

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0h RO/V	**Last TSC Alarm Value [31:0] (TSC_ALARM_VAL_LO):** This field contains bits 31:0 of the last TSC alarm value received from the CPU. This field is reset by PLTRST# assertion.

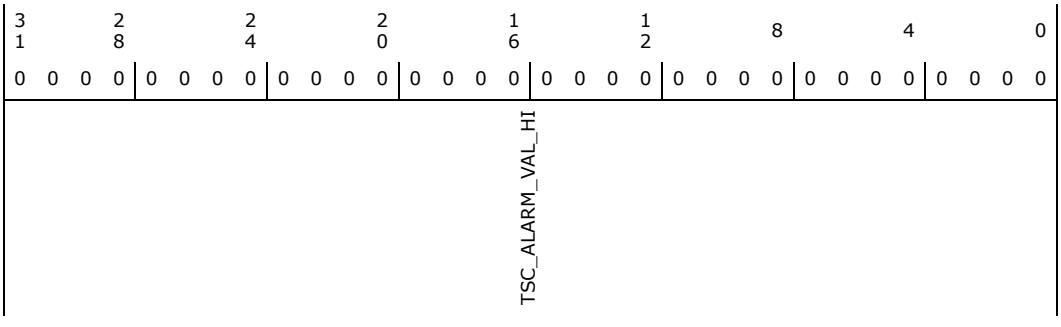
9.10.52 Last TSC Alarm Value[63:32] (TSC_ALARM_HI)—Offset 114h

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0h RO/V	**Last TSC Alarm Value [63:32] (TSC_ALARM_VAL_HI):** This field contains bits 63:32 of the last TSC alarm value received from the CPU. This field is reset by PLTRST# assertion.

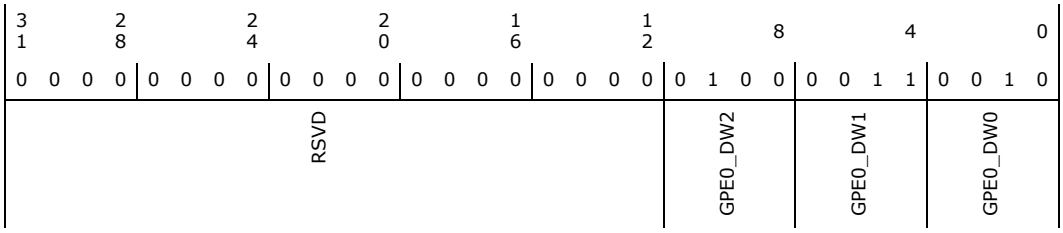
9.10.53 GPIO Configuration (GPIO_CFG)—Offset 120h

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31
Function: 2

Default: 432h



Bit Range	Default & Access	Field Name (ID): Description
31:12	0h RO	Reserved.
11:8	4h RW	**GPIO Group to GPE_DW2 assignment encoding (GPE0_DW2):** This register assigns a specific GPIO Group to the ACPI GPE0[96:64]. 0h = reserved[31:24]&GPP_A[23:0] 1h = reserved[31:24]&GPP_B[23:0] 2h = reserved[31:24]&GPP_C[23:0] 3h = reserved[31:24]&GPP_D[23:0] 4h = reserved[31:13]&GPP_E[12:0] 5h = reserved[31:24&GPP_F[23:0] 6h = reserved[31:24]&GPP_G[23:0] 7h = reserved[31:24]&GP_H[23:0] 8h = reserved[31:11]&GPP_I[10:0] 9h = reserved[31:12]&GPD[11_0] Ah = reserved[31:24]&GPP_J[23:0] Bh = reserved[31:12]&GPP_K[11:0]
7:4	3h RW	**GPIO Group to GPE_DW1 assignment encoding (GPE0_DW1):** This register assigns a specific GPIO Group to the ACPI GPE0[63:32]. 0h = reserved[31:24]&GPP_A[23:0] 1h = reserved[31:24&GPP_B[23:0] 2h = reserved[31:24]&GPP_C[23:0] 3h = reserved[731:24]&GPP_D[23:0] 4h = reserved[31:13]&GPP_E[12:0] 5h = reserved[31:24]&GPP_F[23:0] 6h = reserved[31:24&GPP_G[23:0] 7h = reserved[31:24]&GP_H[23:0] 8h = reserved[31:11]&GPP_I[10:0] 9h = reserved[31:12]&GPD[11_0] Ah = reserved[31:24]&GPP_J[23:0] Bh = reserved[31:12]&GPP_K[11:0]
3:0	2h RW	**GPIO Group to GPE_DW0 assignment encoding (GPE0_DW0):** This register assigns a specific GPIO Group to the ACPI GPE0[31:0]. 0h = reserved[31:24]&GPP_A[23:0] 1h = reserved[31:24]&GPP_B[23:0] 2h = reserved[31:24]&GPP_C[23:0] 3h = reserved[31:24]&GPP_D[23:0] 4h = reserved[31:23]&GPP_E[12:0] 5h = reserved[31:24]&GPP_F[23:0] 6h = reserved[31:24]&GPP_G[23:0] 7h = reserved[31:24]&GP_H[23:0] 8h = reserved[31:11]&GPP_I[10:0] 9h = reserved[31:12]&GPD[11_0] Ah = reserved[31:24]&GPP_J[23:0] Bh = reserved[31:12]&GPP_K[11:0

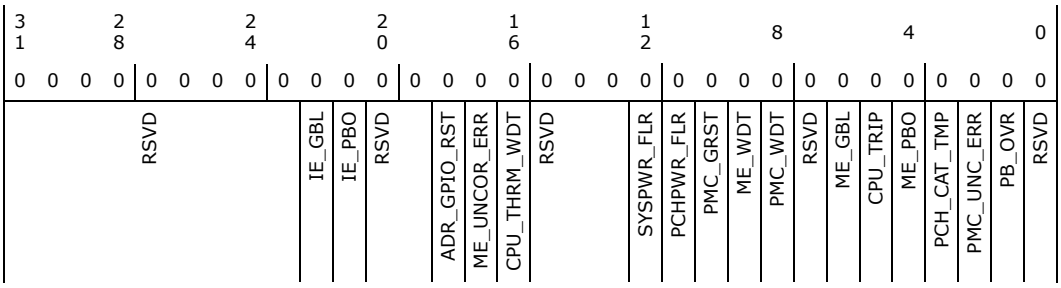
9.10.54 Global Reset Causes Register 0 (GBLRST_CAUSE0) - Offset 124h

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31
Function: 2

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:23	0b RW/1C/V	**RSVD**
22	0b RW/1C/V	**IE Initiated Global Reset (IE_GBL):** This bit is set by hardware when a global reset is triggered by an IE FW. This bit is reset by DSW_PWROK deassertion.
21	0b RW/1C/V	**IE Initiated Power Button Override (IE_PB)**: This bit is set by hardware when a power button override is triggered by IE FW. This bit is reset by DSW_PWROK deassertion.
20:19	0b RW/1C/V	**RSVD**
18	0b RW/1C/V	**ADR GPIO Reset (ADR_GPIO_RST):** This bit is set by hardware when a global reset is triggered by the assertion of the GPIO assigned to ADR. This bit is reset by DSW_PWROK deassertion.
17	0b RW/1C/V	**Intel ME HW Uncorrectable Error (ME_UNCOR_ERR)**: This bit is set by hardware when a global reset is triggered by Intel ME hardware due to the detection of an uncorrectable ECC or parity error on a data read of one of it's SRAMs. This bit is reset by DSW_PWROK deassertion.
16	0b RW/1C/V	**CPU Thermal Runaway Watchdog Timer (CPU_THRM_WDT)**: This bit is set by hardware when a global reset is triggered by the expiration of the CPU Thermal Runaway Watchdog timer. This bit is reset by DSW_PWROK deassertion.
15:13	0b RW/1C/V	**RSVD**
12	0b RW/1C/V	**SYS_PWROK Failure (SYSPWR_FLR)**: This bit is set by hardware when a global reset is triggered by an unexpected loss of SYS_PWROK. This bit is reset by DSW_PWROK deassertion.
11	0b RW/1C/V	**PCH_PWROK Failure (PCHPWR_FLR):** This bit is set by hardware when a global reset is triggered by an unexpected loss of PCH_PWROK. This bit is reset by DSW_PWROK deassertion.
10	0b RW/1C/V	**Power Management Global Reset (PMC_GRST):** This bit is set by hardware when a global reset is triggered by a request from the power management controller. This bit is reset by DSW_PWROK deassertion.
9	0b RW/1C/V	**Intel Management Engine Watchdog Timer (ME_WDT)**: This bit is set by hardware when a global reset is triggered by the second expiration of the Intel Management Engine watchdog timer. This bit is reset by DSW_PWROK deassertion.
8	0b RW/1C/V	**Power Management Controller Watchdog Timer (PMC_WDT)**: This bit is set by hardware when a global reset is triggered by the second expirationof the PMC watchdog timer. This bit is reset by DSW_PWROK deassertion.

Bit Range	Default & Access	Field Name (ID): Description
7	0b RW/1C/V	**RSVD**
6	0b RW/1C/V	**ME Initiated Global Reset (ME_GBL)**: This bit is set by hardware when a global reset is triggered by Intel ME FW. This bit is reset by DSW_PWROK deassertion.
5	0b RW/1C/V	**Processor Thermal Trip (PROC_TRIP):** This bit is set to 1b by hardware when a global reset is triggered by a processor thermal event (i.e., an assertion of the THRMTRIP# pin). This bit is reset by DSW_PWROK deassertion.
4	0b RW/1C/V	**ME Initiated Power Button Override (ME_PBO):** This bit is set by hardware when a global reset is triggered by an Intel ME-initiated power button override. This bit is reset by DSW_PWROK deassertion.
3	0b RW/1C/V	**PCH Catastrophic Temperature Event (PCH_CAT_TMP)**: This bit is set by hardware when a global reset is triggered by a catastrophic temperature event from the PCH internal thermal sensor. This bit is reset by DSW_PWROK deassertion.
2	0b RW/1C/V	**PMC SUS RAM Uncorrectable Error (PMC_UNC_ERR):** This bit is set by hardware when a global reset is triggered due to an uncorrectable parity error on a data read from one of the PMC SUS well register files. This bit is reset by DSW_PWROK deassertion.
1	0b RW/1C/V	**Power Button Override (PB_OVR)**: This bit is set by hardware when a global reset is triggered by a power button override (ie an assertion of the PWRBTN# pin for 5 seconds). This bit is reset by DSW_PWROK deassertion.
00	0b RO	**RSVD**

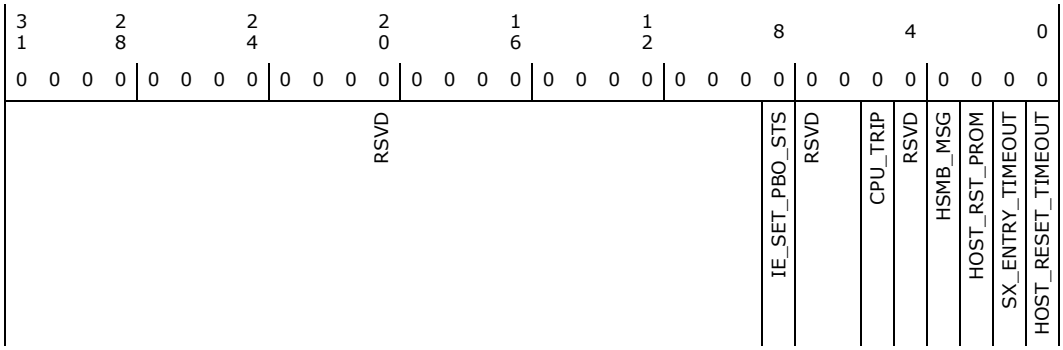
9.10.55 Global Reset Causes Register 1(GBLRST_CAUSE1) - Offset 128h

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31
Function: 2

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:9	0h RO	**RSVD**
8	0b RW/1C/V	**IE Set Power Button Status (IE_SET_PBO_STS):** If this bit is set, the cause of the previous global reset was an IE FW setting the power button overrise status. This bit is reset by DSW_PWROK deassertion.
7:6	00b RW/1C/V	**RSVD**

Bit Range	Default & Access	Field Name (ID): Description
5	0b RW/1C/V	**ME Set Power Button Status (ME_SET_PBO_STS):** This bit is reset by DSW_PWROK deassertion.
4	0b RO	**RSVD**
3	0b RW/1C/V	**Host SMBUs Initiated Reset (HSMB_INIT_RST):** If this bit is set, the cause of the previous global reset was a global reset request received over the host SMBus interface. This bit is reset by DSW_PWROK deassertion.
2	0b RW/1C/V	**Host Partition Reset Promotion (HOST_RST_PROM):** If this bit is set, the cause of the previous global reset was a host partition reset that was promoted to a global reset either due to ME or host policy. This bit is reset by DSW_PWROK deassertion.
1	0b RW/1C/V	**SX Entry Timeout (SX_ENTRY_TIMEOUT)**: If this bit is set, the cause of the previous global reset was an expiration of the timer that runs during SX entry. This bit is reset by DSW_PWROK deassertion.
00	0b RW/1C/V	**Host Partition Reset Timeout (HOST_RESET_TIMEOUT).** If this bit is set, the cause of the previous reset was an expiration of the timer that runs during host partition resets. This bit is reset by DSW_PWROK deassertion.

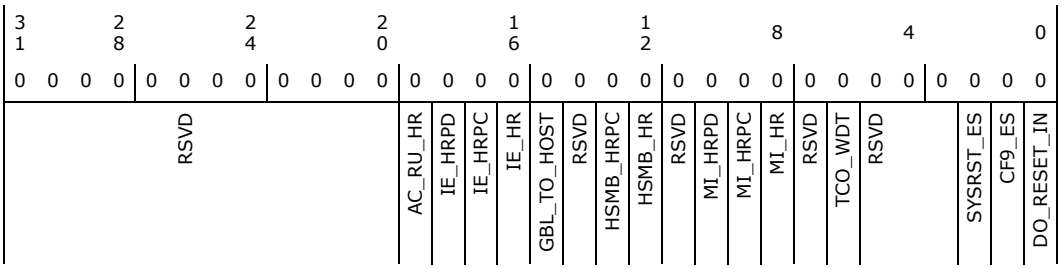
9.10.56 Host Partition Reset Causes (HPR_CAUSE0)l - Offset 12ch

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31
Function: 2

Default: 00000000h



Bit Range	Default & Access	Field Name (ID): Description
31:20	00h RO	**RSVD**
19	0b RO	**AC Reset Unit initiated Host Reset (AC_RU_HR):**
18	0b RO	**IE-Initiated Host Reset with Power Down (IE_HRPD)**
17	0b RO	**IE-Initiated Host Reset with Power Cycle (IE_HRPC)**
16	0b RO	**IE-Initiated Host Reset without Power Cycle (IE_HR)**
15	0b RO	**Global Reset Converted to Host Reset (GBL_TO_HOST):** this bit is set, a global reset occurred while
14	0b RO	**RSVD**

Bit Range	Default & Access	Field Name (ID): Description
13	0b RO	**Host SMBus Host Reset with Power Cycle (HSMB_HRPC):** SMBus initiated host partition reset with power cycle
12	0b RO	**Host SMBus Host Reset without Power Cycle (HSMB_HR):** SMBus initiated host partition reset without power cycle
11	0b RO	**RSVD**
10	0b RO	**ME-Initiated Host Reset with Power Down (MI_HRPD)**
9	0b RO	**ME-Initiated Host Reset with Power Cycle (MI_HRPC)**
8	0b RO	**ME-Initiated Host Reset without Power Cycle (MI_HR)**
7	0b RO	**RSVD**
6	0b RO	**Host TCO Watchdog Timer Second Expiration (TCO_WDT):** Host TCO watchdog timer reached zero for the second time.
5:3	000b RO	**RSVD**
2	0b RO	**SYS_RESET_N (SYSRST_ES):** Assertion of the SYS_RESET_N pin after the 16 ms HW debounce (if that is enabled)
1	0b RO	**Write to CF9 (CF9_ES):** This bit will be set when the host software writes a value of 06h or 0Eh to the CF9 register. ***Note:*** The "shutdown" special cycle from the CPU will also set this bit.
0	0b RO	**DO_RESET_IN_N (DO_RESET_IN):** DO_RESET_N pin asserted into the PCH while the PCH was in S0.

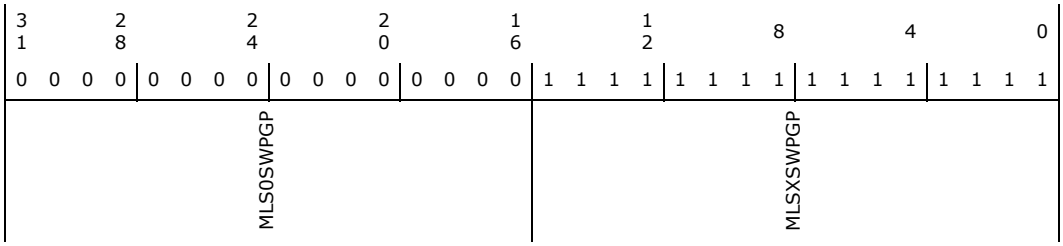
9.10.57 (MODPHY_PM_CFG1)—Offset 200h

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31
Function: 2

Default: FFFFh



Bit Range	Default & Access	Field Name (ID): Description
31:16	0h RW	**MODPHY Lane S0 SUS Well Power Gating Policy [15:0] (MLS0SWPGP):** This is a bit per lane that controls SUS Well Power Gating for a ModPHY lane to be used for S0 and S0ix use models as described above in HAS. Bit 0: Corresponds to ModPHY Lane 0 Bit 1: Corresponds to ModPHY Lane 1 Bit 2: Corresponds to ModPHY Lane 2 : : Bit 15: Corresponds to ModPHY Lane 15 For each lane: 0: Lane power gating not permitted in S0. 1: Lane power gating is permitted in S0. Note that strap information/PCIe Lane reversal information will be factored by BIOS when programming these bits in the configuration registers. Note that it is illegal SW programming to have a bit location to be 1 in this field and the corresponding bit position to be 0 in MLSXSWPGP
15:0	FFFFh RW	**MODPHY Lane Sx SUS Well Power Gating Policy [15:0] (MLSXSWPGP):** This is a bit per lane that controls SUS Well Power Gating for a ModPHY lane when system is in Sx. Bit 0: Corresponds to ModPHY Lane 0 Bit 1: Corresponds to ModPHY Lane 1 Bit 2: Corresponds to ModPHY Lane 2 : : Bit 15: Corresponds to ModPHY Lane 15 For each lane: 0: Lane power gating not permitted in Sx. 1: Lane power gating is permitted in Sx. Note that strap information/PCIe Lane reversal information will be factored by BIOS when programming these bits in the configuration registers. For ease of PMC implementation, this field will be used to manage Sx policies even in S0. In other words, the earlier restriction that BIOS does not have to program this field if MLSPDDGE is 1 does not apply any more. BIOS shall set this field appropriately for all cases.

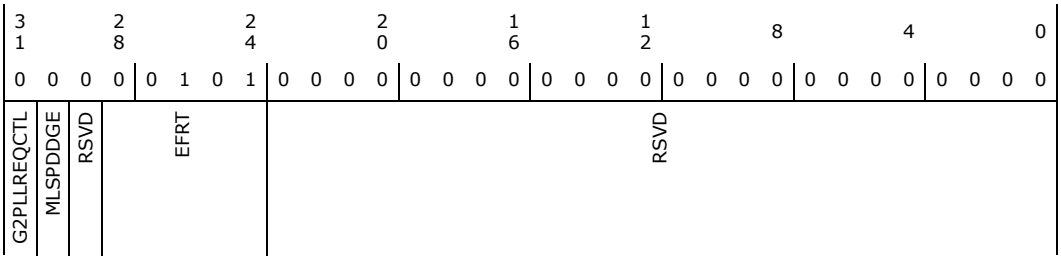
9.10.58 MODPHY Power Management Configuration 2 (MODPHY_PM_CFG2)—Offset 204h

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31
Function: 2

Default: 5000000h



Bit Range	Default & Access	Field Name (ID): Description
31	0h RW	**Gen2PLL Request Control (G2PLLREQCTL):** This bit controls how PMC should treat Gen2PLL power request for ModPHY power gating flows. 0 = PMC should treat Gen2PLL request as non restore power request 1 = PMC should treat Gen2PLL request as restore power request
30	0h RW	**MODPHY Lane SUS Power Domain Dynamic Gating Enable (MLSPDDGE):** When this bit is set to 1, MODPHY Lane SUS Well Dynamic Gating is enabled. When this bit is 0, MODPHY Lane SUS Well Gating can still be done at a more coarse level using MLSXSWPGP and MLS0SWPGP fields. This bit is reset by PLTRST# assertion.
29	0h RO	Reserved
28:24	5h RW	**External FET Ramp Time (EFRT):** This field defines the ramp time of ModPHY FET. 00000b: 00us 00001b: 20us 00010b: 40us ... 00101b: 100us ... 11111b: 620us
23:0	0h RO	Reserved

9.10.59 MODPHY Power Management Configuration 3 (MODPHY_PM_CFG3)—Offset 208h

9.10.60 MODPHY Power Management Configuration 4 (MODPHY_PM_CFG4)—Offset 20Ch

9.10.61 Chipset Initialization Register 324 (CIR324)—Offset 324h

The BIOS may program this register.

9.10.62 Chipset Initialization Register 328 (CIR328)—Offset 328h

The BIOS may program this register.

9.10.63 Chipset Initialization Register 32C (CIR32C)—Offset 32Ch

The BIOS may program this register.

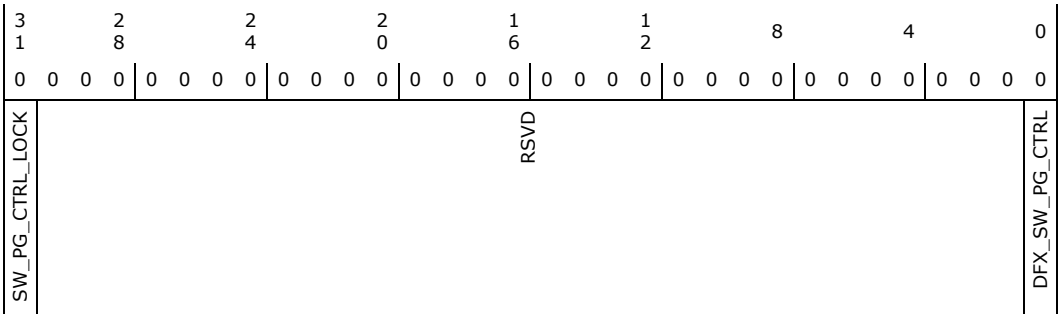
9.10.64 Host SW PG Control Register 1 (HSWPGCR1) —Offset 5D0h

Access Method

Type: PWRMBASE MEM Register (Size: 32 bits)

Device: 31
Function: 2

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31	0h RW/L	**SW PG Req Control Lock (SW_PG_CTRL_LOCK):** 0: All other bits in this register are RW (can be set or cleared). 1: All bits in this register are locked (including this bit). ***Note:*** The BIOS is expected to always write to this bit before handing off control to the OS, even if it has not changed any of the values of this register. This is because the lock bit resets on platform reset, and needs to be set on every boot to S0.
30:1	0h RO	Reserved
0	0h RW/L	**DFX SW PG Req Control (DFX_SW_PG_CTRL):** 0: DFX PGD is not requested to power gate. DFX logic remains active and available for use. 1: DFX PGD is requested to power-gate by the PMC.

§

10 Direct Media Interface

10.1 Acronyms

Acronyms	Description
DMI	Direct Media Interface

10.2 References

Specification	Location
PCI Express Specification*	http://www.pcisig.com/specifications

10.3 Overview

The PCH communicates with the processor using high speed DMI which supports 8 GT/s data rates.

10.4 Signal Description

Name	Type	Description
DMI_RXN[3:0] DMI_RXP[3:0]	I	DMI receive lanes
DMI_TXN[3:0] DMI_TXP[3:0]	O	DMI transmit lanes

10.5 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
DMI_RXN[3:0] DMI_RXP[3:0]	Pull-Down	14K - 26K	
DMI_TXN[3:0] DMI_TXP[3:0]	Pull-Up	14K - 26K	

Note: DMI_TX* pins are terminated to VCC/2 and DMI_RX* pins are terminated to VSS.

10.6 I/O Signal Planes and States

Signal Name	Power Plane	During Reset	Immediately after Reset	S3/S4/S5	Deep Sx
DMI_RXN[3:0] DMI_RXP[3:0]	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	Off
DMI_TXN[3:0] DMI_TXP[3:0]	Primary	Internal Pull-up	Internal Pull-up	Internal Pull-up	Off

10.7 Functional Description

PCH DMI is compliant to the DMI 2.0 specification with a bit rate of 2.5GT/s, 5.0GT/s and 8.0GT/s. The DMI Link is compliant to the PCI Express 3.0 specification for the root complex internal links that support up to 8.0 GT/s, with the exceptions called out in the DMI 2.0 specification and in this document.

Some of key features besides PCI Express Specifications are listed below:

- Addition of LT Memory Write and LT Memory Read TLPs.
- All virtual channels other than VC0 are private and not exposed to the OS.
- Non-unique Transaction IDs are allowed on DMI.
- Downstream requests restricted to VC0 (CPU and peer).
- Shorter than conventional DMI Link Reset sequence.
- DMI can be set to x1 by soft straps for debug. Normal functional operation is x4
- Upstream I/O and Config cycles are not supported.
- DMI does not implement the PCI Express defined Root Complex Register Block and is not OS visible.
- DMI can support data packet sizes up to 256 bytes, and will pass through transactions at whatever data sizes it is presented with (64, 128, 256). The CPU has settings in it’s DMI port for what size maximum data payload it can handle. The max data payload at the PCH I/O ports must not be bigger than what the CPU can handle, and conversely the max CPU data payload sizes must not be bigger than what the I/O ports in the PCH can handle. It is legal to send data payloads that are smaller than the max payloads.

§

11 System Management

11.1 Acronyms

Acronyms	Description
BMC	Baseboard Management Controller
SPD	Serial Presence Detect
TCO	Total Cost of Ownership

11.2 Overview

The PCH provides various functions to make a system easier to manage and to lower the TCO of the system. Features and functions can be augmented using external A/D converters and GPIO, as well as an external micro-controller.

11.3 Feature

The following features and functions are supported by the PCH:

- First timer timeout to generate SMI# after programmable time
  - The first timer timeout causes SMI#, allowing SMM-based recovery from OS lock up.
- Second hard-coded timer timeout to generate reboot.
  - This second timer is used only after the first timeout occurs.
  - The second timeout allows for automatic system reset and reboot if HW error detected.
  - Option to prevent reset the second timeout via HW strap
- Processor present detection
  - Detects if processor fails to fetch the first instruction after reset.
- Various Error detection (such as ECC Errors) indicated by host controller
  - Can generate SMI#, SCI, SERR, NMI, or TCO interrupt.
- Intruder Detect input
  - Can generate TCO interrupt or SMI# when the system cover is removed.
  - INTRUDER# is allowed to go active in any power state, including G3.

11.3.1 Theory of Operation

The System Management functions are designed to allow the system to diagnose failing subsystems. The intent of this logic is that some of the system management functionality can be provided without the aid of an external microcontroller.

11.3.1.1 Detecting a System Lockup

When the processor is reset, it is expected to fetch its first instruction. If the processor fails to fetch the first instruction after reset, the TCO timer times out twice and the PCH asserts PLTRST#.

11.3.1.2 Handling an Intruder

The PCH has an input signal, INTRUDER#, that can be attached to a switch that is activated by the system's case being open. This input has a two RTC clock debounce. If INTRUDER# goes active (after the debouncer), this will set the INTRD_DET bit in the TCO2_STS register. The INTRD_SEL bits in the TCO_CNT register can enable the PCH to cause an SMI# or interrupt. The BIOS or interrupt handler can then cause a transition to the S5 state by writing to the SLP_EN bit.

The software can also directly read the status of the INTRUDER# signal (high or low) by clearing and then reading the INTRD_DET bit. This allows the signal to be used as a GPI if the intruder function is not required.

If the INTRUDER# signal goes inactive some point after the INTRD_DET bit is written as a 1, then the INTRD_DET bit will go to a 0 when INTRUDER# input signal goes inactive.

Note: This is slightly different than a classic sticky bit, since most sticky bits would remain active indefinitely when the signal goes active and would immediately go inactive when a 1 is written to the bit.

Note: The INTRD_DET bit resides in the PCH's RTC well, and is set and cleared synchronously with the RTC clock. Thus, when software attempts to clear INTRD_DET (by writing a 1 to the bit location) there may be as much as two RTC clocks (about 65 µs) delay before the bit is actually cleared. Also, the INTRUDER# signal should be asserted for a minimum of 1 ms to ensure that the INTRD_DET bit will be set.

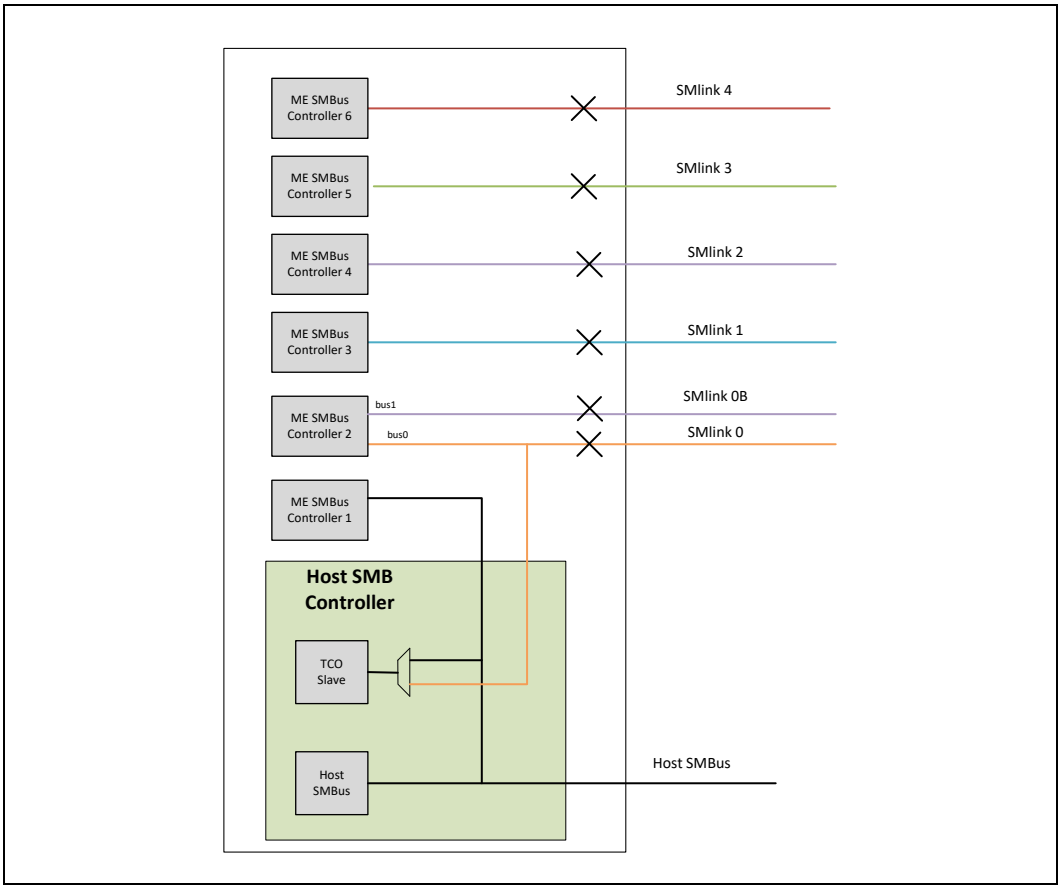
Note: If the INTRUDER# signal is still active when software attempts to clear the INTRD_DET bit, the bit remains set and the SMI is generated again immediately. The SMI handler can clear the INTRD_SEL bits to avoid further SMIs. However, if the INTRUDER# signal goes inactive and then active again, there will not be further SMIs, since the INTRD_SEL bits would select that no SMI# be generated.

11.3.2 TCO Modes

11.3.2.1 TCO Compatible Mode

In TCO Legacy/Compatible mode, only the host SMBus is utilized. The TCO Slave is connected to the host SMBus internally by default. In this mode, the Intel ME SMBus controllers are not used and should be disabled by soft strap. See the latest PCH SPI Flash Programming Guide for more detail.

Figure 11-1. TCO Compatible Mode SMBus Configuration



In TCO Legacy/Compatible mode the PCH can function directly with an external LAN controller or equivalent external LAN controller to report messages to a network management console without the aid of the system processor. This is crucial in cases where the processor is malfunctioning or cannot function due to being in a low-power state. Table 11-1 includes a list of events that will report messages to the network management console.

Table 11-1. Event Transitions that Cause Messages

Event	Assertion?	Deassertion?	Comments
INTRUDER# pin	Yes	No	Must be in "S1 or hung S0" state
Watchdog Timer Expired	Yes	No (NA)	"S1 or hung S0" state entered
SMBALERT# pin	Yes	Yes	Must be in "S1 or hung S0" state
CPU_PWR_FLR	Yes	No	"S1 or hung S0" state entered

11.3.2.2 Advanced TCO Mode

The PCH supports the Advanced TCO mode in which SMLink0 through SMlink4 are used in addition to the host SMBus.

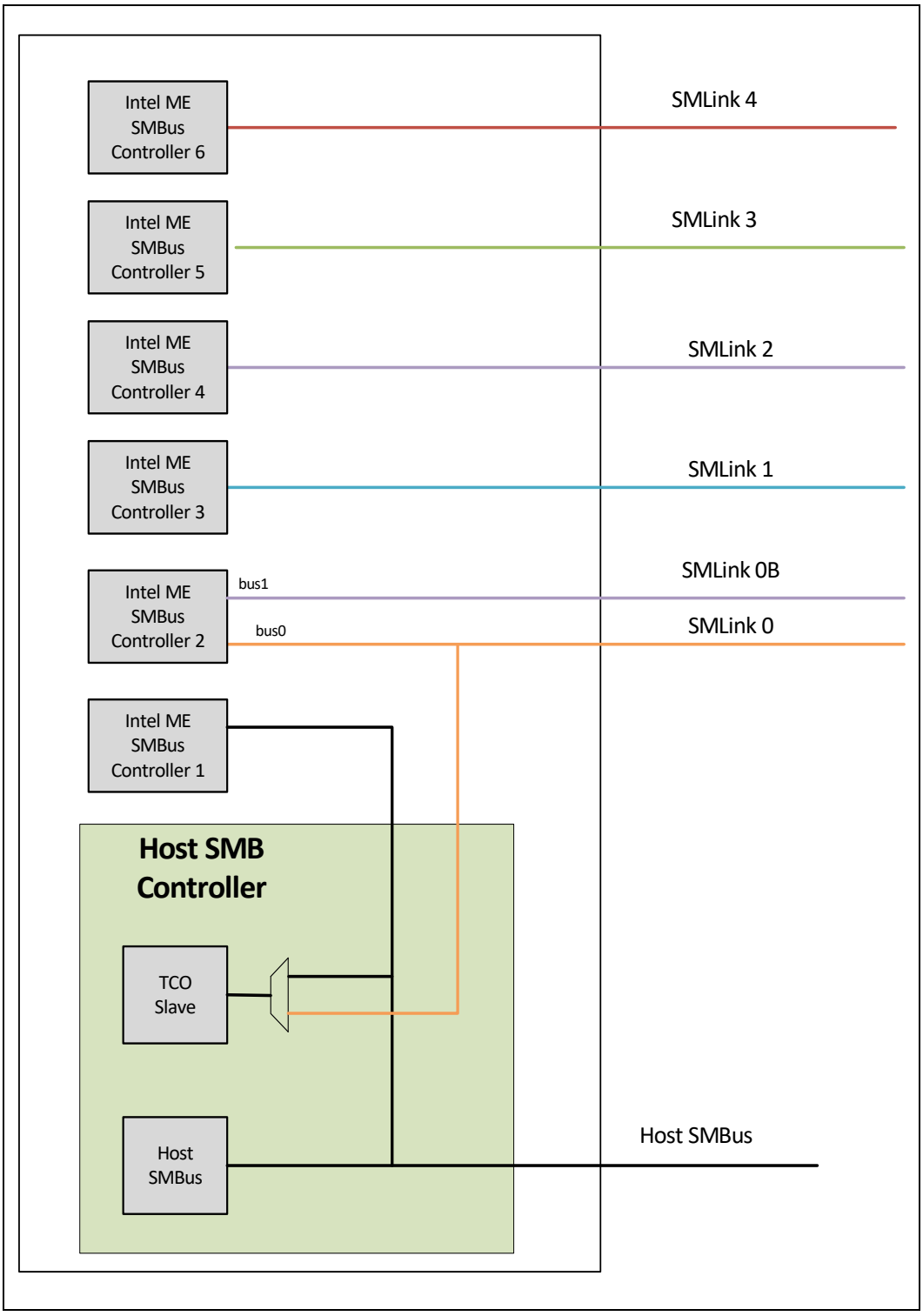
In this mode, the Intel ME SMBus controllers must be enabled by soft strap in the flash descriptor. See Figure 11-2 for more details.

In advanced TCO mode, the TCO slave can either be connected to the host SMBus or the SMLink0. See the latest SPI Flash Programming Guide for more detail.

SMLink0 is needed for the integrated single 1GbE LAN if Intel ME 11 FW is used. When an Intel LAN PHY is connected to SMLink0, a soft strap must be set to indicate that the PHY is connected to SMLink0. The interface will be running at the frequency of up to 1 MHz depending on different factors such as board routing or bus loading when the Fast Mode is enabled using a soft strap. See the latest SPI Flash Programming Guide for more detail. If System Platform Services FW is used, then SMLink0 is used to communicate with the BMC.

SMLink1 is connected to the PMBus for communication with power supplies with Server Platform Services FW. SMLink0b, SMLink2, SMLink3, and SMLink4 are free to be used however the board design chooses to.

Figure 11-2. Advanced TCO Mode



11.4 SMBus TCO I/O Registers Summary

The TCO I/O registers reside in a 32-byte range that starts from the I/O Base Address described in the TCOBAR register in the SMBus PCI Configuration space.

Table 11-2. Summary of SMBus TCO I/O Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	1h	TCO_RLD Register (TRLD)—Offset 0h	4h
2h	2h	TCO_DAT_IN Register (TDI)—Offset 2h	0h
3h	3h	TCO_DAT_OUT Register (TDO)—Offset 3h	0h
4h	5h	TCO1_STS Register (TSTS1)—Offset 4h	0h
6h	7h	TCO2_STS Register (TSTS2)—Offset 6h	0h
8h	9h	TCO1_CNT Register (TCTL1)—Offset 8h	0h
Ah	Bh	TCO2_CNT Register (TCTL2)—Offset Ah	8h
Ch	Dh	TCO_MESSAGE1 and TCO_MESSAGE2 (TMSG)—Offset Ch	0h
Eh	Eh	TCO_WDSTATUS Register (TWDS)—Offset Eh	0h
10h	10h	LEGACY_ELIM Register (LE)—Offset 10h	3h
12h	13h	TCO_TMR Register (TTMR)—Offset 12h	4h

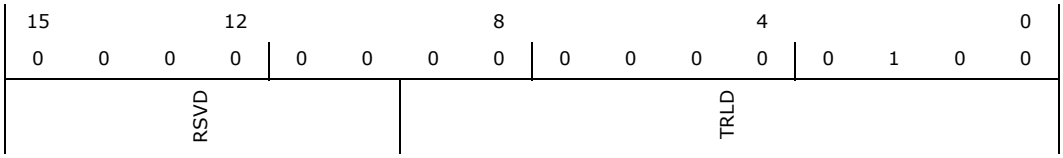
11.4.1 TCO_RLD Register (TRLD)—Offset 0h

Access Method

Type: I/O Register
(Size: 16 bits)

Device: 31
Function: 4

Default: 4h



Bit Range	Default and Access	Field Name (ID): Description
15:10	0h RO	Reserved
9:0	4h RW	**TCORLD (TRLD):** Reading this register will return the current count of the TCO timer. Writing any value to this register will reload the timer to prevent the timeout.

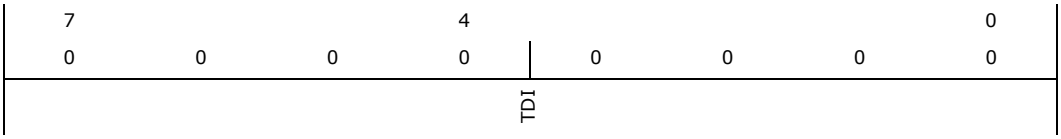
11.4.2 TCO_DAT_IN Register (TDI)—Offset 2h

Access Method

Type: I/O Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW	**TCO_DAT_IN (TDI):** Data Register for passing commands from the OS to the SMI handler. Writes to this register will cause an SMI and set the OS_TCO_SMI bit in the TCO_STS register.

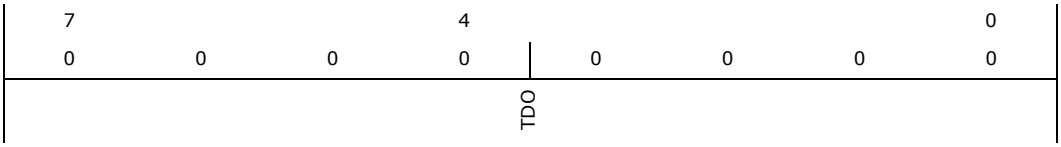
11.4.3 TCO_DAT_OUT Register (TDO)—Offset 3h

Access Method

Type: I/O Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW	**TCO_DAT_OUT (TDO):** Data Register for passing commands from the SMI handler to the OS. Writes to this register will set the TCO_INT_STS bit in the TCO_STS register. It also causes an interrupt, as selected by the TCO_IRQ_SEL bits.

11.4.4 TCO1_STS Register (TSTS1)—Offset 4h

Unless otherwise indicated, these bits are sticky and are cleared by writing a 1 to the corresponding bit position.

Access Method

Type: I/O Register
(Size: 16 bits)

Device: 31
Function: 4

Default: 0h

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD		TCO_SLVSEL	CPUSERR_STS	RSVD	CPUSMI_STS	CPUSCI_STS	BIOSWR_STS	NEWCENTURY_STS	RSVD			TIMEOUT	TCO_INT_STS	OS_TCO_SMI	NMI2SMI_STS

Bit Range	Default and Access	Field Name (ID): Description
15:14	0h RO	Reserved
13	0h RO/Strap	**TCO Slave Select (TCO_SLVSEL):** This register bit indicates the value of TCO Slave Select Soft Strap.
12	0h RW/1C	**CPUSERR_STS (CPUSERR_STS):** This bit is set to 1 if the CPU complex sends a DMI special cycle message indicating that it wants to cause an SERR#. The software must read the MCH to find out why it wanted the SERR#. Software must write a 1 back to this bit to clear it.
11	0h RO	Reserved
10	0h RW/1C	**CPUSMI_STS (CPUSMI_STS):** This bit is set to 1 if the CPU complex sends a DMI special cycle message indicating that it wants to cause an SMI. The software must read the CPU to find out why it wanted the SMI. Software must write a 1 back to this bit to clear it.
9	0h RW/1C	**CPUSCI_STS:** This bit is set to 1 if the CPU complex sends a DMI special cycle message indicating that it wants to cause an SCI. The software must read the CPU to find out why it wanted the SCI. Software must write a 1 back to this bit to clear it.
8	0h RW/1C	**BIOSWR_STS:** Intel PCH sets this bit to 1 and generates an SMI# to indicate an illegal attempt to write to the BIOS located in the FWH that is accessed over the LPC. This occurs when either: a) The BIOSWP bit is changed from 0 to 1 and the LE bit is also set, or b) Any write is attempted to the BIOS and the BIOSWP bit is also set. This bit does not get set to 1 when: 1) a or b above occurs on eSPI controller. 2) a or b above occurs on SPI Flash controller. ***Note:*** On write cycles attempted to the 4 MB lower alias to the BIOS space, the BIOSWR_STS bit will not be set.

Bit Range	Default and Access	Field Name (ID): Description
7	0h RW/1C	**NEWCENTURY_STS (NEWCENTURY_STS):** This bit will be set when the year rolls over from 1999 to 2000. If the bit is already 1, it will remain 1. This bit can be cleared either by software writing a 1 back to the bit position, or by RTCRST# going active. When this bit is set, an SMI# will be generated. However, this will not be a wake event (i.e., if the system is in a sleeping state when the NEWCENTURY_STS bit is set, the system will not wake up). ***Note:*** This bit 7 is not valid when the RTC battery is first put in (or if the RTC battery does not provide sufficient power when the system is unplugged). Software can determine that the RTC well was not maintained by checking the RTC_PWR_STS bit (GEN_PMCON_3 register in the Power Management Controller, D31:F2:A4, bit 2) or by other means (such as doing a checksum on the RTC RAM array). If the RTC well is determined to not have been maintained, the BIOS should set the time to a legal value and then clear the NEWCENTURY_STS bit. ***Note:*** This bit may take up to 3 RTCCLKs for the bit to be cleared when a 1 is written to the bit to clear it. After writing a 1 to the NEWCENTURY_STS bit, software should also not exit the SMI handler until after the bit has been cleared. This is to make sure the SMI is not re-entered. BIOS Assumption: When booting, the BIOS checks the NEWCENTURY_STS bit. If set, the BIOS should increment the value in the RTC RAM register associated with the century. The BIOS should then clear the NEWCENTURY_STS bit. This scenario would occur if the system was asleep when the century rolls over. If the system is in an S0 state (not sleeping) and the SMI# occurs with the NEWCENTURY_STS bit sets, the SMI handler should increment the value in the RTC RAM register and clear the NEWCENTURY_STS bit.
6:4	0h RO	Reserved
3	0h RW/1C	**TIMEOUT (TIMEOUT):** Bit set to 1 by Intel PCH to indicate that the SMI was caused by TCO timer reaching 0. ***Note:*** The SMI handler should clear this bit to prevent an immediate re-entry to the SMI handler.
2	0h RW/1C	**TCO_INT_STS (TCO_INT_STS):** Bit set to 1 when SMI handler caused the interrupt by writing to the TCO_DAT_OUT register.
1	0h RW/1C	**OS_TCO_SMI:** Bit set to 1 when OS code caused an SMI# by writing to the TCO_DAT_IN register.
0	0h RO/V	**NMI2SMI_STS:** The PCH sets this bit when an SMI# occurs because an event occurred that would otherwise have caused an NMI.

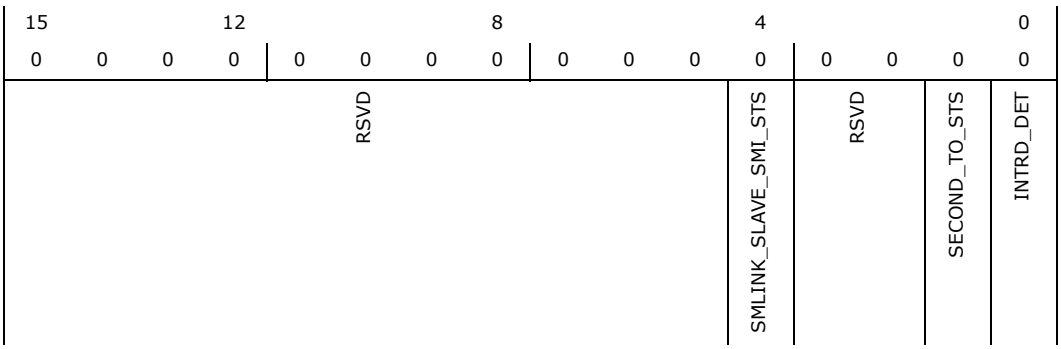
11.4.5 TCO2_STS Register (TSTS2)—Offset 6h

Access Method

Type: I/O Register
(Size: 16 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:5	0h RO	Reserved
4	0h RW/1C	**SMLINK_SLAVE_SMI_STS:** The PCH will set this bit to 1 when it receives the SMI message (encoding 08h in the command type) on the SMLinks Slave Interface. Software clears the bit by writing a 1 to this bit position. This bit is in the resume well. It is reset by RSMRST#, but not by the PCI Reset associated with exit from S3-S5 states. This allows the software (presumably BIOS) to get the interrupt, see this new bit set, and decidedly go into the pre-determined (by local policy) sleep state.
3:2	0h RO	Reserved
1	0h RW/1C	**SECOND_TO_STS:** Intel PCH sets this bit to 1 to indicate that the TIMEOUT bit had been (or is currently) set and a second timeout occurred before the TCO_RLD register was written. If this bit is set and the NO_REBOOT config bit is 0, then the Intel PCH will reboot the system after the second timeout. The reboot is done by asserting PLTRST#. This bit is only cleared by writing a 1 to this bit or by a RSMRST#.
0	0h RW/1C	**INTRD_DET (INTRD_DET):** The bit is set to 1 by the PCH to indicate that an intrusion was detected. This bit is cleared by writing a 1 to this bit or by RTCRST#. ***Note:*** This bit has a recovery time. After writing a 1 to this bit position (to clear it), the bit may be read back as a 1 for up 65 microseconds before it is read as a 0. Software must be aware of this recovery time when reading this bit after clearing it. ***Note:*** If the INTRUDER# signal is active when the software attempts to clear the INTRD_DET bit, the bit will remain as a 1, and the SMI# will be generated again immediately. The SMI handler can clear the INTRD_SEL bits to avoid further SMIs. However, if the INTRUDER# signals goes inactive and then active again, there will not be further SMIs. If the INTRUDER# signal goes inactive some point after the INTRD_DET bit is written as a 1, then the INTRD_DET signal will go to a 0 when INTRUDER# input signal goes inactive. Note that this is slightly different than a classic sticky bit, since most sticky bits would remain active indefinitely when the signal goes active and would immediately go inactive when a 1 is written to the bit.

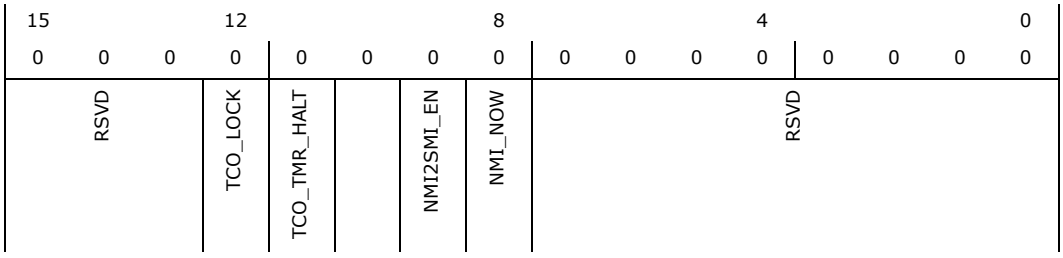
11.4.6 TCO1_CNT Register (TCTL1)—Offset 8h

Access Method

Type: I/O Register
(Size: 16 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:13	0h RO	Reserved
12	0h RW	**TCO_LOCK:** When set to 1, this bit prevents writes from changing the TCO_EN bit (in offset 30h of Power Management I/O space). Once this bit is set to 1, it can not be cleared by software writing a 0 to this bit location. A core-well reset is required to change this bit from 1 to 0. This bit defaults to 0.
11	0h RW	**TCO_TMR_HALT:** 1 = The TCO timer will halt. It will not count, and thus cannot reach a value that would cause an SMI# or to cause the SECOND_TO_STS bit to be set. This will also prevent rebooting. 0 = The TCO timer is enabled to count. This is the default.
10	0h RW	**Reserved**
9	0h RW	**NMI2SMI_EN:** Setting this bit 1 forces all NMIs to instead cause an SMI#, and will be reported in the TCO1_STS register. NMI2SMI_EN bit is set AND the NMI_EN# bit is set to 0, the NMI# will be routed to cause an SMI#. No NMI will be caused. However, if the GBL_SMI_EN bit is not set, then no SMI# will be generated, either. If NMI2SMI_EN is set but the NMI_EN# bit is set to 1, then no NMI or SMI# will be generated. The following table shows the possible combinations: NMI_EN#, GBL_SMI_EN 00: No SMI# based on NMI events (since no SMI# at all because SMI_EN = 0) 01: SMI# will be caused based on NMI events 10: No SMI# at all because SMI_EN is 0 11: No SMI# based on NMI events because NMI_EN#=1
8	0h RW	**NMI_NOW:** Writing a 1 to this bit causes an NMI. This allows the BIOS or SMI handler to force entry to the NMI handler. The NMI handler is expected to clear this bit. Another NMI will not be generated until the bit is cleared by writing a 1 back to the same bit position.
7:0	0h RO	Reserved

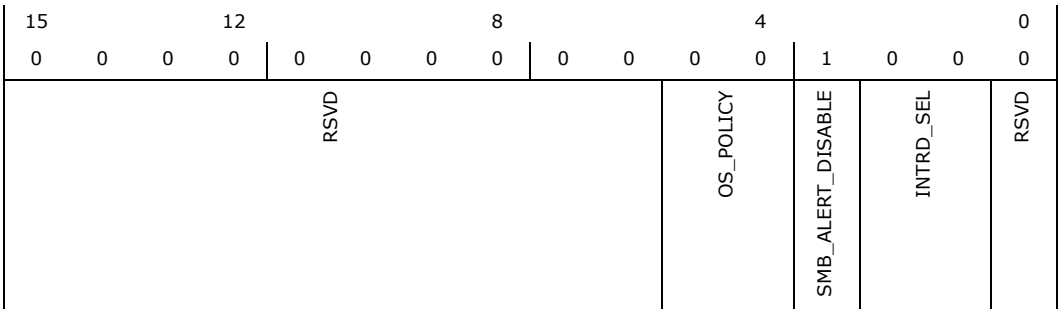
11.4.7 TCO2_CNT Register (TCTL2)—Offset Ah

Access Method

Type: I/O Register
(Size: 16 bits)

Device: 31
Function: 4

Default: 8h



Bit Range	Default and Access	Field Name (ID): Description
15:6	0h RO	Reserved.
5:4	0h RW	**OS_POLICY (OS_POLICY):** OS-based software writes to these bits to select the policy that the BIOS will use after the platform resets due the WDT. The following convention is recommended for the BIOS and OS: 00 Boot normally 01 Shut down 10 Do not load OS. Hold in pre-boot state and use LAN to determine next step 11 Reserved Implementation. ***Note:*** These are just scratch pad bits. They should not be reset when the TCO logic resets the platform due to Watchdog Timer.
3	1h RW	**SMB_ALERT_DISABLE:** Disables muxed GPIO/SMBALERT# signal as an alert source for the heartbeats and the SMBus slave. At reset (RSMRST# pin assertion only), this bit is set and the muxed GPIO/SMBALERT# alerts are disabled.
2:1	0h RW	**INTRD_SEL (INTRD_SEL):** Selects the action to take if the INTRUDER# signal goes active. 11: Reserved 01: Interrupt (as selected by TCO_INT_SEL). 10: SMI# 00 INTRUDER# doesn't cause SMI# or interrupt
0	0h RO	Reserved

11.4.8 TCO_MESSAGE1 and TCO_MESSAGE2 (TMSG)—Offset Ch

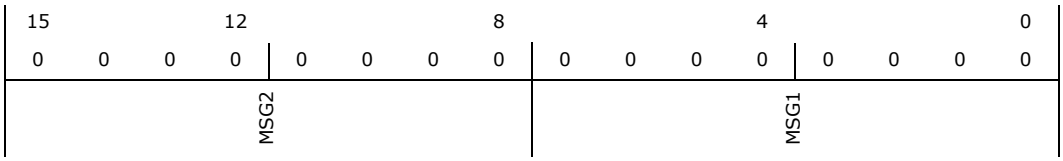
TCOBASE+0Ch (MSG1) TCOBASE+0Dh (MSG2) BIOS can write into these registers to indicate its boot progress. The external micro-controller can read these registers to monitor the boot progress.

Access Method

Type: I/O Register
(Size: 16 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:8	0h RW	**TCO_MESSAGE2 (MSG2)**
7:0	0h RW	**TCO_MESSAGE1 (MSG1)**

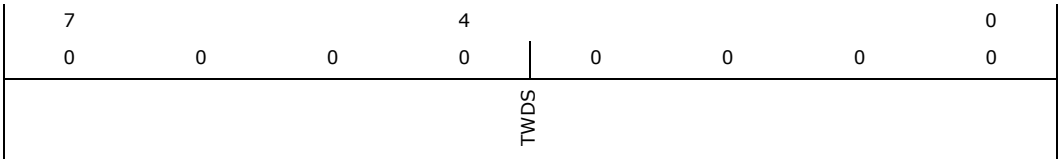
11.4.9 TCO_WDSTATUS Register (TWDS)—Offset Eh

Access Method

Type: I/O Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW	**TCO_WDSTATUS Register (TWDS):** The BIOS or system management software can write into this register to indicate more details on the boot progress. The register will rest to 00h based on a RSMRST# (but not PCI Reset). The external micro-controller can read this register to monitor boot progress.

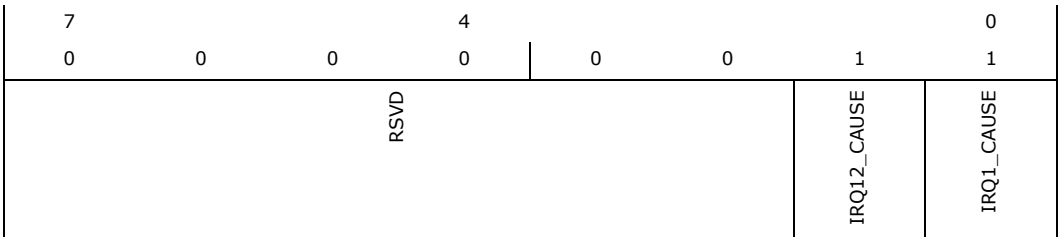
11.4.10 LEGACY_ELIM Register (LE)—Offset 10h

Access Method

Type: I/O Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 3h



Bit Range	Default and Access	Field Name (ID): Description
7:2	0h RO	Reserved.
1	1h RW	**IRQ12_CAUSE (IRQ12_CAUSE):** When software sets the bit to 1, IRQ12 will be high (asserted). When software sets the bit to 0, IRQ12 will be low (not asserted). Default for this bit is 1.
0	1h RW	**IRQ1_CAUSE (IRQ1_CAUSE):** When software sets the bit to 1, IRQ1 will be high (asserted). When software sets the bit to 0, IRQ1 will be low (not asserted). Default for this bit is 1.

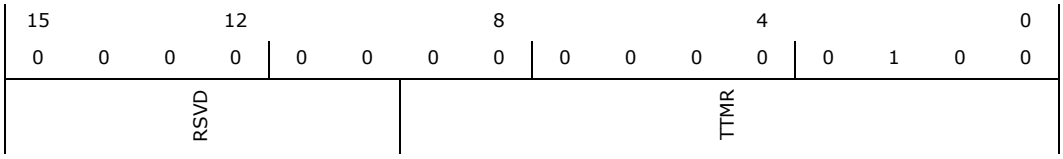
11.4.11 TCO_TMR Register (TTMR)—Offset 12h

Access Method

Type: I/O Register
(Size: 16 bits)

Device: 31
Function: 4

Default: 4h



Bit Range	Default and Access	Field Name (ID): Description
15:10	0h RO	Reserved.
9:0	4h RW	**TCOTMR (TTMR):** Value that is loaded into the timer each time the TCO_RLD register is written. Values of 0000h or 0001h will be ignored and should not be attempted. The timer is clocked at approximately 0.6 seconds, and thus allows timeouts ranging from 1.2 second to 613.8 seconds. ***Note:*** The timer has an error of ± 1 tick (0.6s).

§

12 High Precision Event Timer

12.1 References

Specification	Location
IA-PC HPET (High Precision Event Timers) Specification, Revision 1.0a	http://www.intel.com/content/www/us/en/software-developers/software-developers-hpet-spec-1-0a.html

12.2 Overview

This function provides a set of timers that can be used by the operating system. The timers are defined such that the operating system may assign specific timers to be used directly by specific applications. Each timer can be configured to cause a separate interrupt.

The PCH provides eight timers. The timers are implemented as a single counter, and each timer has its own comparator and value register. The counter increases monotonically. Each individual timer can generate an interrupt when the value in its value register matches the value in the main counter.

Timer 0 supports periodic interrupts.

The registers associated with these timers are mapped to a range in memory space (much like the I/O APIC). However, it is not implemented as a standard PCI function. The BIOS reports to the operating system the location of the register space via ACPI. The hardware can support an assignable decode space; however, the BIOS sets this space prior to handing it over to the operating system. It is not expected that the operating system will move the location of these timers once it is set by the BIOS.

12.2.1 Timer Accuracy

The timers are accurate over any 1 ms period to within 0.05% of the time specified in the timer resolution fields.

Within any 100-microsecond period, the timer reports a time that is up to two ticks too early or too late. Each tick is less than or equal to 100 ns; thus, this represents an error of less than 0.2%.

The timer is monotonic. It does not return the same value on two consecutive reads (unless the counter has rolled over and reached the same value).

The main counter uses the PCH's 24-MHz crystal as its clock. The accuracy of the main counter is as accurate as the crystal that is used in the system.

12.2.2 Timer Off-Load

The PCH supports a timer off-load feature that allows the HPET timers to remain operational during very low power S0 operational modes when the 24-MHz clock is disabled. The clock source during this off-load is the Real Time Clock's 32.768-kHz clock. This clock is calibrated against the 24-MHz clock during boot time to an accuracy that ensures the error introduced by this off-load is less than 10 ppb (.000001%).

When the 24-MHz clock is active, the 64-bit counter will increment by one each cycle of the 24-MHz clock when enabled. When the 24-MHz clock is disabled, the timer is maintained using the RTC clock. The long-term (> 1 msec) frequency drift allowed by the HPET specification is 500 ppm. The off-load mechanism ensures that it contributes < 1ppm to this, which will allow this specification to be easily met given the clock crystal accuracies required for other reasons.

Timer off-load is prevented when there are HPET comparators active.

The HPET timer in the PCH runs typically on the 24-MHz crystal clock and is off-loaded to the 32-kHz clock once the CPU enters C10. This is the state where there are no C10 wake events pending and when the off-load calibrator is not running. HPET timer re-uses this 28 bit calibration value calculated by PMC when counting on the 32-kHz clock. During C10 entry, PMC sends an indication to HPET to off-load and keeps the indication active as long as the CPU is in C10 on the 32-kHz clock. The HPET counter will be off-loaded to the 32-kHz clock domain to allow the 24-MHz clock to shut down when it has no active comparators.

12.2.3 Off-Loadable Timer

The Off-loadable Timer Block consists of a 64b fast clock counter and an 82b slow clock counter. During fast clock mode the counter increments by one on every rising edge of the fast clock. During slow clock mode, the 82 bit slow clock counter will increment by the value provided by the Off-load Calibrator.

The Off-loadable Timer will accept an input to tell it when to switch to the slow RTC clock mode and provide an indication of when it is using the slow clock mode. The switch will only take place on the slow clock rising edge, so for the 32-kHz RTC clock the max delay is around 30 microseconds to switch to or from slow clock mode. Both of these flags will be in the fast clock domain.

When transitioning from fast clock to slow clock, the fast clock value will be loaded into the upper 64b of the 82b counter, with the 18 LSBs set to zero. The actual transition through happens in two stages to avoid metastability. There is a fast clock sampling of the slow clock through a double flop synchronizer. Following a request to transition to the slow clock, the edge of the slow clock is detected and this causes the fast clock value to park. At this point the fast clock can be gated. On the next rising edge of the slow clock, the parked fast clock value (in the upper 64b of an 82b value) is added to the value from the Off-load Calibrator. On subsequent edges while in slow clock mode the slow clock counter increments its count by the value from the Off-load Calibrator.

When transitioning from slow clock to fast clock, the fast clock waits until it samples a rising edge of the slow clock through its synchronizer and then loads the upper 64b of the slow clock value as the fast count value. It then de-asserts the indication that slow clock mode is active. The 32-kHz clock counter no longer counts. The 64-bit MSB will be over-written when the 32-kHz counter is reloaded once conditions are met to enable the 32-kHz TSC counter but the 18-bit LSB is retained and it is not cleared out during the next reload cycle to avoid losing the fractional part of the counter.

After initiating a transition from fast clock to slow clock and parking the fast counter value, the fast counter no longer tracks. This means if a transition back to fast clock is requested before the entry into off-load slow clock mode completes, the Off-loadable Timer must wait until the next slow clock edge to restart. This case effectively performs the fast clock to slow clock and back to fast clock on the same slow clock edge.

12.2.4 Interrupt Mapping

The interrupts associated with the various timers have several interrupt mapping options. When reprogramming the HPET interrupt routing scheme (LEG_RT_CNF bit in the General Config Register), a spurious interrupt may occur. This is because the other source of the interrupt (8254 timer) may be asserted. Software should mask interrupts prior to clearing the LEG_RT_CNF bit.

12.2.4.1 Mapping Option #1 (Legacy Replacement Option)

In this case, the Legacy Replacement Rout bit (LEG_RT_CNF) is set. This forces the mapping found in Table 12-1.

Table 12-1. Legacy Replacement Routing

Timer	8259 Mapping	APIC Mapping	Comment
0	IRQ0	IRQ2	In this case, the 8254 timer will not cause any interrupts
1	IRQ8	IRQ8	In this case, the RTC will not cause any interrupts.
2 and 3	Per IRQ Routing Field.	Per IRQ Routing Field	
4, 5, 6, 7	not available	not available	
Note: The Legacy Option does not preclude delivery of IRQ0/IRQ8 using processor interrupts messages.			

12.2.4.2 Mapping Option #2 (Standard Option)

In this case, the Legacy Replacement Rout bit (LEG_RT_CNF) is 0. Each timer has its own routing control. The interrupts can be routed to various interrupts in the 8259 or I/O APIC. A capabilities field indicates which interrupts are valid options for routing. If a timer is set for edge-triggered mode, the timers should not be shared with any legacy interrupts.

For the PCH, the only supported interrupt values are as follows:

Timer 0 and 1: IRQ20, 21, 22, and 23 (I/O APIC only).

Timer 2: IRQ11 (8259 or I/O APIC) and IRQ20, 21, 22, and 23 (I/O APIC only).

Timer 3: IRQ12 (8259 or I/O APIC) and IRQ 20, 21, 22, and 23 (I/O APIC only).

Note: Interrupts from Timer 4, 5, 6, 7 can only be delivered using processor message interrupts.

12.2.4.3 Mapping Option #3 (Processor Message Option)

In this case, the interrupts are mapped directly to processor messages without going to the 8259 or I/O (x) APIC. To use this mode, the interrupt must be configured to edge-triggered mode. The Tn_PROCMSG_EN_CNF bit must be set to enable this mode.

When the interrupt is delivered to the processor, the message is delivered to the address indicated in the Tn_PROCMSG_INT_ADDR field. The data value for the write cycle is specified in the Tn_PROCMSG_INT_VAL field.

Note: The processor message interrupt delivery option has HIGHER priority and is mutually exclusive to the standard interrupt delivery option. Thus, if the Tn_PROCMSG_EN_CNF bit is set, the interrupts will be delivered directly to the processor, rather than by means of the APIC or 8259.

The processor message interrupt delivery can be used even when the legacy mapping is used.

12.2.5 Periodic Versus Non-Periodic Modes

12.2.5.1 Non-Periodic Mode

Timer 0 is configurable to 32- (default) or 64-bit mode, whereas timers 1:7 only support 32-bit mode.

Warning: Software must be careful when programming the comparator registers. If the value written to the register is not sufficiently far in the future, then the counter may pass the value before it reaches the register and the interrupt will be missed. The BIOS should pass a data structure to the operating system to indicate that the operating system should not attempt to program the periodic timer to a rate faster than 5 microseconds.

All of the timers support non-periodic mode.

Refer to Section 2.3.9.2.1 of the *IA-PC HPET Specification* for more details of this mode.

12.2.5.2 Periodic Mode

Timer 0 is the only timer that supports periodic mode. Refer to Section 2.3.9.2.2 of the *IA-PC HPET Specification* for more details of this mode.

If the software resets the main counter, the value in the comparator's value register needs to reset as well. This can be done by setting the TIMERn_VAL_SET_CNF bit. Again, to avoid race conditions, this should be done with the main counter halted. The following usage model is expected:

1. Software clears the ENABLE_CNF bit to prevent any interrupts.
2. Software Clears the main counter by writing a value of 00h to it.
3. Software sets the TIMER0_VAL_SET_CNF bit.
4. Software writes the new value in the TIMER0_COMPARATOR_VAL register.
5. Software sets the ENABLE_CNF bit to enable interrupts.

The Timer 0 Comparator Value register cannot be programmed reliably by a single 64-bit write in a 32-bit environment, except if only the periodic rate is being changed during run-time. If the actual Timer 0 Comparator Value needs to be reinitialized, then the following software solution will always work, regardless of the environment:

1. Set TIMER0_VAL_SET_CNF bit.
2. Set the lower 32 bits of the Timer0 Comparator Value register.
3. Set TIMER0_VAL_SET_CNF bit.
4. Set the upper 32 bits of the Timer0 Comparator Value register.

12.2.6 Enabling the Timers

The BIOS or operating system PnP code should route the interrupts. This includes the Legacy Rout bit, Interrupt Rout bit (for each timer), and interrupt type (to select the edge or level type for each timer).

The Device Driver code should do the following for an available timer:

1. Set the Overall Enable bit (Offset 10h, bit 0).

2. Set the timer type field (selects one-shot or periodic).
3. Set the interrupt enable.
4. Set the comparator value.

12.2.7 Interrupt Levels

Interrupts directed to the internal 8259s are active high. See Section 20.8, "Advanced Programmable Interrupt Controller (APIC) (D31:F0)" for information regarding the polarity programming of the I/O APIC for detecting internal interrupts.

If the interrupts are mapped to the 8259 or I/O APIC and set for level-triggered mode, they can be shared with legacy interrupts. They may be shared although it is unlikely for the operating system to attempt to do this.

If more than one timer is configured to share the same IRQ (using the TIMERn_INT_ROUT_CNF fields), then the software must configure the timers to level-triggered mode. Edge-triggered interrupts cannot be shared.

12.2.8 Handling Interrupts

Section 2.4.6 of the *IA-PC HPET Specification* describes handling interrupts.

12.2.9 Issues Related to 64-Bit Timers with 32-Bit Processors

Section 2.4.7 of the *IA-PC HPET Specification* describes issues related to 64-bit timers with 32-bit processors.

12.3 HPET Memory Mapped Registers Summary

The timer registers are memory mapped directly (rather than indexed) to allow the CPU to access each register without having to use an index register. This ensures accesses are safe for multi-threaded environments. The timer register space is 1024 bytes. The registers are generally aligned on 64-bit boundaries to simplify implementation with IA64 processors. In the PCH, there are four possible memory address ranges beginning at 1) FED0_0000h, 2) FED0_1000h, 3) FED0_2000h, 4) FED0_3000h. The choice of address range should be selected by assigning the High Performance Event Timer Configuration (HPTC) register fields in the configuration space of the Primary to Sideband Bridge. All registers are implemented in the Primary power well, and all bits are reset by PLTRST#. Reads to reserved registers or bits will return a value of 0. Behavorial Rules:

1. Software can read or write the various bytes in these registers using 32-bit or 64-bit accesses. Software must not attempt to read or write across register boundaries. For example, a 32-bit access should be to offset x0h, x4h, x8h, or xCh. 32-bit accesses should not be to 01h, 02h, 03h, 05h, 06h, 07h, 09h, 0Ah, 0Bh, 0Dh, 0Eh, or 0Fh. Any accesses to these offsets will result in an unexpected behavior, and may result in a master abort. However, these accesses should not result in system hangs. 64-bit accesses can only be to x0h and must not cross 64-bit boundaries.
2. Software should not write to read-only registers.
3. Software should not expect any particular or consistent value when reading reserved registers or bits.

Table 12-2. Summary of HPET Memory Mapped Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	7h	General Capabilities and ID Register (GEN_CAP_ID)—Offset 0h	27BC86B8086A701h
10h	17h	General Config Register (GEN_CFG)—Offset 10h	0h
20h	27h	General Interrupt Status Register (GEN_INT_STS)—Offset 20h	0h
F0h	F7h	Main Counter Value (MAIN_CNTR)—Offset F0h	0h
100h	107h	Timer n Config and Capabilities (TMRn_CNF_CAP)—Offset 100h	F0000000008030h
108h	10Fh	Timer n Comparator Value (TMRn_CMP_VAL)—Offset 108h	FFFFFFFFFFFFFFFFh

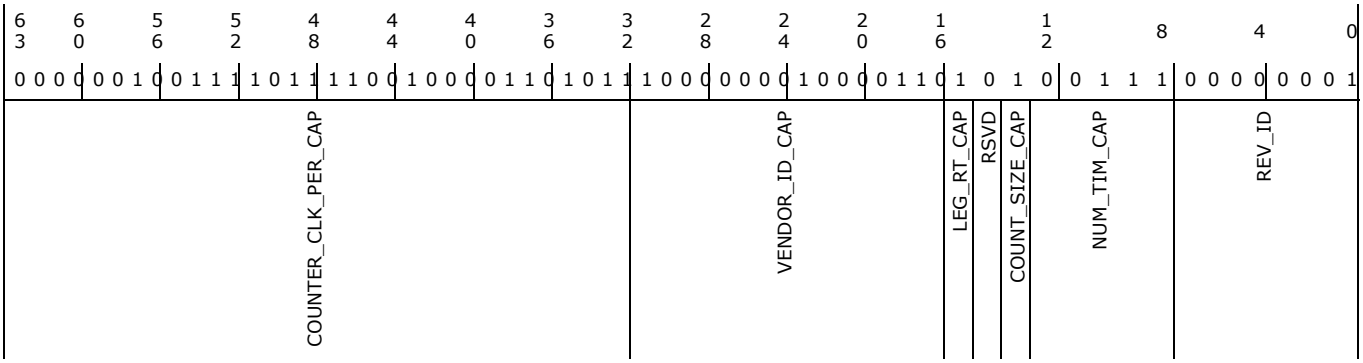
12.3.1 General Capabilities and ID Register (GEN_CAP_ID)—Offset 0h

Access Method

Type: MEM Register
(Size: 64 bits)

Device:
Function:

Default: 27BC86B8086A701h



Bit Range	Default and Access	Field Name (ID): Description
63:32	27BC86Bh RO	**Main Counter Tick Period (COUNTER_CLK_PER_CAP):** This read-only field indicates the period at which the counter increments in femtoseconds (10^-15 seconds). The PCH HPET timers use a 24 MHz clock, which has a period of 41,666,667 femtoseconds. Therefore this register will always return 027BC86Bh when read.
31:16	8086h RO	**Vendor ID (VENDOR_ID_CAP):** These bits will return 8086h when read to reflect Intel as the vendor.
15	1h RO	**Legacy Rout Capable (LEG_RT_CAP):** This bit will always be 1 when read, indicating support for the Legacy Interrupt Rout.
14	0h RO	Reserved

Bit Range	Default and Access	Field Name (ID): Description
13	1h RO	**Counter Size (COUNT_SIZE_CAP):** This bit will return 1 when read to indicate support for 64-bit counters allowing 64- or 32-bit mode operation.
12:8	7h RO	**Number of Timers (NUM_TIM_CAP):** This value in this field will be 07h to indicate support for 8 timers in the timer block.
7:0	1h RO	**Revision ID (REV_ID):** This field indicates which revision of the function is implemented. Default value will be 01h.

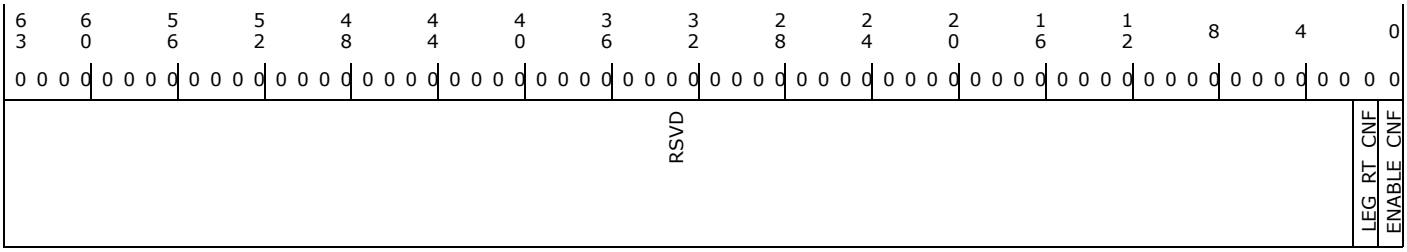
12.3.2 General Config Register (GEN_CFG)—Offset 10h

Access Method

Type: MEM Register
(Size: 64 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
63:2	0h RO	Reserved.
1	0h RW	**Legacy Rout (LEG_RT_CNF):** If the ENABLE_CNF bit and the LEG_RT_CNF bit are both set, the interrupts will be routed as follows: • Timer 0 is routed to IRQ0 in 8259 or IRQ2 in the I/O APIC • Timer 1 is routed to IRQ8 in 8259 or IRQ8 in the I/O APIC • Timer 2-n is routed as per the routing in the timer n Configuration registers. • If the Legacy Replacement Rout bit is set, the individual routing bits for Timers 0 and 1 (APIC) will have no impact. • If the Legacy Replacement Rout bit is not set, the individual routing bits for each of the timers are used. • This bit will default to 0. BIOS can set it to 1 to enable the legacy replacement routing, or 0 to disable the legacy replacement routing.
0	0h RW	**Overall Enable (ENABLE_CNF):** This bit must be set to enable any of the timers to generate interrupts. If this bit is 0, then the main counter will halt (will not increment) and no interrupts will be caused by any of these timers. For level-triggered interrupts, if an interrupt is pending when the ENABLE_CNF bit is changed from 1 to 0, the interrupt status indications (in the various Txx_INT_STS bits) will not be cleared. Software must write to the Txx_INT_STS bits to clear the interrupts. ***Note:*** This bit will default to 0. BIOS can set it to 1 or 0.

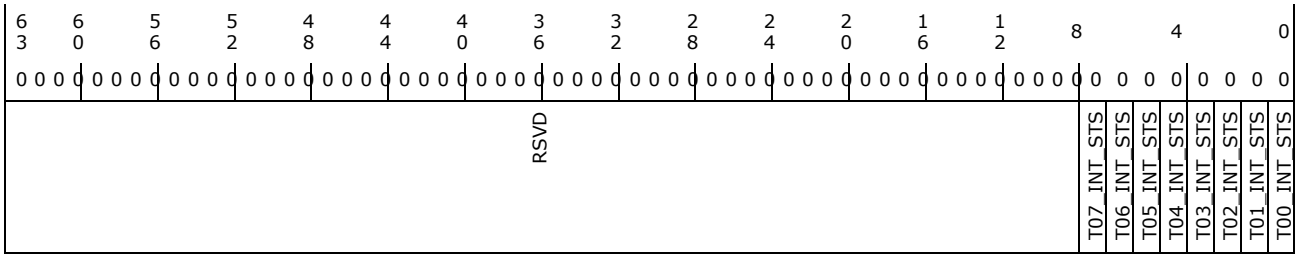
12.3.3 General Interrupt Status Register (GEN_INT_STS)—Offset 20h

Access Method

Type: MEM Register
(Size: 64 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
63:8	0h RO	Reserved
7	0h RW/C	**Timer 7 Interrupt Active (T07_INT_STS):** Same functionality as Timer 0.
6	0h RW/C	**Timer 6 Interrupt Active (T06_INT_STS):** Same functionality as Timer 0.
5	0h RW/C	**Timer 5 Interrupt Active (T05_INT_STS):** Same functionality as Timer 0.
4	0h RW/C	**Timer 4 Interrupt Active (T04_INT_STS):** Same functionality as Timer 0.
3	0h RW/C	**Timer 3 Interrupt Active (T03_INT_STS):** Same functionality as Timer 0.
2	0h RW/C	**Timer 2 Interrupt Active (T02_INT_STS):** Same functionality as Timer 0.
1	0h RW/C	**Timer 1 Interrupt Active (T01_INT_STS):** Same functionality as Timer 0.
0	0h RW/C	**Timer 0 Interrupt Active (T00_INT_STS):** The functionality of this bit depends on whether the edge or level-triggered mode is used for this timer. (default = 0) If set to level-triggered mode: This bit will be set by hardware if the corresponding timer interrupt is active. Once the bit is set, it can be cleared by software writing a 1 to the same bit position. Writes of 0 to this bit will have no effect. If set to edge-triggered mode: This bit should be ignored by software. Software should always write 0 to this bit. ***Note:*** Defaults to 0. In edge triggered mode, this bit will always read as 0 and writes will have no effect.

12.3.4 Main Counter Value (MAIN_CNTR)—Offset F0h

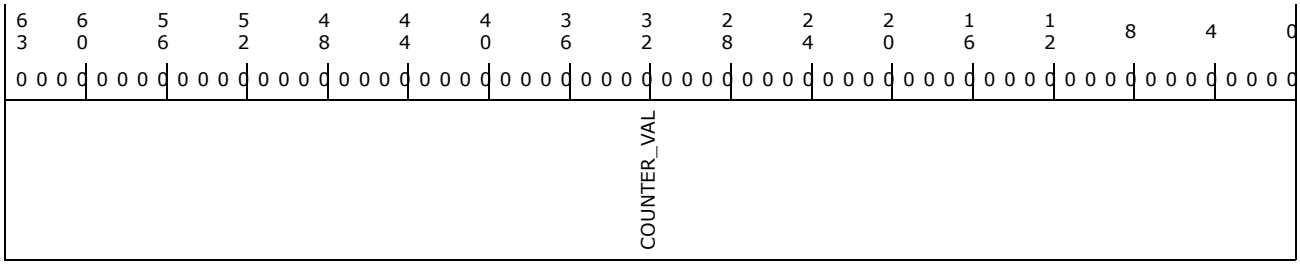
Software can read the various bytes in this register using 32- or 64-bit accesses. The 32-bit accesses may only be done to offset 0F0h or 0F4h. The 64-bit accesses may only be done to 0F0h. Writes to this register should only be done while the counter is halted. Reads to this register return the current value of the main counter. If 32-bit software attempts to read a 64-bit counter, it should first halt the counter. Since this will delay the interrupts for all of the timers, this should be done only if the consequences are understood. It is strongly recommended that 32-bit software only operate the timer in 32-bit mode. Reads to this register are monotonic. No two consecutive reads will return the same value. The second of two reads will always return a larger value (unless the timer has rolled over to 0)

Access Method

Type: MEM Register
(Size: 64 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
63:0	0h RW/V	**Counter Value (COUNTER_VAL):** Reads return the current value of the counter. Writes load the new value to the counter. ***Notes:*** 1. Writes to this register should only be done while the counter is halted. 2. Reads to this register return the current value of the main counter. 3. The 32-bit counters will always return 0 for the upper 32-bits of this register. 4. If 32-bit software attempts to read a 64-bit counter, it should first halt the counter. Since this will delay the interrupts for all of the timers, this should be done only if the consequences are understood. It is strongly recommended that 32-bit software only operate the timer in 32-bit mode. 5. Reads to this register are monotonic. No two consecutive reads will return the same value. The second of two reads will always return a larger value (unless the timer has rolled over to 0)

12.3.5 Timer n Config and Capabilities (TMRn_CNF_CAP)—Offset 100h

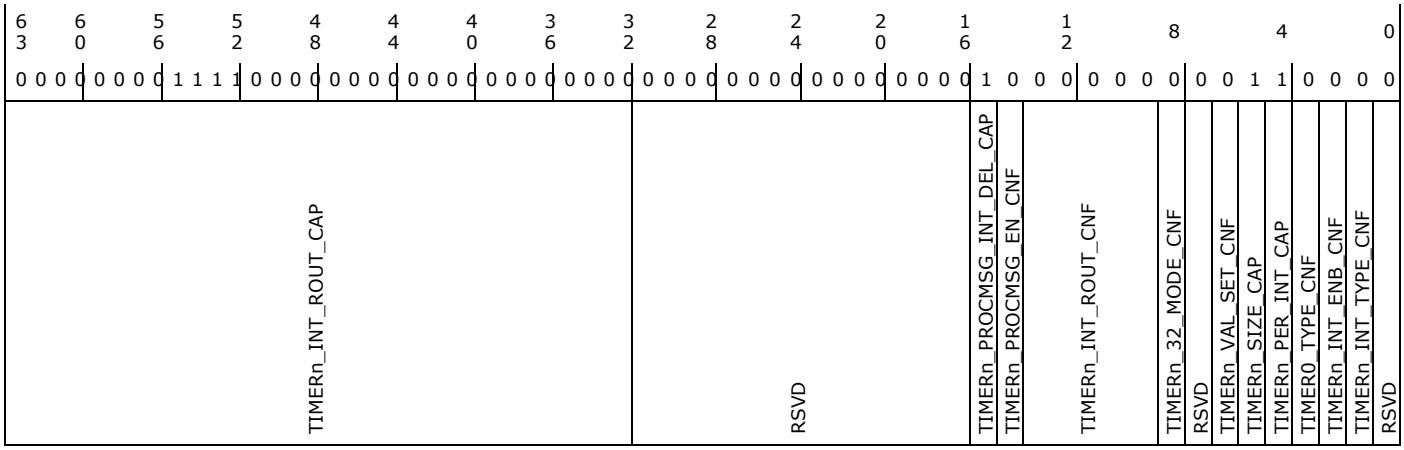
Timer 0: 100–107h,
Timer 1: 120–127h,
Timer 2: 140–147h,
Timer 3: 160–167h,
Timer 4: 180–187h,
Timer 5: 1A0–1A7h,
Timer 6: 1C0–1C7h,
Timer 7: 1E0–1E7h,
The letter n can be 0, 1, 2, 3, 4, 5, 6, or 7 referring to Timer 0, 1, 2, 3, 4, 5, 6, or 7.

Access Method

Type: MEM Register
(Size: 64 bits)

Device:
Function:

Default: F0000000008030h



Bit Range	Default and Access	Field Name (ID): Description
63:32	F00000h RO	**Timer n Interrupt Rout (TIMERn_INT_ROUT_CAP):** This 32-bit read-only field indicates to which interrupts in the 8259 or I/O (x) APIC this timers interrupt can be routed to. This is used in conjunction with the TIMERn_INT_ROUT_CNF field. Writes to this field will have no effect. ***Note:*** If interrupt is handled via 8259, only interrupts 0-15 are applicable and valid. Each bit in this field corresponds to a particular interrupt. For example, if this timers interrupt can be mapped to interrupts 16, 18, 20, 22, or 24, then bits 16, 18, 20, 22, and 24 in this field will be set to 1. All other bits will be 0. Timer 0,1 : Bits 20, 21, 22, and 23 in this field will have a value of 1. All other bits will be 0. Timer 2 : Bits 11, 20, 21, 22, and 23 in this field will have a value of 1. All other bits will be 0. If IRQ 11 is used, software should ensure IRQ 11 is not shared with any other devices to guarantee the proper operation of this timer. Timer 3 : Bits 12, 20, 21, 22, and 23 in this field will have a value of 1. All other bits will be 0. If IRQ 12 is used, software should ensure IRQ 12 is not shared with any other devices to guarantee the proper operation of this timer. Timer 4-7: This field is always 0 as interrupts from these timers can only be delivered via direct FSB interrupt messages.
31:16	0h RO	Reserved
15	1h RO	**Timer n Processor Message Interrupt Delivery Capability (TIMERn_PROCMSG_INT_DEL_CAP):** This bit is always read as 1, since the Intel PCH HPET implementation supports the direct processor interrupt delivery.
14	0h RW	**Timer n Processor Message Interrupt Delivery Enable (TIMERn_PROCMSG_EN_CNF):** When set, this will force the interrupts for Timer n to be delivered directly as processor messages, rather than using the 8259 or I/O (x) APIC. In this case, the TIMERn_INT_ROUTE_CNF field in this register will be ignored and the TIMERn_PROCMSG_ROUT register will be used instead. Timer 0, 1, 2, 3: This bit is a read/write bit. Timer 4, 5, 6, 7: This bit is always Read-Only 1 as interrupt from these timers canonly be delivered via direct FSB interrupt messages.
13:9	0h RW	**Timer n Interrupt Route (TIMERn_INT_ROUT_CNF):** This 5-bit field indicates the routing for the interrupt to the 8259 or I/O APIC. A maximum of 32 interrupts are supported. Software writes to this field to select which interrupt in the 8259 or I/O (x)APIC will be used for this timers interrupt. The default value for this register is 00h. If the Legacy Rout bit is set, then Timers 0 and 1 will have a different routing, and this bit field has no effect for those two timers. If the TIMERn_FSB_EN_CNF bit is set, then the interrupt will be delivered directly to the FSB, and this bit field has no effect. If interrupt is handled via 8259, only interrupts 0-15 are applicable and valid. Software must not program any value other than 0-15 in this field. Software must ensure that the value is valid for a particular timer as indicated by the TIMERn_INT_ROUTE_CAP field for that timer. The PCH logic does not check the validity of the value written. For Timers 4-7, this field is always Read-Only 0 as interrupts from these timers can only be delivered via direct FSB interrupt messages.
8	0h RW	**Timer n 32-bit Mode (TIMERn_32_MODE_CNF):** Software can set this bit to force a 64-bit timer to behave as a 32-bit timer. This is typically needed if the software is not willing to halt the main counter to read or write a particular timer, and the software not capable of do an atomic 64-bit read to the timer. When TIMER0_32MODE_CNF is set to '1', the hardware counter will essentially be doing 32-bit operation on comparator match and rollovers i.e., the upper 32-bit of the Timer 0 Comparator Value register is ignored. The upper 32-bit of the main counter is not involved in any roll over from lower 32-bit of the main counter and becomes all zero's. For timer 0, this bit will be read/write and default to 0. For timers 1-7, this bit will always read as 0 and writes will have no effect (since these seven timers are 32-bits).
7	0h RO	Reserved
6	0h RO	**Timer n Value Set (TIMERn_VAL_SET_CNF):** Software uses this bit only for timers that have been set to periodic mode. By writing this bit to a 1, the software is then allowed to directly set the timers accumulator. Software does not have to write this bit back to 0 (it automatically clears). This bit will return 0 when read.Software should not write a 1 to this bit position if the timer is set to non-periodic mode. Writes will only have an effect for Timer 0 if it is set to periodic mode. Writes will have no effect for Timers 1-7 as they do not support the periodic mode.
5	1h RO	**Timer n Size (TIMERn_SIZE_CAP):** Read-only Indicator of the timers size capability.1: 64-bits0: 32-bits.The value is 1 (64-bits) for timer 0, and 0 (32-bits) for timers 1-7.
4	1h RO	**Periodic Interrupt Capable (TIMERn_PER_INT_CAP):** If this read-only bit is 1, then the hardware supports a periodic mode for this timers interrupt. The value is 1 (periodic supported) for timer 0, and 0 (not supported) for timers 1-7.

Bit Range	Default and Access	Field Name (ID): Description
3	0h RW	**Timer 0 Type (TIMER0_TYPE_CNF):** Setting this bit to 1 enables the timer to generate a periodic interrupt if it is capable of doing so. If the TIMERn_PER_INT_CAP bit is 0, then this bit will always return 0 when read and writes will have no impact. For timer 0, this bit will be read/write, with default of 0. For timers 1-7, this bit will be read-only, with a fixed value of 0.
2	0h RW	**Timer n Interrupt Enable (TIMERn_INT_ENB_CNF):** Setting this bit to 1 enables the timer to generate a periodic interrupt if it is capable of doing so. If the TIMERn_PER_INT_CAP bit is 0, then this bit will always return 0 when read and writes will have no impact. For timer 0, this bit will be read/write, with default of 0. For timers 1-7, this bit will be read-only, with a fixed value of 0.
1	0h RW	**Timer Interrupt Type (TIMERn_INT_TYPE_CNF):** Determines whether an edge or level interrupt will be used for this timer (when enabled). 0: Edge-triggered. If another interrupt occurs, another edge will be generated. 1: Level-triggered. The interrupt will be held active until it is cleared by writing to the bit in the General Interrupt Status Register. If another interrupt occurs before the interrupt is cleared, the interrupt will remain active. The default value is 0, edge-triggered. The interrupt type for any timer should be set before that timer generates any interrupts. If the interrupt type is changed dynamically, there will be some delay before the new type takes effect. That delay is not specified. Timer 0-3: This bit is a read/write bit as both edge and level triggered modes are supported. Timer 4-7: This bit is always Read-Only 0 as only edge-triggered mode is supported.
0	0h RO	Reserved

12.3.6 Timer n Comparator Value (TMRn_CMP_VAL)—Offset 108h

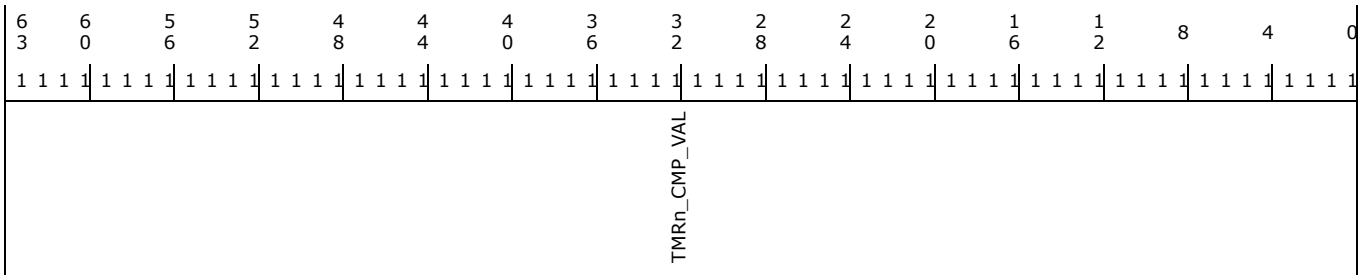
Timer 0: 108h – 10Fh
Timer 1: 128h – 12Fh
Timer 2: 148h – 14Fh
Timer 3: 168h – 16Fh
Timer 4: 188h – 18Fh
Timer 5: 1A8h – 1AFh
Timer 6: 1C8h – 1CFh
Timer 7: 1E8h – 1EFh

Access Method

Type: MEM Register
(Size: 64 bits)

Device:
Function:

Default: FFFFFFFFFFFFFFFFh



Bit Range	Default and Access	Field Name (ID): Description
63:0	FFFFFFFFF FFFFFFFh RW/V	**Timer n Comparator Value (TMRn_CMP_VAL):** If the timer is configured to non-periodic mode, when the main counter equals the value last written to this register, the corresponding interrupt can be generated (if so enabled). The value in this register does not change based on the interrupt being generated. If the timer is configured to periodic mode (supported only for Timer 0), when the main counter equals the value last written to this register, the corresponding interrupt can be generated (if so enabled). When this time out occurs, the value in this register is increased by the value last written to the register. For example, in periodic mode if the value written to the register is 0000123h: 1. An interrupt will be generated when the main counter reaches 00000123h. 2. The value in this register will then be adjusted by the hardware to 00000246h. 3. Another interrupt will be generated when the main counter reaches 00000246h. 4. The value in this register will then be adjusted by the hardware to 00000369h As each periodic interrupt occurs, the value in this register will increment. When the incremented value is greater than the maximum value possible for this register (FFFFFFFFh for a 32-bit timer or FFFFFFFFFFFFFFFFh for a 64-bit timer), the value will wrap around through 0. For example, if the current value in a 32-bit timer is FFFF0000h and the last value written to this register is 20000, then after the next interrupt the value will change to 00010000h The default value for each timer is all 1s for the bits that are implemented. For example, a 32-bit timer will have a default value of 00000000FFFFFFFFh. A 64-bit timer will have a default value of FFFFFFFFFFFFFFFFh. Software can read or write the various bytes in this register using 32-bit or 64-bit accesses. 32-bit accesses may only be done to offset 1x8h or 1xCh. 64-bit accesses may only be done to 1x8h. Comparator value. Timer 0 is 64-bits wide. Timers 1-7 are 32-bits wide.

§

13 Thermal Management

13.1 Acronyms

Acronyms	Description
DTS	Digital Thermal Sensor
$T_{accuracy}$	Thermal Accuracy
BMC	Baseboard Management Controller

13.2 PCH Thermal Sensor

The PCH incorporates an on-die digital thermal sensor for thermal management.

13.2.1 Modes of Operation

The DTS has two uses when enabled:

1. Provide the PCH temperature in units of 1/2 °C to the BMC. There is a 9-bit field for the temperature, with a range from -50 °C to 205.5 °C.
2. Allow programmed trip points to cause alerts via an interrupt (SCI, SMI, and INTx) or shut down the system (unconditionally transitions the system to S5) with a programmable catastrophic trip point.

13.2.2 Temperature Trip Point

The internal thermal sensor reports three trip points: cool, hot, and catastrophic trip points in the order of increasing temperature.

Crossing the cool trip point when going from higher to lower temperature may generate an interrupt. Crossing the hot trip point going from lower to higher temp may generate an interrupt. Each trip point has control register bits to select what type of interrupt is generated.

Crossing the cool trip point while going from low to higher temperature or crossing the hot trip point while going from high to lower temperature will not cause an interrupt.

When triggered, the catastrophic trip point will transition the system to S5 unconditionally.

In the Thermal Mechanical Design Guide, the standard references for these trip points have another name. In order to make sure the same references are used when comparing the datasheet to the TMDG, here is how the conversions go:

Catastrophic trip point = TEMP_HI

Thermal Alert High = TEMP_MID

Thermal Alert Low = $T_{CONTROL}$

13.2.3 Thermal Sensor Accuracy ($T_{accuracy}$)

$T_{accuracy}$ for the PCH is ±5 °C in the temperature range 50 – 110 °C, ±7 °C for temperatures from 30 °C – 50 °C and ±10 °C for temperatures from -10 – 30 °C. This value is based on product characterization and is not ensured by manufacturing test.

13.2.4 Thermal Reporting to a BMC

To support a platform BMC that is managing the system thermals, the PCH provides the ability for the BMC to read the PCH temperature over SMLink1 and over eSPI interface. The BMC will issue an SMBus read or eSPI OOB Channel request and receives a single byte of data, indicating a temperature between 1 °C and 254 °C, where 255 (0xFF) indicates that the sensor is not enabled yet.

Upon reset, the value driven to the BMC will be either 0xFF or 0x00.This indicates that BIOS has not enabled the reporting yet. When the BMC receives 0xFF or 0x00 for the temperature, it knows that the thermal sensor is not enabled and can assume that the system is in the boot phase with unknown temperature.

After the sensor is enabled, the BMC will receive a value between 0x01 and 0x7F (0 °C to 127 °C). If the BMC ever sees a value between 0x80 and 0xFE, that indicates an error has occurred, since the PCH should have shut down the platform before the temperature ever reached 128 °C (Catastrophic trip point will be below 128 °C). The PCH itself does not monitor the temperature and will not flag any error on the temperature value.

13.2.5 Thermal Trip Signal (PHOT#)

The PCH provides PHOT# signal to indicate that it has exceeded some temperature limit. The limit is set by BIOS. The temperature limit (programmed into the PHL register) is compared to the present temperature. If the present temperature is greater than the PHL value then the pin is asserted.

PHOT# is an O/D output and requires a pull-up on the motherboard.

The PCH evaluates the temperature from the thermal sensor against the programmed temperature limit every second.

13.2.6 Thermal Sensor Programming

Refer to the Intel® C620 Series Chipset BIOS specifications for recommendations and details.

13.3 Thermal Reporting Configuration Registers Summary

Table 13-1. Summary of Thermal Reporting Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	1h	Vendor Identification (VID)—Offset 0h	8086h
2h	3h	Device Identification (DID)—Offset 2h	31h
4h	5h	Command (CMD)—Offset 4h	0h
6h	7h	Status (STS)—Offset 6h	10h
8h	8h	Revision Identification (RID)—Offset 8h	0h
9h	9h	Programming Interface (PI)—Offset 9h	0h
Ah	Ah	Sub Class Code (SCC)—Offset Ah	80h
Bh	Bh	Base Class Code (BCC)—Offset Bh	11h
Ch	Ch	Cache Line Size (CLS)—Offset Ch	0h
Dh	Dh	Latency Timer (LT)—Offset Dh	0h
Eh	Eh	Header Type (HTYPE)—Offset Eh	0h
10h	13h	Thermal Base (TBAR)—Offset 10h	4h
14h	17h	Thermal Base High DWord (TBARH)—Offset 14h	0h
2Ch	2Dh	Subsystem Vendor ID (SVID)—Offset 2Ch	0h
2Eh	2Fh	Subsystem ID (SID)—Offset 2Eh	0h
34h	37h	Capabilities Pointer (CAP_PTR)—Offset 34h	50h
3Ch	3Ch	Interrupt Line (INTLN)—Offset 3Ch	0h
3Dh	3Dh	Interrupt Pin (INTPN)—Offset 3Dh	0h
40h	43h	BIOS Assigned Thermal Base Address (TBARB)—Offset 40h	4h
44h	47h	BIOS Assigned Thermal Base High DWord (TBARBH)—Offset 44h	0h
48h	48h	Control Bits (CB)—Offset 48h	0h
50h	51h	PCI Power Management Capability ID (PID)—Offset 50h	8001h
52h	53h	Power Management Capabilities (PC)—Offset 52h	23h
54h	57h	Power Management Control And Status (PCS)—Offset 54h	8h
80h	81h	Message Signaled Interrupt Identifiers (MID)—Offset 80h	5h
82h	83h	Message Signaled Interrupt Message Control (MC)—Offset 82h	0h
84h	87h	Message Signaled Interrupt Message Address (MA)—Offset 84h	0h
88h	8Bh	Message Signaled Interrupt Message Data (MD)—Offset 88h	0h

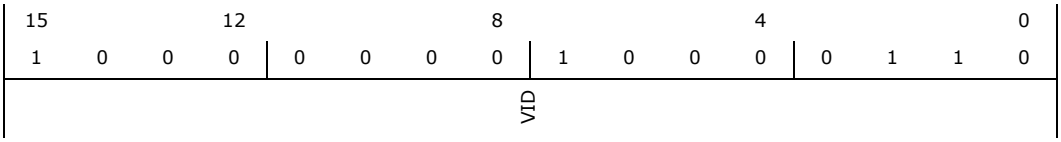
13.3.1 Vendor Identification (VID)—Offset 0h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 20
Function: 2

Default: 8086h



Bit Range	Default and Access	Field Name (ID): Description
15:0	8086h RO	**Vendor ID (VID):** Indicates that Intel is the vendor.

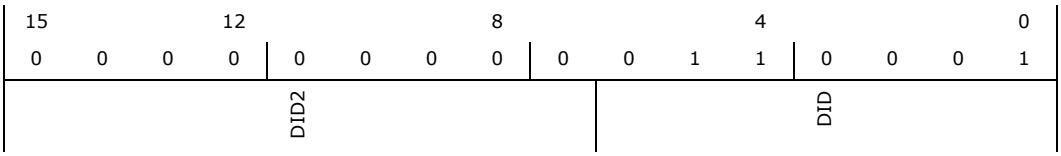
13.3.2 Device Identification (DID)—Offset 2h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 20
Function: 2

Default: 31h



Bit Range	Default and Access	Field Name (ID): Description
15:7	0h RO	**Device ID Part2 (DID2):** Indicates the device number
6:0	31h RO	**Device ID (DID):** Indicates the device number for Thermal controller

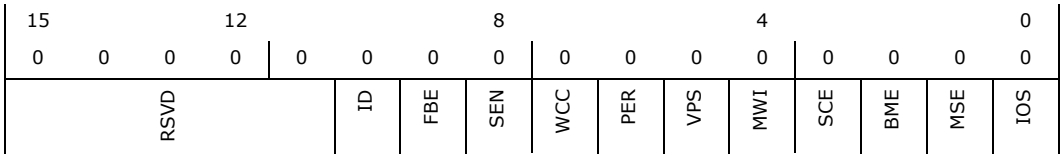
13.3.3 Command (CMD)—Offset 4h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:11	0h RO	Reserved
10	0h RW	**Interrupt Disable (ID):** Enables the device to assert an INTx#. When set, the Thermal logics INTx# signal will be de-asserted. When cleared AND MSI is not enabled, the INTx# signal may be asserted. ***Note:*** This bit has no affect on MSI generation.
9	0h RO	**Fast Back to Back Enable (FBE):** Not implemented. Hardwired to 0.
8	0h RW	**SERR Enable (SEN):** When set to 1 and an error occurs, SERR# is signaled to the system.
7	0h RO	**Wait Cycle Control (WCC):** Not implemented. Hardwired to 0.
6	0h RO	**Parity Error Response (PER):** Not implemented. Hardwired to 0.
5	0h RO	**VGA Palette Snoop (VPS):** Not implemented. Hardwired to 0.
4	0h RO	**Memory Write and Invalidate Enable (MWI):** Not implemented. Hardwired to 0.
3	0h RO	**Special Cycle Enable (SCE):** Not implemented. Hardwired to 0.
2	0h RW	**Bus Master Enable (BME):** When 1, enables
1	0h RW	**Memory Space Enable (MSE):** When set, enables memory space accesses to the Thermal registers.
0	0h RO	**I/O Space (IOS):** The Thermal logic does not implement I/O Space; therefore; this bit is hardwired to 0.

13.3.4 Status (STS)—Offset 6h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 20
Function: 2

Default: 10h

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
DPE	SERRS	RMA	RTA	STA	DEVT		MDPE	FBC	RSVD	C66	CLIST	IS	RSVD		

Bit Range	Default and Access	Field Name (ID): Description
15	0h RO	**Detected Parity Error (DPE):** This bit is set whenever a parity error is seen on the internal interface for this function, regardless of the setting of bit 6 in the command register. Software clears this bit by writing a '1' to this bit location. The thermal sensor unit never checks parity.
14	0h RW/1C	**SERR# Status (SERRS):** Not implemented. Hardwired to 0.
13	0h RO	**Received Master Abort (RMA):** Not implemented. Hardwired to 0.
12	0h RO	**Received Target Abort (RTA):** Not implemented. Hardwired to 0.
11	0h RW/1C	**Signaled Target-Abort (STA):** May be asserted on errors
10:9	0h RO	**DEVSEL# Timing Status (DEVT):** Does not apply. Hardwired to 0.
8	0h RO	**Master Data Parity Error (MDPE):** Not implemented. Hardwired to 0.
7	0h RO	**Fast Back to Back Capable (FBC):** Does not apply. Hardwired to 0.
6	0h RO	Reserved
5	0h RO	**66 MHz Capable (C66):** Does not apply. Hardwired to 0.
4	1h RO	**Capabilities List Exists (CLIST):** Indicates that the controller contains a capabilities pointer list. The first item is pointed to by looking at configuration offset 34h.
3	0h RO	**Interrupt Status (IS):** Reflects the state of the INTx# signal at the input of the enable/disable circuit. This bit is a 1 when the INTx# is asserted. This bit is a 0 after the interrupt is cleared (independent of the state of the Interrupt Disable bit in the command register). This bit is not set if MSI is enabled.
2:0	0h RO	Reserved.

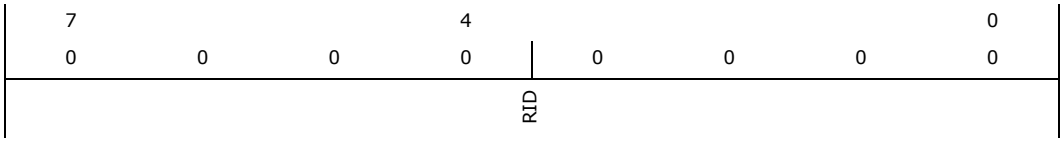
13.3.5 Revision Identification (RID)—Offset 8h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO/V	**Revision ID (RID):** Indicates the device specific revision identifier.

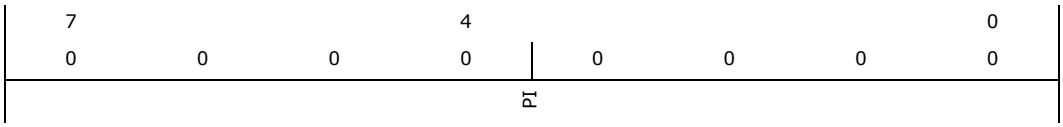
13.3.6 Programming Interface (PI)—Offset 9h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO	**Programming Interface (PI):** PCH Thermal logic has no standard programming interface.

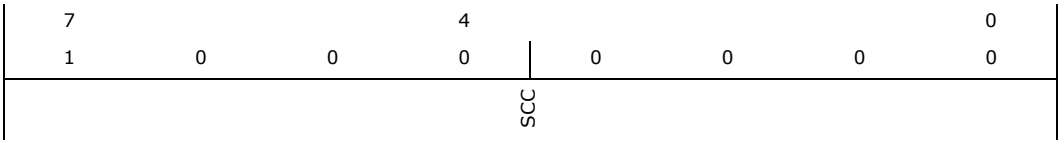
13.3.7 Sub Class Code (SCC)—Offset Ah

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 2

Default: 80h



Bit Range	Default and Access	Field Name (ID): Description
7:0	80h RO	**Sub Class Code (SCC):** Value assigned to PCH Thermal logic.

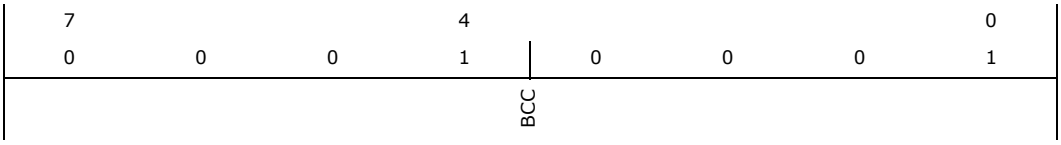
13.3.8 Base Class Code (BCC)—Offset Bh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 2

Default: 11h



Bit Range	Default and Access	Field Name (ID): Description
7:0	11h RO	**Base Class Code (BCC):** Value assigned to PCH Thermal logic.

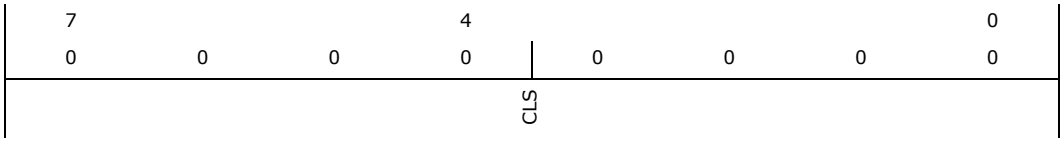
13.3.9 Cache Line Size (CLS)—Offset Ch

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO	**Cache Line Size (CLS):** Doesn't apply to PCI Bus Target-only devices.

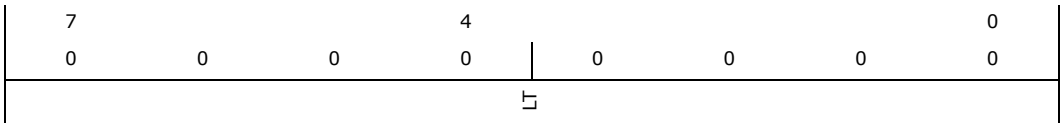
13.3.10 Latency Timer (LT)—Offset Dh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO	**Latency Timer (LT):** Doesn't apply to PCI Bus Target-only devices.

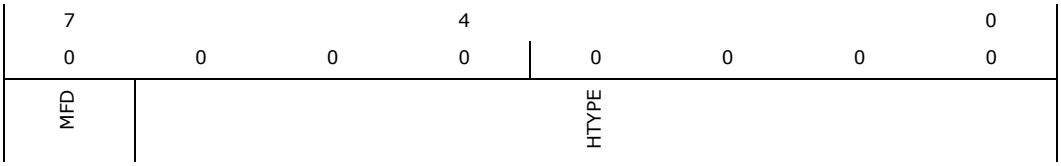
13.3.11 Header Type (HTYPE)—Offset Eh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7	0h RO	**Multi-Function Device (MFD):** This bit is '0' because a multi-function device only needs to be marked as such in Function 0, and the Thermal registers are not in Function 0.
6:0	0h RO	**Header Type (HTYPE):** Implements Type 0 Configuration header.

13.3.12 Thermal Base (TBAR)—Offset 10h

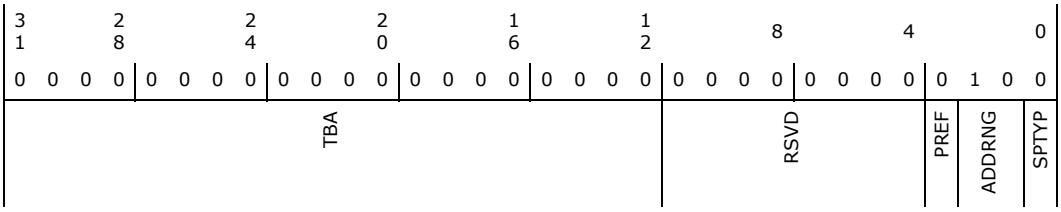
This BAR creates 4 KB of memory space to signify the base address of thermal memory mapped configuration registers. This memory space is active when the Command (CMD) register Memory Space Enable (MSE) bit is set and either TBAR[31:12] or TBARH are programmed to a non-zero address. This BAR is owned by the Operating System, and allows the OS to locate the Thermal registers in system memory space.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 20
Function: 2

Default: 4h



Bit Range	Default and Access	Field Name (ID): Description
31:12	0h RW	**Thermal Base Address (TBA):** Base address for the Thermal logic memory mapped configuration registers. 4 KB bytes are requested by hardwiring bits 11:4 to 0's.
11:4	0h RO	Reserved
3	0h RO	**Prefetchable (PREF):** Indicates that this BAR is NOT prefetchable.
2:1	2h RO	**Address Range (ADDRNG):** Indicates that this BAR can be located anywhere in 64-bit address space.
0	0h RO	**Space Type (SPTYP):** Indicates that this BAR is located in memory space.

13.3.13 Thermal Base High DWord (TBARH)—Offset 14h

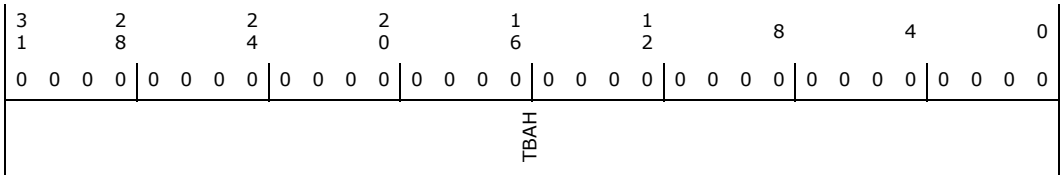
This BAR extension holds the high 32 bits of the 64-bit TBAR. In conjunction with TBAR, it creates 4 KB of memory space to signify the base address of Thermal memory mapped configuration registers.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Thermal Base Address High (TBAH):** TBAR bits [61:32].

13.3.14 Subsystem Vendor ID (SVID)—Offset 2Ch

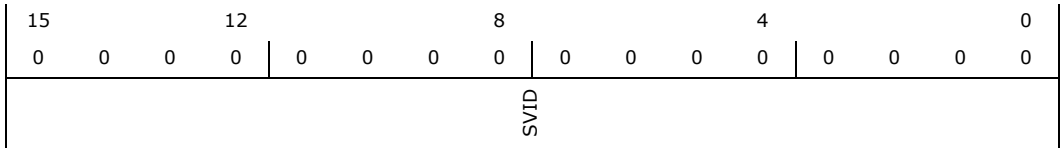
This register should be implemented for any function that could be instantiated more than once in a given system,. The SVID register, in combination with the Subsystem ID register, enables the operating environment to distinguish one subsystem from the other(s).
Software (BIOS) will write the value to this register. After that, the value can be read, but writes to the register will have no effect. The write to this register should be combined with the write to the SID to create one 32-bit write. This register is not affected by $D3_{HOT}$ to D0 reset.

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:0	0h RW/O	**SVID (SVID):** These RWO bits have no PCH functionality.

13.3.15 Subsystem ID (SID)—Offset 2Eh

This register should be implemented for any function that could be instantiated more than once in a given system. The SID register, in combination with the Subsystem Vendor ID register make it possible for the operating environment to distinguish one subsystem from the other(s).

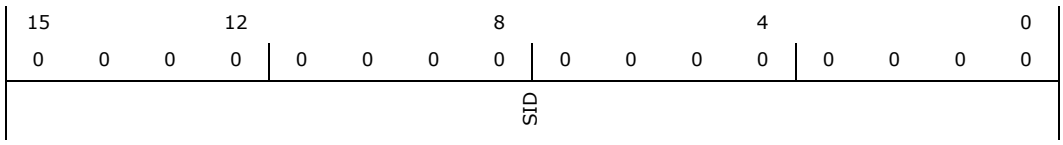
Software (BIOS) will write the value to this register. After that, the value can be read, but writes to the register will have no effect. The write to this register should be combined with the write to the SVID to create one 32-bit write. This register is not affected by $D3_{HOT}$ to D0 reset.

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:0	0h RW/O	**SID (SID):** These RWO bits have no PCH functionality.

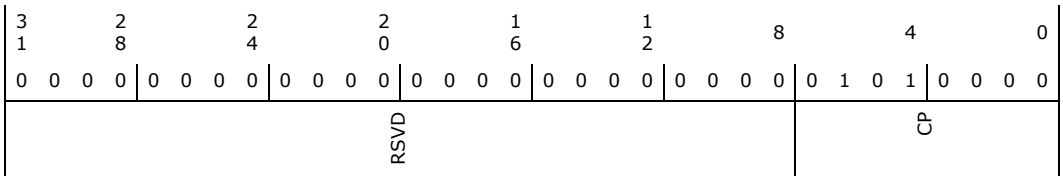
13.3.16 Capabilities Pointer (CAP_PTR)—Offset 34h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 20
Function: 2

Default: 50h



Bit Range	Default and Access	Field Name (ID): Description
31:8	0h RO	Reserved
7:0	50h RO	**Capability Pointer (CP):** Indicates that the first capability pointer offset is offset 50h (Power Management Capability).

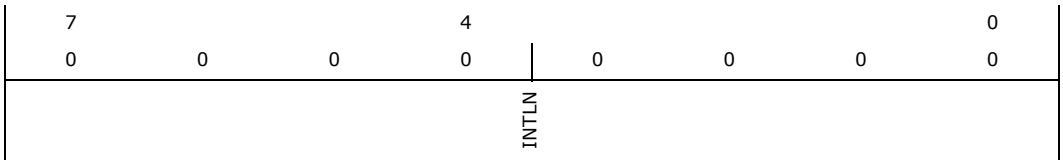
13.3.17 Interrupt Line (INTLN)—Offset 3Ch

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW	**Interrupt Line (INTLN):** PCH hardware does not use this field directly. It is used to communicate to software the interrupt line that the interrupt pin is connected to.

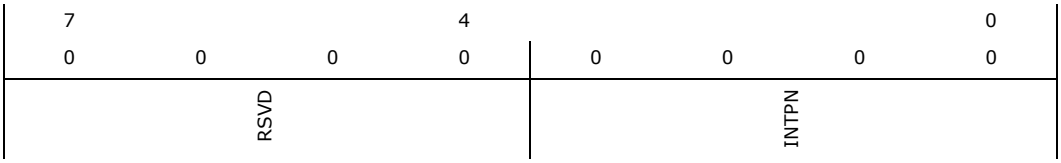
13.3.18 Interrupt Pin (INTPN)—Offset 3Dh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:4	0h RO	Reserved.
3:0	0h RW/O	**Interrupt Pin (INTPN):** This reflects the value of the interrupt pin used by this device.

13.3.19 BIOS Assigned Thermal Base Address (TBARB)—Offset 40h

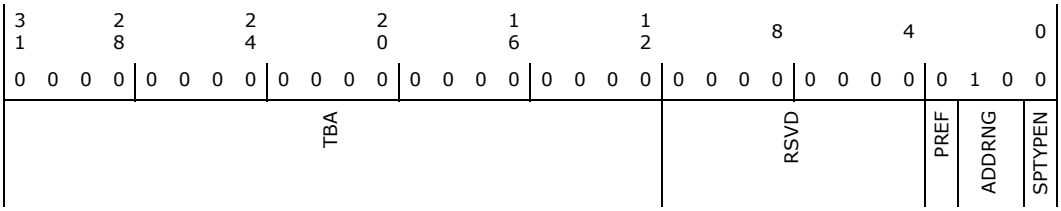
This BAR creates 4 KB of memory space to signify the base address of Thermal memory mapped configuration registers. This memory space is active when TBARB.SPTYPEN is asserted. This BAR is owned by the BIOS, and allows the BIOS to locate the Thermal registers in system memory space. If both TBAR and TBARB are programmed, then the OS and BIOS each have their own independent view of the Thermal registers, and must use the TSIU, TCIU, and TBIU registers to denote Thermal registers ownership/ availability.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 20
Function: 2

Default: 4h



Bit Range	Default and Access	Field Name (ID): Description
31:12	0h RW	**Thermal Base Address (TBA):** Base address for the Thermal logic memory mapped configuration registers. 4KB bytes are requested by hardwiring bits [11:4] to 0's.
11:4	0h RO	Reserved
3	0h RO	**Prefetchable (PREF):** Indicates that this BAR is NOT pre-fetchable.
2:1	2h RO	**Address Range (ADDRNG):** Indicates that this BAR can be located anywhere in 64 bit address space.
0	0h RW	**Space Type Enable (SPTYPEN):** When set to 1b by software, enables the decode of this memory BAR.

13.3.20 BIOS Assigned Thermal Base High DWord (TBARBH)—Offset 44h

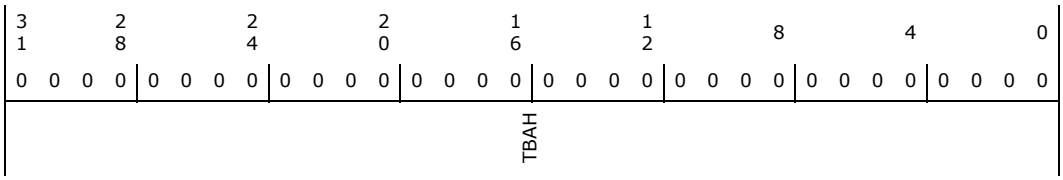
This BAR extension holds the high 32 bits of the 64-bit TBARB.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Thermal Base Address High (TBAH):** TBAR bits [61:32].

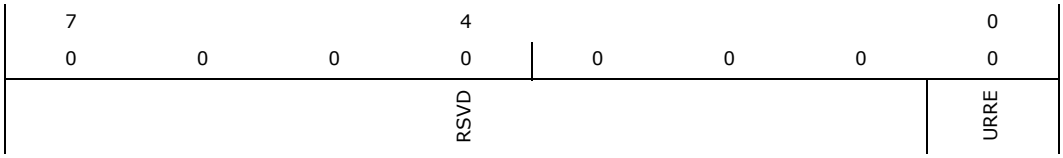
13.3.21 Control Bits (CB)—Offset 48h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:1	0h RO	Reserved
0	0h RW	**UR Reporting Enable (URRE):** When '1', the agent will set the URD bit. If SERR# enable (SEN) is set, then the agent will also send SERR# to the system. ***Note:*** Both URRE and SEN must be set to generate an SERR#.

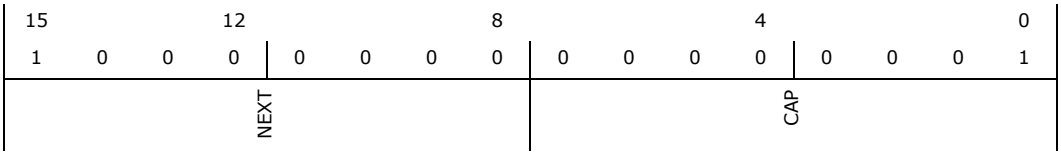
13.3.22 PCI Power Management Capability ID (PID)—Offset 50h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 20
Function: 2

Default: 8001h



Bit Range	Default and Access	Field Name (ID): Description
15:8	80h RO	**Next Capability (NEXT):** Indicates that the next capability is MSI.
7:0	1h RO	**Cap ID (CAP):** Indicates that this pointer is a PCI power management capability.

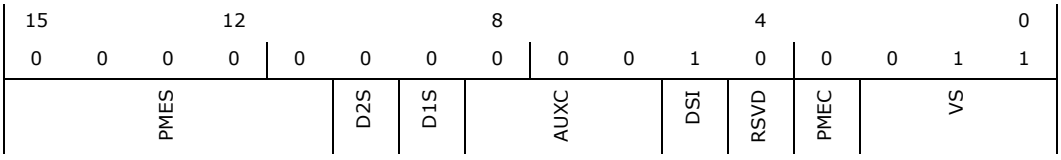
13.3.23 Power Management Capabilities (PC)—Offset 52h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 20
Function: 2

Default: 23h



Bit Range	Default and Access	Field Name (ID): Description
15:11	0h RO	**PME_Support (PMES):** Indicates PME# is not supported.
10	0h RO	**D2_Support (D2S):** The D2 state is not supported.
9	0h RO	**D1_Support (D1S):** The D1 state is not supported.
8:6	0h RO	**Aux_Current (AUXC):** PME# from $D3_{COLD}$ state is not supported, therefore this field is 000b.
5	1h RO	**Device Specific Initialization (DSI):** Indicates that device-specific initialization is required.
4	0h RO	Reserved
3	0h RO	**PME Clock (PMEC):** Does not apply. Hardwired to 0.
2:0	3h RO	**Version (VS):** Indicates support for Revision 1.2 of the PCI Power Management Specification.

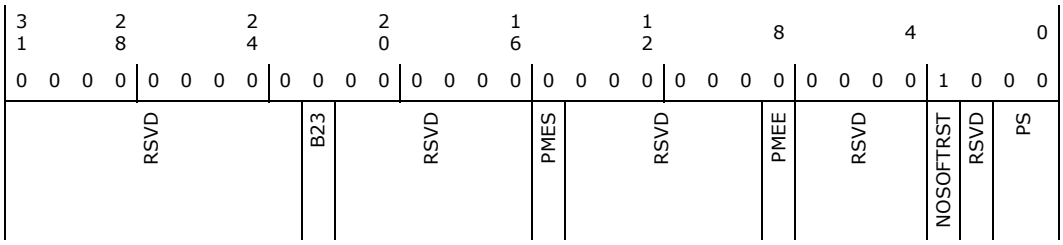
13.3.24 Power Management Control And Status (PCS)—Offset 54h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 20
Function: 2

Default: 8h



Bit Range	Default and Access	Field Name (ID): Description
31:23	0h RO	Reserved
22	0h RO	**B2/B3 Support (B23):** Does not apply. Hardwired to 0.
21:16	0h RO	Reserved
15	0h RO	**PME Status (PMES):** This bit is always zero, since this PCI Function does not generate PME#.
14:9	0h RO	Reserved

Bit Range	Default and Access	Field Name (ID): Description
8	0h RO	**PME Enable (PMEE):** This bit is always zero, since this PCI Function does not generate PME#.
7:4	0h RO	Reserved
3	1h RO	**No Soft Reset (NOSOFTRST):)**, this bit indicates that devices transitioning from $D3_{HOT}$ to D0 because of PowerState commands do not perform an internal reset. Configuration context is preserved. Upon transition from $D3_{HOT}$ to D0 initialized state, no additional operating system intervention is required to preserve Configuration Context belong writing the PowerState bits.
2	0h RO	Reserved
1:0	0h RW	**Power State (PS):** This field is used both to determine the current power state of the Thermal controller and to set a new power state. The values are: 00 = D0 state 11 = $D3_{HOT}$ state If software attempts to write a value of 10b or 01b in to this field, the write operation must complete normally, however, the data is discarded and no state change occurs. When in the $D3_{HOT}$ states, the Thermal controller's configuration space is available, but the I/O and memory spaces are not. Additionally, interrupts are blocked. When software changes this value from the $D3_{HOT}$ state to the D0 state, no internal warm (soft) reset is generated.

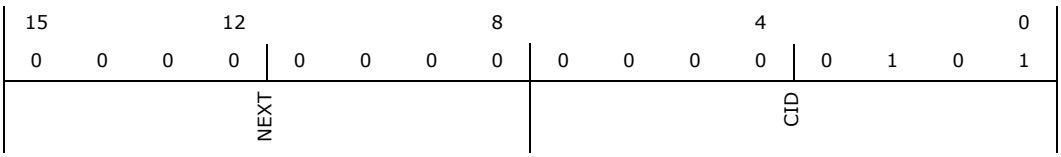
13.3.25 Message Signaled Interrupt Identifiers (MID)—Offset 80h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 20
Function: 2

Default: 5h



Bit Range	Default and Access	Field Name (ID): Description
15:8	0h RO	**Next Pointer (NEXT):** Indicates this is the latest pointer.
7:0	5h RO	**Capability ID (CID):** Capabilities ID indicates MSI.

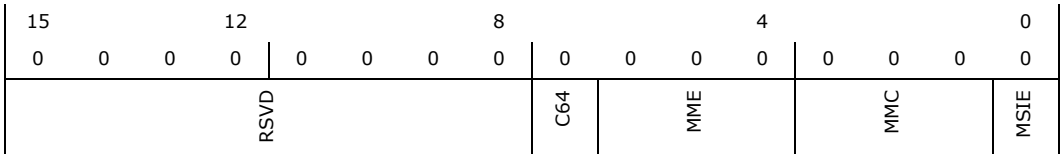
13.3.26 Message Signaled Interrupt Message Control (MC)—Offset 82h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:8	0h RO	Reserved
7	0h RO	**64 Bit Address Capable (C64):** Capable of generating a 32-bit message only.
6:4	0h RW	**Multiple Message Enable (MME):** These bits are RW for software compatibility, but only one message is ever sent by the root port.
3:1	0h RO	**Multiple Message Capable (MMC):** Only one message is required.
0	0h RW	**MSI Enable (MSIE):** If set, MSI is enabled and traditional interrupt pins are not used to generate interrupts. CMD.BME must be set for an MSI to be generated. If CMD.BME is cleared, and this bit is set, no interrupts (not even pin based) are generated.

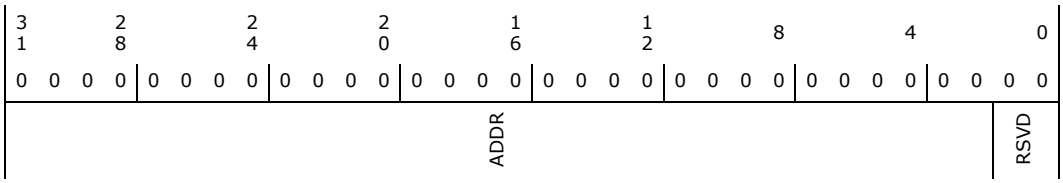
13.3.27 Message Signaled Interrupt Message Address (MA)—Offset 84h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:2	0h RW	**Address (ADDR):** Lower 32 bits of the system specified message address, always DW aligned.
1:0	0h RO	Reserved

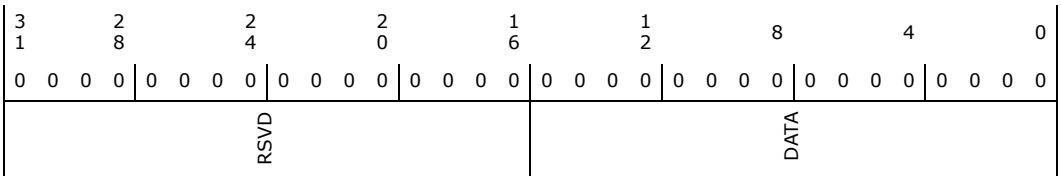
13.3.28 Message Signaled Interrupt Message Data (MD)—Offset 88h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved
15:0	0h RW	**Data (DATA):** This 16-bit field is programmed by system software if MSI is enabled. Its content is driven onto the lower word (PCI AD[15:0]) during the data phase of the MSI memory write transaction.

13.4 Thermal Reporting Memory Mapped Registers Summary

The Thermal Reporting Registers are located in the memory space mapped by TBAR (OS) and/or TBARB (BIOS), in the offset range from 0h to 0FFh.
All registers are reset by PLTRST#.

Table 13-2. Summary of Thermal Reporting Memory Mapped Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	1h	Temperature (TEMP)—Offset 0h	0h
4h	4h	Thermal Sensor Control (TSC)—Offset 4h	0h
6h	6h	Thermal Sensor Status (TSS)—Offset 6h	0h
8h	8h	Thermal Sensor Enable and Lock (TSEL)—Offset 8h	0h
Ah	Ah	Thermal Sensor Reporting Enable and Lock (TSREL)—Offset Ah	0h
Ch	Ch	Thermal Sensor SMI Control (TSMIC)—Offset Ch	0h
10h	11h	Catastrophic Trip Point (CTT)—Offset 10h	1FFh

Table 13-2. Summary of Thermal Reporting Memory Mapped Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
14h	15h	Thermal Alert High Value (TAHV)—Offset 14h	0h
18h	19h	Thermal Alert Low Value (TALV)—Offset 18h	0h
1Ch	1Dh	Thermal Sensor Power Management (TSPM)—Offset 1Ch	800h
40h	43h	Throttle Levels (TL)—Offset 40h	0h
50h	53h	Throttle Level 2 (TL2)—Offset 50h	0h
60h	61h	PCH Hot Level (PHL)—Offset 60h	0h
62h	62h	PHL Control (PHLC)—Offset 62h	0h
80h	80h	Thermal Alert Status (TAS)—Offset 80h	0h
82h	82h	PCI Interrupt Event Enables (TSPIEN)—Offset 82h	0h
84h	84h	General Purpose Event Enables (TSGPEN)—Offset 84h	0h
F0h	F0h	Thermal Controller Function Disable (TCFD)—Offset F0h	0h

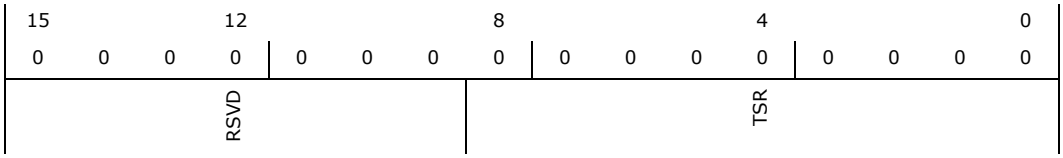
13.4.1 Temperature (TEMP)—Offset 0h

Access Method

Type: MEM Register
(Size: 16 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:9	0h RO	Reserved
8:0	0h RO	**TS Reading (TSR):** The die temperature with resolution of degree 0.5C and an offset of -50C. Thus a reading of 0x121 is 94.5C.

13.4.2 Thermal Sensor Control (TSC)—Offset 4h

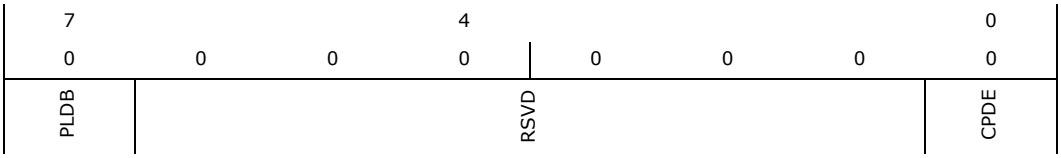
This register controls the operation of the thermal sensor.

Access Method

Type: MEM Register
(Size: 8 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7	0h RW/O/L	**Policy Lock-Down Bit (PLDB):** When written to 1, this bit prevents any more writes to this register (offset 04h) and to CTT (offset 0x10).
6:1	0h RO	Reserved.
0	0h RW/L	**Catastrophic Power-Down Enable (CPDE):** When set to 1, the power management logic (PMC) transitions to the S5 state when a catastrophic temperature is detected by the sensor. The transition to the S5 state must be unconditional (like the Power Button Override Function). Note that the thermal sensor and response logic is in the core/main power well, therefore, detection of a catastrophic temperature is limited to times when this well is powered and out of reset.

13.4.3 Thermal Sensor Status (TSS)—Offset 6h

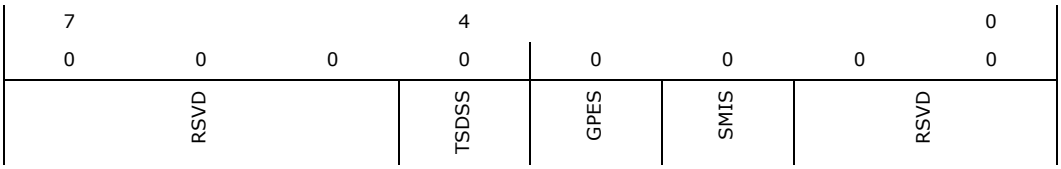
This read only register provides trip point and other status of the thermal sensor.

Access Method

Type: MEM Register
(Size: 8 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:5	0h RO	Reserved.
4	0h RO	**Thermal Sensor Dynamic Shutdown Status (TSDSS):** Thermal Sensor Dynamic Shutdown Status (TSDSS): This bit indicates the status of the thermal sensor circuit when TSEL.ETS=1. 1: Thermal sensor is fully operational. 0: Thermal sensor is in a dynamic shutdown state.
3	0h RW/1C	**GPE Status (GPES):** Set when GPE is enabled for a trip event. SW must write a 1 to this bit to clear the GPE status. Note that GPE can be configured to cause an SMI or SCI. As long as this bit is set, the GPE indication to the global GPE logic is asserted.
2	0h RW/1C	**SMI Status (SMIS):** Set when SMI is enabled for a trip event. SW must write a 1 to this bit to clear the SMI status. As long as this bit is set, the SMI indication to the global SMI logic is asserted.
1:0	0h RO	Reserved

13.4.4 Thermal Sensor Enable and Lock (TSEL)—Offset 8h

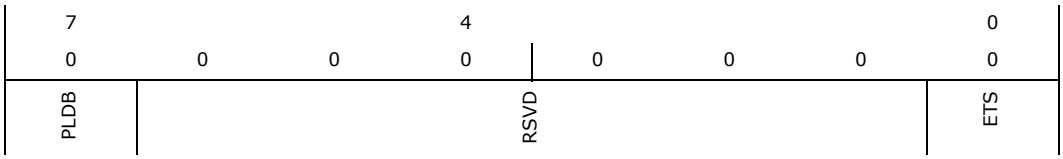
This register controls the operation of the thermal sensor.

Access Method

Type: MEM Register
(Size: 8 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7	0h RW/O/L	**Policy Lock-Down Bit (PLDB):** Policy Lock-Down Bit: When written to 1, this bit prevents any more writes to this register and to TTCB, Test1, Test2, Test3, Test4, Test5, Test6 and Test7 registers.
6:1	0h RO	Reserved
0	0h RW/L	**Enable TS (ETS):** 1: Enables the thermal sensor. Until this bit is set, no thermometer readings or trip events will occur. If SW reads the TEMP register before the sensor is enabled, it will read 0x0. The value of this bit is sent to the thermal sensor. ***Note:*** If the sensor is running and valid temperatures have been captured in TEMP and then ETS is cleared, TEMP will retain its old value. Clearing ETS does not force TEMP to 0x00. 0: Disables the sensor.

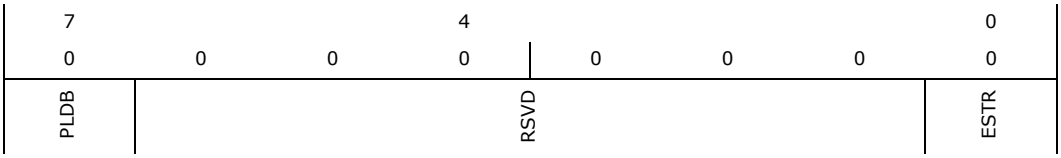
13.4.5 Thermal Sensor Reporting Enable and Lock (TSREL)—Offset Ah

Access Method

Type: MEM Register
(Size: 8 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7	0h RW/O/L	**Policy Lock-Down Bit (PLDB):** Policy Lock-Down Bit: When written to 1, this bit prevents any more writes to this register (offset 0Ah)
6:1	0h RO	Reserved
0	0h RW/L	**Enable SMBus Temperature Reporting (ESTR):** 1: Enables the reporting of the PCH temperature to the SMBus. Note that this must also be set if Intel ME needs access to the PCH temperature. Once enabled this bit should not be cleared by SW. If it is cleared then the EC may get an undefined value. SW has no need to dynamically disable and then re-enable this bit. 0: Disables EC temperature reporting.

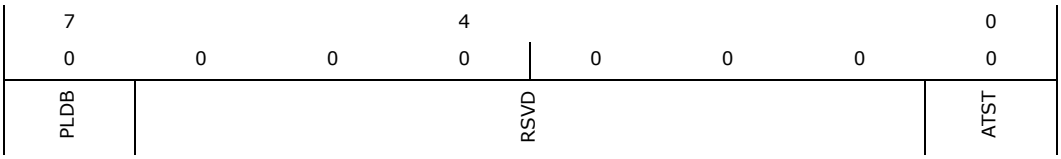
13.4.6 Thermal Sensor SMI Control (TSMIC)—Offset Ch

Access Method

Type: MEM Register
(Size: 8 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7	0h RW/O/L	**Policy Lock-Down Bit (PLDB):** When written to 1, this bit prevents any more writes to this register (offset 0Ch)
6:1	0h RO	Reserved
0	0h RW/L	**SMI Enable on Alert Thermal Sensor Trip (ATST):** 1: Enables SMI# assertions on alert thermal sensor events for either low-to-high or high-to-low events. (Both edges are enabled by this one bit.) 0: Disables SMI# assertions for alert thermal events.

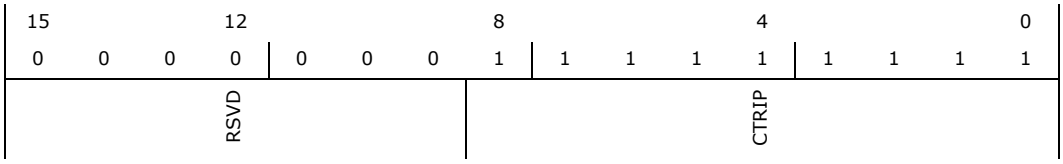
13.4.7 Catastrophic Trip Point (CTT)—Offset 10h

Access Method

Type: MEM Register
(Size: 16 bits)

Device: 20
Function: 2

Default: 1FFh



Bit Range	Default and Access	Field Name (ID): Description
15:9	0h RO	Reserved
8:0	1FFh RW/L	**Catastrophic Temperature TRIP (CTRIP):** When the current temperature reading is = to the value in this register, a catastrophic trip event is signaled. The value of this register must not be changed while TSEL.ETS is set. The value in this register is sent to the thermal sensor. This register is locked by TSC[7].

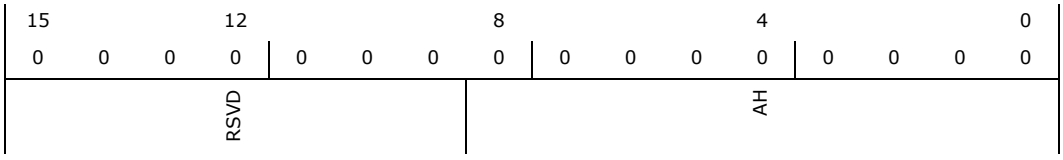
13.4.8 Thermal Alert High Value (TAHV)—Offset 14h

Access Method

Type: MEM Register
(Size: 16 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:9	0h RO	Reserved
8:0	0h RW	**Alert High (AH):** Alert High (AH) - Sets the high value for the alert indication. See the later section for usage. The value of this register must not be changed while TSEL.ETS is set. The value in this register is sent to the thermal sensor. This register is not lockable, so that SW can change the values during runtime.

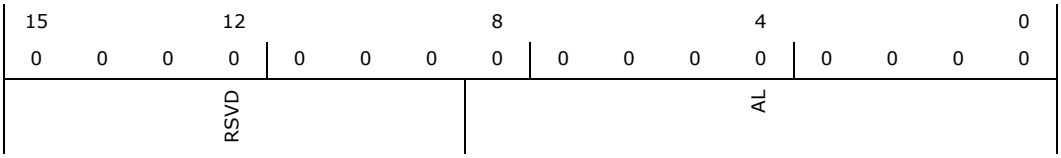
13.4.9 Thermal Alert Low Value (TALV)—Offset 18h

Access Method

Type: MEM Register
(Size: 16 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:9	0h RO	Reserved
8:0	0h RW	**Alert Low (AL):** Sets the low value for the alert indication. See the later section for usage. The value of this register must not be changed while TSEL.ETS is set. The value in this register is sent to the thermal sensor. This register is not lockable, so that SW can change the values during runtime.

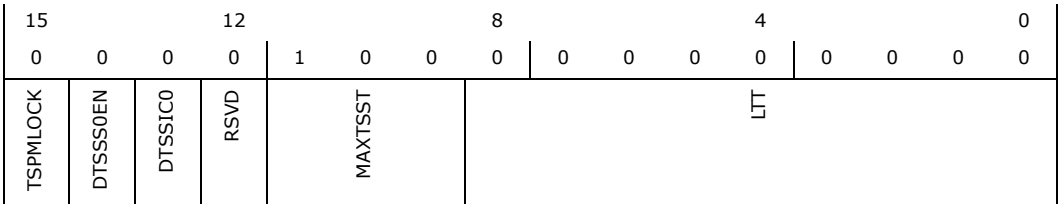
13.4.10 Thermal Sensor Power Management (TSPM)—Offset 1Ch

Access Method

Type: MEM Register
(Size: 16 bits)

Device: 20
Function: 2

Default: 800h



Bit Range	Default and Access	Field Name (ID): Description
15	0h RW/L	**Thermal Sensor Power Management Lock (TSPMLOCK):** Thermal Sensor Power Management Lock (TSPMLOCK): Setting this bit to a 1 causes the rest of the bits in this register to be locked.
14	0h RW/L	**Dynamic Thermal Sensor Shutdown in S0 Idle Enable (DTSSS0EN):** Dynamic Thermal Sensor Shutdown in S0 idle Enable (DTSSS0EN): 1: Dynamic thermal sensor shutdown in S0 idle is enabled. When set to 1, the power management logic shuts down the thermal sensor when the CPU is in a C-state and TEMP.TSR andlt;= LTT.LTT. 0: Dynamic thermal sensor shutdown in S0 idle is disabled.
13	0h RW/L	**Dynamic Thermal Sensor Shutdown in C0 Allowed (DTSSIC0):** Dynamic Thermal Sensor Shutdown in C0 Allowed (DTSSIC0) 0: CPU must be in a non-C0 state to allow PCH thermal sensor shutdown. 1: CPU can be in a C0 or non-C0 state to allow PCH thermal sensor shutdown.
12	0h RO	Reserved
11:9	4h RW/L	**Maximum Thermal Sensor Shutdown Time (MAXTSST):** Maximum Thermal Sensor Shutdown Time (MAXTSST) - sets the maximum time that the thermal sensor will be held in a shutdown state assuming no other wake conditions. This register is used to set the expiration time of a timer that is used to wake up the thermal sensor on expiration. 000: 1 s 001: 2 s 010: 4 s 011: 8 s 100: 16 s 101-111: Reserved
8:0	0h RW	**Low Temp Threshold (LTT):** Low Temp Threshold (LTT) - Sets the low maximum temp value used for dynamic thermal sensor shutdown consideration. See DTSSS0EN for details. This register field is not lockable, so that SW can change the values during runtime.

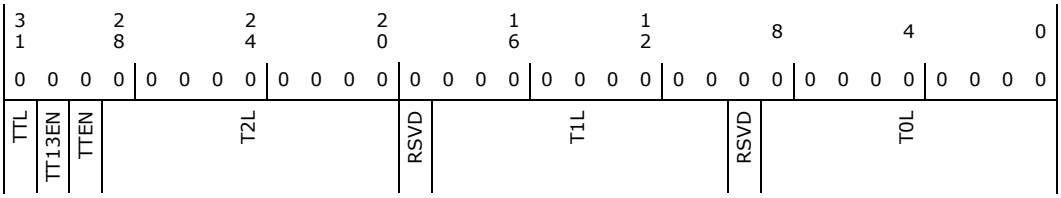
13.4.11 Throttle Levels (TL)—Offset 40h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RW/L	**TT.Lock (TTL):** When set to 1, this entire register (TL) is locked and remains locked until the next platform reset.
30	0h RW/L	**TT.State13 Enable (TT13EN):** When set to 1, then PMSync state 13 will force at least T2 state.
29	0h RW/L	**TT Enable (TTEN):** When set the thermal throttling states are enabled. At reset, BIOS must set bits 28:0 and then do a separate write to set bit 29 to enable throttling. SW may set bit 31 at the same time it sets bit 29 if it wishes to lock the register. If SW wishes to change the values of 28:0, it must first clear the TTEN bit, then change the values in 28:0, and then re-enable TTEN. It is legal to set bits 31, 30 and 29 with the same write. This bit must not be set unless the thermal sensor is already enabled (set TSC[7]=1 and TSC[3:2] = 10).
28:20	0h RW/L	**T2 Level (T2L):** When TTEN = 1 AND TSE = 1 AND (T2L = TSR[8:0] T1L), then the system is in T2 state. When TTEN = 1 AND TSE = 1 AND (TSR[8:0] T2L), then the system is in T3 state. ***Note:*** The T3 condition overrides PMSync[13] and forces the system to T3 if both cases are true. **Software Note:** T2L must be programmed to a value greater than T1L if TTEN=1.
19	0h RO	Reserved
18:10	0h RW/L	**T1 Level (T1L):** When TTEN = 1 AND TSE = 1 AND (T1L = TSR[8:0] T0L), then the system is in T1 state. **Software Note:** T1L must be programmed to a value greater than T0L if TTEN=1.
9	0h RO	Reserved.
8:0	0h RW/L	**T0 Level (T0L):** When TEMP.TSR[8:0] is less than or equal to T0L OR TT.Enable is 0 OR TSE = 0, then the system is in T0 state.

13.4.12 Throttle Level 2 (TL2)—Offset 50h

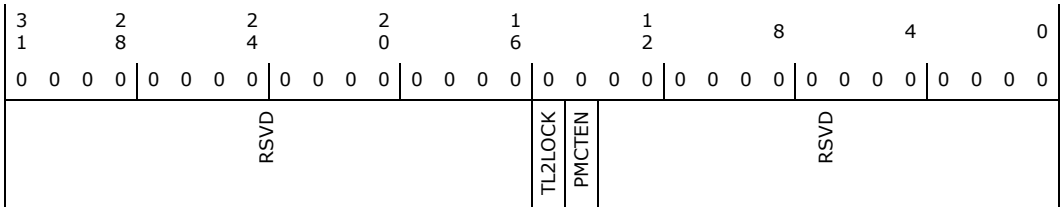
Throttle Level 2

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved
15	0h RW/L	**TL2 lock (TL2LOCK):** TL2.Lock - When set to 1, this entire register (TL2) is locked and remains locked until the next platform reset.
14	0h RW/L	**PMC Throttling Enable (PMCTEN):** PMC Throttling Enable (PMCTEN) - When set to 1 and the PMC is requesting throttling, force at least the T-state that PMC is requesting.
13:0	0h RO	Reserved

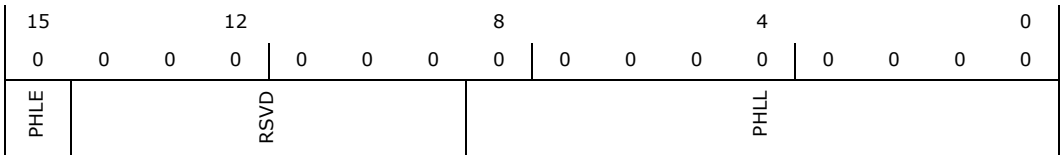
13.4.13 PCH Hot Level (PHL)—Offset 60h

Access Method

Type: MEM Register
(Size: 16 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15	0h RW/L	**PHL Enable (PHLE):** When set and the current temperature reading, TSR, is greater than or equal to PHLL, then the PHOT# pin will be asserted (active low).
14:9	0h RO	Reserved.
8:0	0h RW/L	**PHL Level (PHLL):** Temperature value used for PHOT# pin.

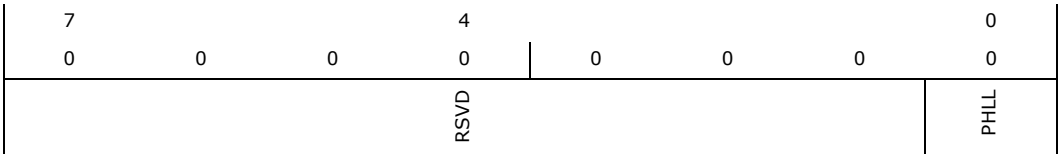
13.4.14 PHL Control (PHLC)—Offset 62h

Access Method

Type: MEM Register
(Size: 8 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:1	0h RO	Reserved
0	0h RW/L	**PHL Lock (PHLL):** When written to a 1, then both PHL and PHLC are locked

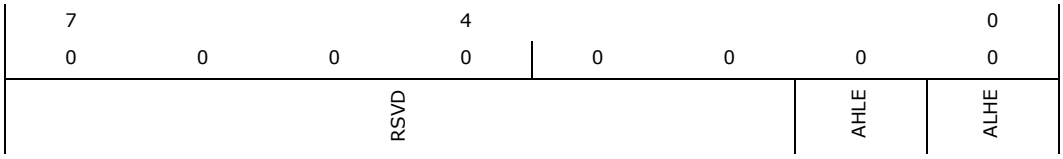
13.4.15 Thermal Alert Status (TAS)—Offset 80h

Access Method

Type: MEM Register
(Size: 8 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RW/1C	**Alert High-to-Low Event (AHLE):** 1: Indicates that a Hot Thermal Sensor trip event occurred based on a lower to higher temperature transition thru the trip point. 0: No trip for this event Software must write a 1 to clear this status bit.
0	0h RW/1C	**Alert Low-to-High Event (ALHE):** 1: Indicates that an Aux Thermal Sensor trip event occurred based on a lower to higher temperature transition thru the trip point. 0: No trip for this event Software must write a 1 to clear this status bit.

13.4.16 PCI Interrupt Event Enables (TSPIEN)—Offset 82h

This register controls the conditions that result in the PCI Interrupt signal from the Thermal Sensor (TS) logic to assert.

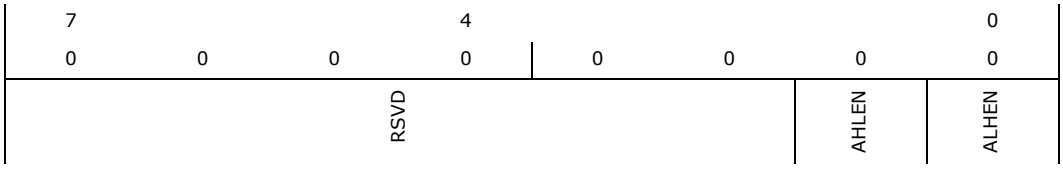
Note: There is a separate enable register per sensor.

Access Method

Type: MEM Register
(Size: 8 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RW	**Alert High-to-Low Enable (AHLEN):** When set to 1, the thermal sensor logic asserts the Thermal logic PCI INTx signal when the corresponding status bit is set in the Thermal Error Status register. When cleared, the corresponding status bit does not result in PCI INTx.
0	0h RW	**Alert Low-to-High Enable (ALHEN):** See the description for bit 1.

13.4.17 General Purpose Event Enables (TSGPEN)—Offset 84h

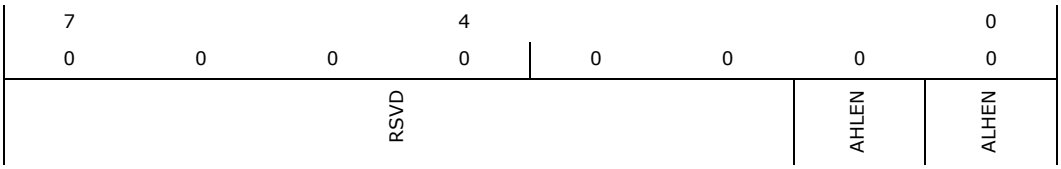
This register controls the conditions that result in the General Purpose Event (GPE) flag (TSS[3]) being set. When the TS GPE signal asserts, the GPE block reports a 1 in the TCOSCI_STS bit.

Access Method

Type: MEM Register
(Size: 8 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RW	**Alert High-to-Low Enable (AHLEN):** When set to 1, the thermal sensor logic asserts its General Purpose Event signal to the GPE block when the corresponding status bit is set in the Thermal Error Status register. When cleared, the corresponding status bit does not result in the GPE signal assertion.
0	0h RW	**Alert Low-to-High Enable (ALHEN):** See the description for bit 1.

13.4.18 Thermal Controller Function Disable (TCFD)—Offset F0h

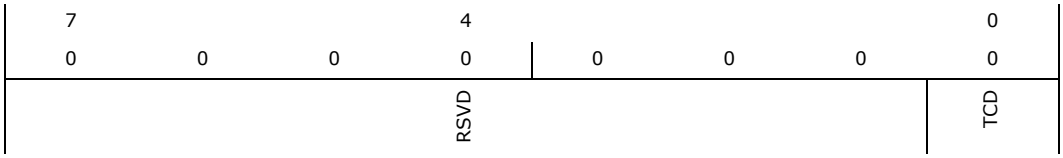
Function Disable bit

Access Method

Type: MEM Register
(Size: 8 bits)

Device: 20
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:1	0h RO	Reserved
0	0h RW	**Thermal Controller Disable (TCD):** Thermal Controller Disable (TCD): When set, the Thermal Controller, is disabled.

§

14 8254 Timers

14.1 Overview

The PCH contains two counters that have fixed uses. All registers and functions associated with the 8254 timers are in the core well. The 8254 unit is clocked by a 14.318 MHz clock derived from 24 MHz crystal clock.

Counter 1 has been removed in this and subsequent generations of PCH. Make sure your software does not need this function.

Counter 0, System Timer

This counter functions as the system timer by controlling the state of IRQ0 and is typically programmed for Mode 3 operation. The counter produces a square wave with a period equal to the product of the counter period (838 ns) and the initial count value. The counter loads the initial count value one counter period after software writes the count value to the counter I/O address. The counter initially asserts IRQ0 and decrements the count value by two each counter period. The counter negates IRQ0 when the count value reaches 0. It then reloads the initial count value and again decrements the initial count value by two each counter period. The counter then asserts IRQ0 when the count value reaches 0, reloads the initial count value, and repeats the cycle, alternately asserting and negating IRQ0.

Counter 2, Speaker Tone

This counter provides the speaker tone and is typically programmed for Mode 3 operation. The counter provides a speaker frequency equal to the counter clock frequency (1.193 MHz) divided by the initial count value. The speaker must be enabled by a write to port 061h (see NMI Status and Control ports).

14.1.1 Timer Programming

The counter/timers are programmed in the following fashion:

1. Write a control word to select a counter.
2. Write an initial count for that counter.
3. Load the least and/or most significant bytes (as required by Control Word Bits 5, 4) of the 16-bit counter.
4. Repeat with other counters.

Only two conventions need to be observed when programming the counters. First, for each counter, the control word must be written before the initial count is written. Second, the initial count must follow the count format specified in the control word (least significant byte only, most significant byte only, or least significant byte, and then most significant byte).

A new initial count may be written to a counter at any time without affecting the counter's programmed mode. Counting is affected as described in the mode definitions. The new count must follow the programmed count format.

If a counter is programmed to read/write two-byte counts, the following precaution applies – a program must not transfer control between writing the first and second byte to another routine which also writes into that same counter. Otherwise, the counter will be loaded with an incorrect count.

The Control Word Register at port 43h controls the operation of all three counters. Several commands are available:

- **Control Word Command.** Specifies which counter to read or write, the operating mode, and the count format (binary or BCD).
- **Counter Latch Command.** Latches the current count so that it can be read by the system. The countdown process continues.
- **Read Back Command.** Reads the count value, programmed mode, the current state of the OUT pins, and the state of the Null Count Flag of the selected counter.

Table 14-1 lists the six operating modes for the interval counters.

Table 14-1. Counter Operating Modes

Mode	Function	Description
0	Out signal on end of count (=0)	Output is 0. When count goes to 0, output goes to 1 and stays at 1 until counter is reprogrammed.
1	Hardware retriggerable one-shot	Output is 0. When count goes to 0, output goes to 1 for one clock time.
2	Rate generator (divide by n counter)	Output is 1. Output goes to 0 for one clock time, then back to 1 and counter is reloaded.
3	Square wave output	Output is 1. Output goes to 0 when counter rolls over, and counter is reloaded. Output goes to 1 when counter rolls over, and counter is reloaded, and so on.
4	Software triggered strobe	Output is 1. Output goes to 0 when count expires for one clock time.
5	Hardware triggered strobe	Output is 1. Output goes to 0 when count expires for one clock time.

14.1.2 Reading from the Interval Timer

It is often desirable to read the value of a counter without disturbing the count in progress. There are three methods for reading the counters—a simple read operation, counter Latch command, and the Read-Back command. Each is explained below.

With the simple read and counter latch command methods, the count must be read according to the programmed format; specifically, if the counter is programmed for two byte counts, two bytes must be read. The two bytes do not have to be read one right after the other. Read, write, or programming operations for other counters may be inserted between them.

14.1.2.1 Simple Read

The first method is to perform a simple read operation. The counter is selected through Port 40h (Counter 0) or 42h (Counter 2).

Note: Performing a direct read from the counter does not return a determinate value, because the counting process is asynchronous to read operations. However, in the case of Counter 2, the count can be stopped by writing to the GATE bit in Port 61h.

14.1.2.2 Counter Latch Command

The Counter Latch command, written to Port 43h, latches the count of a specific counter at the time the command is received. This command is used to ensure that the count read from the counter is accurate, particularly when reading a two-byte count. The count value is then read from each counter's Count register as was programmed by the Control register.

The count is held in the latch until it is read or the counter is reprogrammed. The count is then unlatched. This allows reading the contents of the counters on the fly without affecting counting in progress. Multiple Counter Latch Commands may be used to latch more than one counter. Counter Latch commands do not affect the programmed mode of the counter in any way.

If a Counter is latched and then, some time later, latched again before the count is read, the second Counter Latch command is ignored. The count read is the count at the time the first Counter Latch command was issued.

14.1.2.3 Read Back Command

The Read Back command, written to Port 43h, latches the count value, programmed mode, and current states of the OUT pin and Null Count flag of the selected counter or counters. The value of the counter and its status may then be read by I/O access to the counter address.

The Read Back command may be used to latch multiple counter outputs at one time. This single command is functionally equivalent to several counter latch commands, one for each counter latched. Each counter's latched count is held until it is read or reprogrammed. Once read, a counter is unlatched. The other counters remain latched until they are read. If multiple count Read Back commands are issued to the same counter without reading the count, all but the first are ignored.

The Read Back command may additionally be used to latch status information of selected counters. The status of a counter is accessed by a read from that counter's I/O port address. If multiple counter status latch operations are performed without reading the status, all but the first are ignored.

Both count and status of the selected counters may be latched simultaneously. This is functionally the same as issuing two consecutive, separate Read Back commands. If multiple count and/or status Read Back commands are issued to the same counters without any intervening reads, all but the first are ignored.

If both count and status of a counter are latched, the first read operation from that counter returns the latched status, regardless of which was latched first. The next one or two reads, depending on whether the counter is programmed for one or two type counts, returns the latched count. Subsequent reads return unlatched count.

14.2 8254 Timer Registers Summary

Table 14-2. Summary of 8254 Timer Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
40h	40h	Counter 0 - Interval Timer Status Byte Format Register (C0_ITSBFR)—Offset 40h	C4h
40h	40h	Counter 0 - Counter Access Ports Register (C0_CAPR)—Offset 40h	0h
42h	42h	Counter 2 - Interval Timer Status Byte Format Register (C2_ITSBFR)—Offset 42h	0h
42h	42h	Counter 2 - Counter Access Ports Register (C2_CAPR)—Offset 42h	0h
43h	43h	Timer Control Word Register (TCW)—Offset 43h	0h
43h	43h	Read Back Command (RBC)—Offset 43h	C0h
43h	43h	Counter Latch Command (CLC)—Offset 43h	0h

14.2.1 Counter 0 - Interval Timer Status Byte Format Register (C0_ITSBFR)—Offset 40h

Each counter's status byte can be read following a Read Back Command. If latch status is chosen (bit 4=0, Read Back Command) as a read back option for a given counter, the next read from the counter's Counter Access Ports Register (40h for counter 0 and 42h for counter 2) returns the status byte. The status byte returns the following.

Access Method

Type: I/O Register
(Size: 8 bits)

Device:
Function:

Default: C4h

7			4				0
1	1	0	0	0	1	0	0
COPS	CRSTS	RW_SLT_STS		MD_SLT_STS			CDT_STS

Bit Range	Default and Access	Field Name (ID): Description
7	1h RO	**Counter OUT Pin State (COPS):** When this bit is a 1, the OUT pin of the counter is also a 1. When this bit is a 0, the OUT pin of the counter is also a 0.
6	1h RO	**Count Register Status (CRSTS):** This bit indicates when the last count written to the Count Register (CR) has been loaded into the counting element (CE). The exact time this happens depends on the counter mode, but until the count is loaded into the counting element (CE), the count value will be incorrect. 0 Count has been transferred from CR to CE and is available for reading. 1 Null Count. Count has not been transferred from CR to CE and is not yet available for reading.
5:4	0h RO	**Read/Write Selection Status (RW_SLT_STS):** These reflect the read/write selection made through bits[5:4] of the control register. The binary codes returned during the status read match the codes used to program the counter read/write selection. 00 Counter Latch Command 01 Read/Write Least Significant Byte (LSB) 10 Read/Write Most Significant Byte (MSB) 11 Read/Write LSB then MSB
3:1	2h RO	**Mode Selection Status (MD_SLT_STS):** These bits return the counter mode programming. The binary code returned matches the code used to program the counter mode, as listed under the bit function above. 000 0 Out signal on end of count (=0) 001 1 Hardware retriggerable one-shot x10 2 Rate generator (divide by n counter) x11 3 Square wave output 100 4 Software triggered strobe 101 5 Hardware triggered strobe
0	0h RO	**Countdown Type Status (CDT_STS):** This bit reflects the current countdown type, ether 0 for binary countdown or a 1 for binary coded decimal (BCD) countdown.

14.2.2 Counter 0 - Counter Access Ports Register (C0_CAPR)—Offset 40h

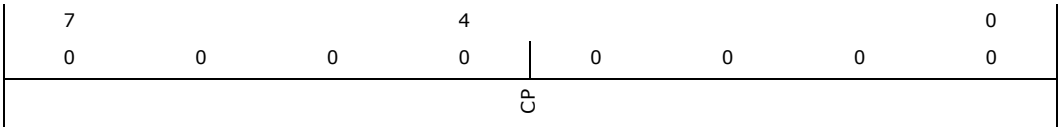
*Address should be 40h

Access Method

Type: I/O Register
(Size: 8 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW	**Counter Port (CP):** Each counter port address is used to program the 16-bit Count Register. The order of programming, either LSB only, MSB only, or LSB then MSB, is defined with the Interval Counter Control Register at port 43h. The counter port is also used to read the current count from the Count Register, and return the status of the counter programming following a Read Back Command.

14.2.3 Counter 2 - Interval Timer Status Byte Format Register (C2_ITSBFR)—Offset 42h

Same definition as counter 0

14.2.4 Counter 2 - Counter Access Ports Register (C2_CAPR)—Offset 42h

Same definition as Counter 0 - Counter Access Ports Register

14.2.5 Timer Control Word Register (TCW)—Offset 43h

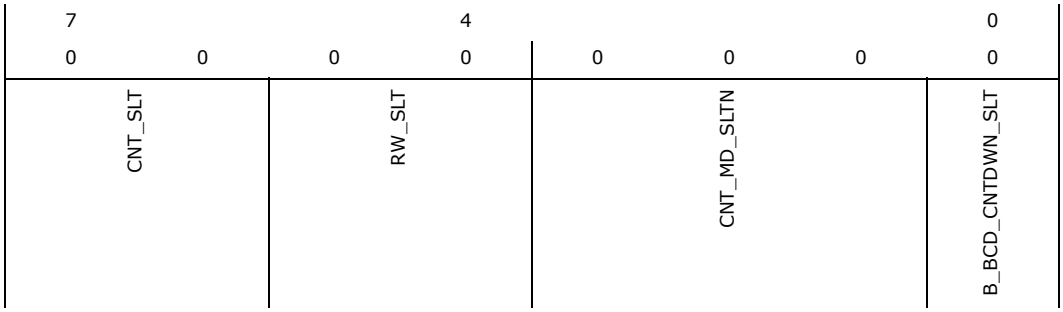
This register is programmed prior to any counter being accessed to specify counter modes. Following reset, the control words for each register are undefined and each counter output is 0. Each timer must be programmed to bring it into a known state. There are two special commands that can be issued to the counters through this register, the Read Back Command and the Counter Latch Command. When these commands are chosen, several bits within this register are redefined.

Access Method

Type: I/O Register
(Size: 8 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:6	0h WO	**Counter Select (CNT_SLT):** The Counter Selection bits select the counter the control word acts upon as shown below. The Read Back Command is selected when bits[7:6] are both 1 00 Counter 0 select 01 Reserved 10 Counter 2 select 11 Read Back Command
5:4	0h WO	**Read/Write Select: (RW_SLT):** These bits are the read/write control bits. The actual counter programming is done through the counter port (40h for counter 0 and 42h for counter 2) 00 Counter Latch Command 01 Read/Write Least Significant Byte (LSB) 10 Read/Write Most Significant Byte (MSB) 11 Read/Write LSB then MSB
3:1	0h WO	**Counter Mode Selection (CNT_MD_SLTN):** These bits select one of six possible modes of operation for the selected counter. 000 0 Out signal on end of count (=0) 001 1 Hardware retriggerable one-shot x10 2 Rate generator (divide by n counter) x11 3 Square wave output 100 4 Software triggered strobe 101 5 Hardware triggered strobe
0	0h WO	**Binary/BCD Countdown Select (B_BCD_CNTDWN_SLT):** 0 Binary countdown is used. The largest possible binary count is 2^16 1 Binary coded decimal (BCD) count is used.The largest possible BCD count is 10^4

14.2.6 Read Back Command (RBC)—Offset 43h

The Read Back Command is used to determine the count value, programmed mode, and current states of the OUT pin and Null count flag of the selected counter or counters. Status and/or count may be latched in any or all of the counters by selecting the counter during the register write. The count and status remain latched until read, and further latch commands are ignored until the count is read.
Both count and status of the selected counters may be latched simultaneously by setting both bit 5 and bit 4 to 0. If both are latched, the first read operation from that counter returns the latched status. The next one or two reads, depending on whether the counter is programmed for one or two byte counts, returns the latched count. Subsequent reads return an unlatched count.

Access Method

Type: I/O Register
(Size: 8 bits)

Device:
Function:

Default: C0h

7				4			0
1	1	0	0	0	0	0	0
RBC		LCSC	LSSC	CNT_2_SLT	RSVD	CNT_0_SLT	RSVD

Bit Range	Default and Access	Field Name (ID): Description
7:6	3h WO	**Read Back Command (RBC):** Must be 11 to select the Read Back Command
5	0h WO	**Latch Count of Selected Counters (LCSC):** 0 Current count value of the selected counters will be latched 1 Current count will not be latched
4	0h WO	**Latch Status of Selected Counters (LSSC):** 0 Status of the selected counters will be latched 1 Status will not be latched
3	0h WO	**Counter 2 Select (CNT_2_SLT):** When set to 1, Counter 2 count and/or status will be latched
2	0h RO	Reserved
1	0h WO	**Counter 0 Select (CNT_0_SLT):** When set to 1, Counter 0 count and/or status will be latched.
0	0h RO	Reserved

14.2.7 Counter Latch Command (CLC)—Offset 43h

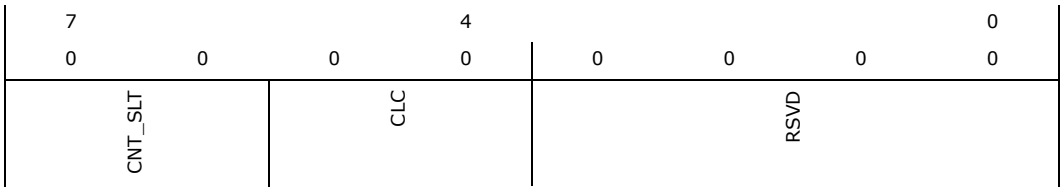
The Counter Latch Command latches the current count value. This command is used to insure that the count read from the counter is accurate. The count value is then read from each counter's count register through the Counter Ports Access Ports Register (40h for counter 0 and 42h for counter 2). The count must be read according to the programmed format i.e., if the counter is programmed for two byte counts, two bytes must be read. The two bytes do not have to be read one right after the other (read, write, or programming operations for other counters may be inserted between the reads). If a counter is latched once and then latched again before the count is read, the second Counter Latch Command is ignored.

Access Method

Type: I/O Register
(Size: 8 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:6	0h WO	**Counter Selection (CNT_SLT):** These bits select the counter for latching. If 11 is written, then the write is interpreted as a read back command. 00 = Counter 0 01 = Reserved 10 = Counter 2
5:4	0h WO	**Counter Latch Command (CLC):** Write 00 to select the Counter Latch Command.
3:0	0h RO	Reserved

§

15 Integrated High Definition Audio

15.1 Acronyms

Acronyms	Description
HDA	High Definition Audio
PCM	Pulse Code Modulation
SoC	System On Chip

15.2 References

Specification	Location
High Definition Audio Specification	http://www.intel.com/content/www/us/en/standards/high-definition-audio-specification.html

15.3 Overview

The Intel® High Definition Audio (Intel® HD Audio) subsystem is a collection of controller, DSP, memory, and link that provide an audio experience in the platform. The controller, memory, and link form the basic audio controller to provide the streaming of audio from host software to an external audio codec with the host CPU providing the audio enrichment.

15.4 Signal Description

Name	Type	Description
HD Audio Signals		
HDA_RST_N	O	**Intel HD Audio Reset**: Master H/W reset to internal/external codecs.
HDA_SYNCS	O	**Intel HD Audio Sync:** 48-kHz fixed rate frame sync to the codecs. Also used to encode the stream number.
HDA_BLK	O	**Intel HD Audio Bit Clock:** Up to 24-MHz serial data clock generated by the Intel HD Audio controller.
HDA_SDO	O	**Intel HD Audio Serial Data Out:** Serial TDM data output to the codecs. The serial output is double-pumped for a bit rate of up to 48 Mb/s.
HDA_SDI_0	I	**Intel HD Audio Serial Data In 0:** Serial TDM data input from the two codec(s). The serial input is single-pumped for a bit rate of up to 24 Mb/s. These signals contain integrated pull-down resistors, which are enabled while the primary well is powered.
HDA_SDI_1	I	**Intel HD Audio Serial Data In 1:** Serial TDM data input from the two codec(s). The serial input is single-pumped for a bit rate of up to 24 Mb/s. These signals contain integrated pull-down resistors, which are enabled while the primary well is powered.

15.5 Integrated Pull-Ups and Pull-Downs

Table 15-1. Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value (Ω)	Notes
HDA_SYNC	Pull-Down	20K	
HDA_SDO	Pull-Down	20k	
HDA_SDI[1:0]	Pull-Down	20K	
Note: Integrated pull-ups and pull-downs will not be valid until all the voltages have reached a valid level.			

15.6 I/O Signal Planes and States

Table 15-2. I/O Signal Planes and States

Signal Name	Power Plane	During Reset	Immediately after Reset	S3/S4/S5	Deep Sx
HD Audio Interface					
HDA_RST_N	Primary	Driven Low	Driven Low	Driven Low	OFF
HDA_SYNC	Primary	Internal Pull-down	Driven Low	Internal Pull-down	OFF
HDA_BLK	Primary	Driven Low	Driven Low	Driven Low	OFF
HDA_SDO	Primary	Internal Pull-down	Driven Low	Internal Pull-down	OFF
HDA_SDI[1:0]	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF
Note: Integrated pull-ups and pull-downs will not be valid until all the voltages have reached a valid level.					

15.7 Features

The Integrated High Definition Audio subsystem features are listed below.

15.7.1 Intel High Definition Audio Controller Capabilities

- PCI / PCI Express controller.
- Supports data transfers, descriptor fetches, and DMA position writes using VC0.
  - VC1 is supported as a backup usage model in case low latency is crucial.
- Independent Bus Master logic for 14 general purpose streams: five input and nine output.
- Supports variable length stream slots.
- Supports up to:
  - 14 streams (five input, nine output)
  - 16 channels per stream
  - 32 bits/sample
  - 192 kHz sample rate
- Supports memory-based command/response transport.
- Supports optional Immediate Command/Response mechanism.
- Supports output and input stream synchronization.
- Supports global time synchronization.

- Supports MSI interrupt delivery.
- Support for ACPI D3 and D0 Device States.
- Supports Function Level Reset (FLR).
  - — Only if exposed as PCI Express device.
- Supports Intel's Power Optimizer power management.
  - — Support 1 ms of buffering with all DMA running with maximum bandwidth.
  - — Support 10 ms of buffering with one output DMA and one input DMA running at two channels, 96 kHz, 16-bit audio.

15.7.2 Intel HD Audio link capabilities

- Two SDI signals to support two external codecs.
- Drives variable frequency (6 MHz to 24 MHz) BCLK to support:
  - — SDO double pumped up to 48 Mb/s
  - — SDI's single pumped up to 24 Mb/s
- Provides cadence for 44.1 kHz-based sample rate output.
- Supports 1.5V, 1.8V and 3.3V modes.

15.8 High Definition Audio (D31:F3) PCI Configuration Registers Summary

Table 15-3. Summary of High Definition Audio (D31:F3) PCI Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	1h	Vendor Identification (VID)—Offset 0h	8086h
2h	3h	Device ID (DID)—Offset 2h	8C20h
4h	5h	Command (CMD)—Offset 4h	0h
6h	7h	Status (STS)—Offset 6h	10h
8h	8h	Revision Identification (RID)—Offset 8h	0h
9h	9h	Programming Interface (PI)—Offset 9h	0h
Ah	Ah	Sub Class Code (SCC)—Offset Ah	3h
Bh	Bh	Base Class Code (BCC)—Offset Bh	4h
Ch	Ch	Cache Line Size (CLS)—Offset Ch	0h
Dh	Dh	Latency Timer (LT)—Offset Dh	0h
Eh	Eh	Header Type (HTYPE)—Offset Eh	0h
10h	13h	Intel HD Audio Base Lower Address (HDALBA)—Offset 10h	4h
14h	17h	Intel HD Audio Base Upper Address (HDAUBA)—Offset 14h	0h
18h	1Bh	Shadowed PCI Configuration Lower Base Address (SPCLBA)—Offset 18h	4h
1Ch	1Fh	Shadowed PCI Configuration Upper Base Address (SPCUBA)—Offset 1Ch	0h
2Ch	2Dh	Subsystem Vendor ID (SVID)—Offset 2Ch	0h
2Eh	2Fh	Subsystem ID (SID)—Offset 2Eh	0h
30h	30h	Capability Pointer (CAPPTR)—Offset 30h	50h
31h	31h	Interrupt Line (INTLN)—Offset 31h	0h
32h	32h	Interrupt Pin (INTPN)—Offset 32h	1h

Table 15-3. Summary of High Definition Audio (D31:F3) PCI Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
44h	47h	Power Gating Control (PGCTL)—Offset 44h	0h
48h	4Bh	Clock Gating Control (CGCTL)—Offset 48h	803F01F9h
50h	51h	PCI Power Management Capability ID (PID)—Offset 50h	6001h
52h	53h	Power Management Capabilities (PC)—Offset 52h	C843h
54h	57h	Power Management Control And Status (PCS)—Offset 54h	8h
60h	61h	MSI Capability ID (MID)—Offset 60h	7005h
62h	63h	Message Signal Interrupt Message Control (MMC)—Offset 62h	80h
64h	67h	MSI Message Lower Address (MMLA)—Offset 64h	0h
68h	6Bh	MSI Message Upper Address (MMUA)—Offset 68h	0h
6Ch	6Dh	Message Signal Interrupt Capability ID Register (MMD)—Offset 6Ch	0h
70h	71h	PCI Express Capability ID (PXID)—Offset 70h	10h
72h	73h	PCI Express Capabilities (PXC)—Offset 72h	91h
74h	77h	Device Capabilities (DEVCAP)—Offset 74h	10000000h
78h	79h	Device Control (DEVC)—Offset 78h	2800h
7Ah	7Bh	Device Status (DEVS)—Offset 7Ah	10h
F8h	FBh	Manufacturing Process (MANID)—Offset F8h	0h
100h	103h	Virtual Channel Enhanced Capability Header (VCCAP)—Offset 100h	0h
104h	107h	Port VC Capability Register 1 (PVCCAP1)—Offset 104h	1h
108h	10Bh	Port VC Capability Register 2 (PVCCAP2)—Offset 108h	0h
10Ch	10Dh	Port VC Control Register (PVCCTL)—Offset 10Ch	0h
10Eh	10Fh	Port VC Status Register (PVCSTS)—Offset 10Eh	0h
110h	113h	VC0 Resource Capability Register (VC0CAP)—Offset 110h	0h
114h	117h	VC0 Resource Control Register (VC0CTL)—Offset 114h	800000FFh
11Ah	11Bh	VC0 Resource Status Register (VC0STS)—Offset 11Ah	0h
11Ch	11Fh	VCi Resource Capability Register (VCiCAP)—Offset 11Ch	0h
120h	123h	VCi Resource Control Register (VCiCTL)—Offset 120h	0h
126h	127h	VCi Resource Status Register (VCiSTS)—Offset 126h	0h
130h	133h	Root Complex Link Declaration Enhanced (RCCAP)—Offset 130h	10005h
134h	137h	Element Self Description (ESD)—Offset 134h	F000100h
140h	143h	Link 1 Description (L1DESC)—Offset 140h	1h
148h	14Bh	Link 1 Lower Address (L1LADD)—Offset 148h	0h
14Ch	14Fh	Link 1 Upper Address (L1UADD)—Offset 14Ch	0h

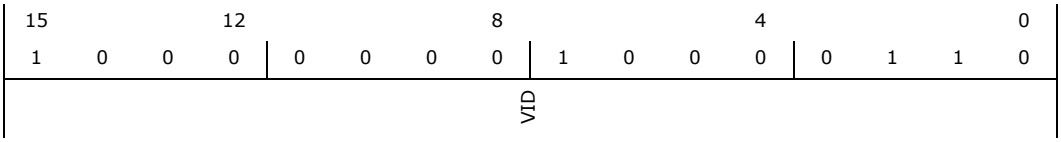
15.8.1 Vendor Identification (VID)—Offset 0h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 3

Default: 8086h



Bit Range	Default and Access	Field Name (ID): Description
15:0	8086h RO	**Vendor ID (VID):** Indicates that Intel is the vendor.

15.8.2 Device ID (DID)—Offset 2h

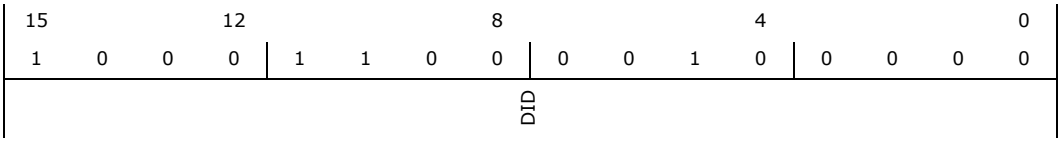
This register is not affected by $D3_{HOT}$ to D0 reset or FLR.

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 3

Default: 8C20h



Bit Range	Default and Access	Field Name (ID): Description
15:0	8C20h RO/V	**Device ID (DID):** Indicates the device number assigned by the SIG. Bits [2:0] of the DID is controlled by fuse. IOSF Sideband Set ID Value message initializes bits [15:7] of this register value, whilst bits [6:0] of this register value are hardcoded. See the Global Device ID table in Register and Memory Mappings CSpec chapter for the DID value.

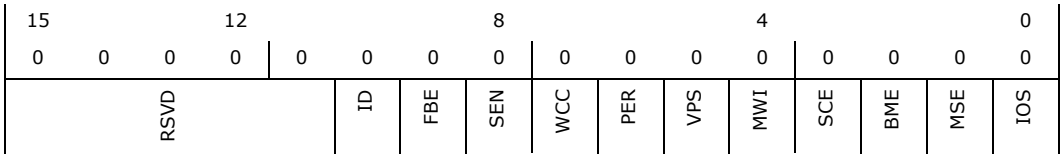
15.8.3 Command (CMD)—Offset 4h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:11	0h RO	Reserved.
10	0h RW	**Interrupt Disable (ID):** Enables the device to assert an INTx#. When set, the Intel® HD Audio controller's INTx# signal will be de-asserted. When cleared, the INTx# signal may be asserted. Note that this bit does not affect the generation of MSIs.
9	0h RO	**Fast Back to Back Enable (FBE):** Not implemented. Hardwired to 0.
8	0h RW	**SERR Enable (SEN):** Functionality not implemented. This bit is R/W to pass PCIe compliance testing.
7	0h RO	**Wait Cycle Control (WCC):** Not implemented. Hardwired to 0.
6	0h RW	**Parity Error Response (PER):** Functionality not implemented. This bit is R/W to pass PCIe compliance testing.
5	0h RO	**VGA Palette Snoop (VPS):** Not implemented. Hardwired to 0.
4	0h RO	**Memory Write and Invalidate Enable (MWI):** Not implemented. Hardwired to 0.
3	0h RO	**Special Cycle Enable (SCE):** Not implemented. Hardwired to 0.
2	0h RW	**Bus Master Enable (BME):** 1 = Enable, 0 = Disable. Controls standard PCI Express bus mastering capabilities for Memory and I/O, reads and writes. ***Note:*** This also controls MSI generation since MSI are essentially Memory writes.
1	0h RW	**Memory Space Enable (MSE):** When set, enables memory space accesses to the Intel HD Audio controller.
0	0h RO	**I/O Space (IOS):** The Intel HD Audio controller does not implement I/O Space, therefore this bit is hardwired to 0.

15.8.4 Status (STS)—Offset 6h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 3

Default: 10h

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
DPE	SERRS	RMA	RTA	STA	DEVT		MDPE	FBC	RSVD	C66	CLIST	IS	RSVD		

Bit Range	Default and Access	Field Name (ID): Description
15	0h RO	**Detected Parity Error (DPE):** Not implemented. Hardwired to 0.
14	0h RO	**SERR# Status (SERRS):** Not implemented. Hardwired to 0.
13	0h RW/1C/V	**Received Master Abort (RMA):** If the completion status received from IOSF is UR, this bit is set. SW writes a 1 to this bit to clear it.
12	0h RW/1C/V	**Received Target Abort (RTA):** If the completion status received from IOSF is CA, this bit is set. SW writes a 1 to this bit to clear it.
11	0h RO	**Signaled Target-Abort (STA):** Not implemented. Hardwired to 0.
10:9	0h RO	**DEVSEL# Timing Status (DEVT):** Does not apply. Hardwired to 0.
8	0h RO	**Master Data Parity Error (MDPE):** Not implemented. Hardwired to 0.
7	0h RO	**Fast Back to Back Capable (FBC):** Does not apply. Hardwired to 0.
6	0h RO	Reserved
5	0h RO	**66 MHz Capable (C66):** Does not apply. Hardwired to 0.
4	1h RO	**Capabilities List Exists (CLIST):** Indicates that the controller contains a capabilities pointer list. The first item is pointed to by looking at configuration offset 34h.
3	0h RO/V	**Interrupt Status (IS):** Reflects the state of the INTx# signal at the input of the enable/disable circuit. This bit is a 1 when the INTx# is asserted. This bit is a 0 after the interrupt is cleared (independent of the state of the Interrupt Disable bit in the command register). Note that this bit is not set by an MSI.
2:0	0h RO	Reserved

15.8.5 Revision Identification (RID)—Offset 8h

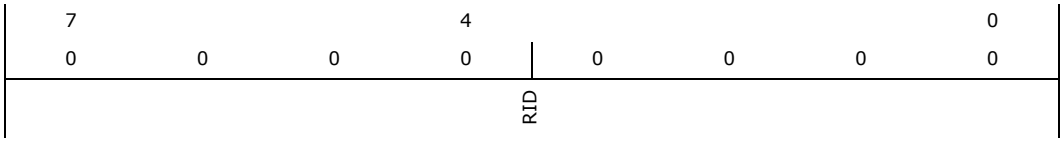
This register is not affected by $D3_{HOT}$ to D0 reset or FLR.

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO/V	**Revision ID (RID):** Indicates the device specific revision identifier. IOSF Sideband Set ID Value message initializes this register value.

15.8.6 Programming Interface (PI)—Offset 9h

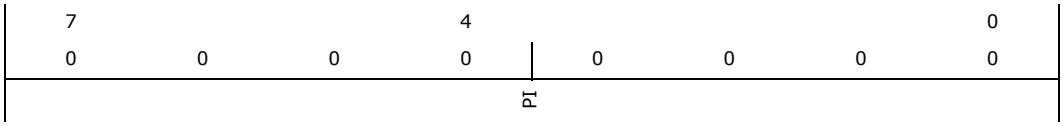
This register is not affected by $D3_{HOT}$ to D0 reset or FLR.

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW/L	**Programming Interface (PI):** Value assigned to the Intel HD Audio subsystem. Locked when FNCFG.BCLD = 1.

15.8.7 Sub Class Code (SCC)—Offset Ah

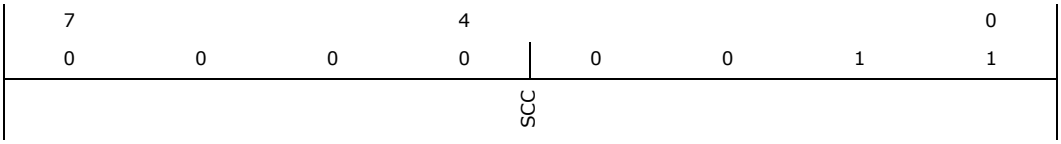
This register is not affected by $D3_{HOT}$ to D0 reset or FLR.

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 3

Default: 3h



Bit Range	Default and Access	Field Name (ID): Description
7:0	3h RW/L	**Sub Class Code (SCC):** This indicates the device is an Intel HD Audio audio device, in the context of a multimedia device. Locked when FNCFG.BCLD = 1.

15.8.8 Base Class Code (BCC)—Offset Bh

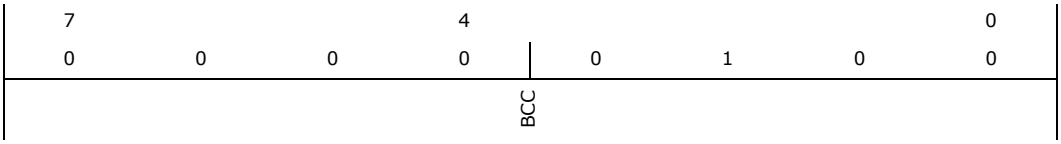
This register is not affected by $D3_{HOT}$ to D0 reset or FLR.

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 3

Default: 4h



Bit Range	Default and Access	Field Name (ID): Description
7:0	4h RW/L	**Base Class Code (BCC):** This register indicates that the function implements a multimedia device. Locked when FNCFG.BCLD = 1.

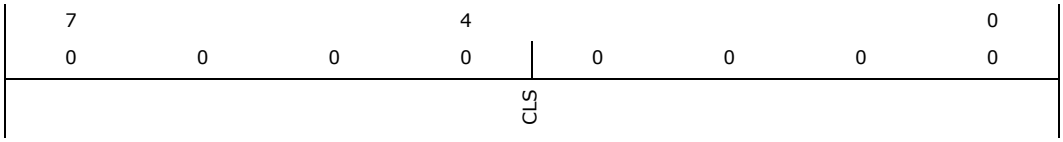
15.8.9 Cache Line Size (CLS)—Offset Ch

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW	**Cache Line Size (CLS):** Does not apply to PCI Express. *PCI Express Specification* requires this to be implemented as a R/W register but has no functional impact on the PCH.

15.8.10 Latency Timer (LT)—Offset Dh

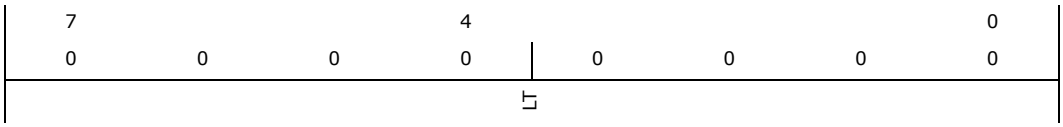
RO. Hardwired to 00.

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW/L	**Latency Timer (LT):** Does not apply to PCI Express. RO as 00h if PCI Express. If configured to appear as PCI device, maintain RW for Legacy PCI SW compliancy. Locked when FNCFG.HDASPCID = 0.

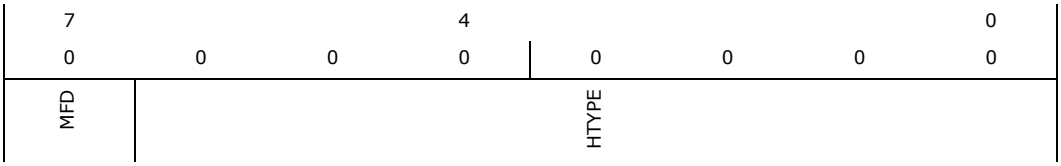
15.8.11 Header Type (HTYPE)—Offset Eh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7	0h RW/L	**Multi Function Device (MFD):** Value of 0 indicates a single function device. Value of 1 indicates a multi function device. Locked when FNCFG.BCLD = 1.
6:0	0h RO	**Header Type (HTYPE):** Implements Type 0 Configuration header.

15.8.12 Intel HD Audio Base Lower Address (HDALBA)—Offset 10h

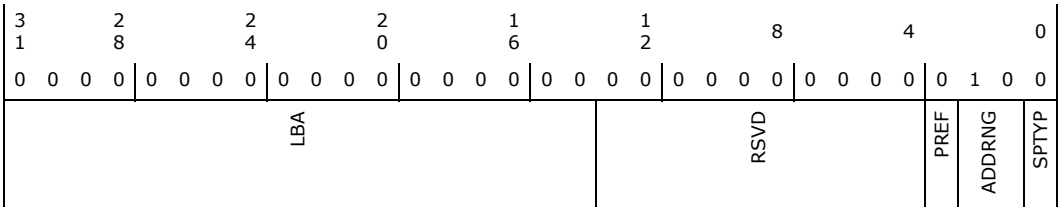
This BAR creates a selected size of memory space to signify the base address of the Intel HD Audio memory mapped configuration registers depending on implementation.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 3

Default: 4h



Bit Range	Default and Access	Field Name (ID): Description
31:14	0h RW	**Lower Base Address (LBA):** Base address for the Intel HD Audio subsystem s memory mapped configuration registers. 16 KB are requested by hardwiring bits [13:4] to 0s.
13:4	0h RO	Reserved

Bit Range	Default and Access	Field Name (ID): Description
3	0h RO	**Prefetchable (PREF):** Indicates that this BAR is NOT pre-fetchable.
2:1	2h RO	**Address Range (ADDRNG):** Indicates that this BAR can be located anywhere in 64-bit address space.
0	0h RO	**Space Type (SPTYP):** Indicates that this BAR is located in memory space.

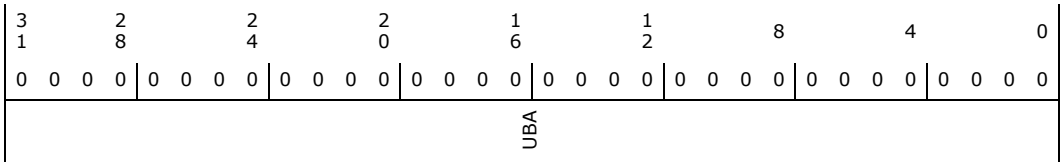
15.8.13 Intel HD Audio Base Upper Address (HDAUBA)—Offset 14h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Intel HD Audio Upper Base Address (UBA):** Upper 32 bits of the base address for the Intel HD Audio controller's memory mapped configuration registers.

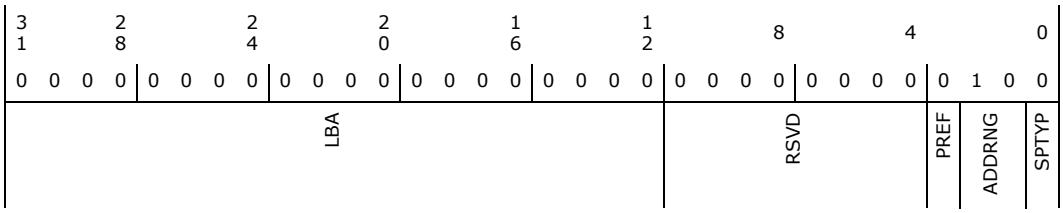
15.8.14 Shadowed PCI Configuration Lower Base Address (SPCLBA)—Offset 18h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 3

Default: 4h



Bit Range	Default and Access	Field Name (ID): Description
31:12	0h RW/L	**Lower Base Address (LBA):** Base address for the PCI Configuration register shadowed to memory mapped. 4 KB is requested by hardwiring bits [11:4] to 0s. Locked when PCICFGCTL0.SPCBAD = 1.
11:4	0h RO	Reserved
3	0h RO	**Prefetchable (PREF):** Indicates that this BAR is NOT pre-fetchable.
2:1	2h RO/V	**Address Range (ADDRNG):** Indicates that this BAR can be located anywhere in 64-bit address space.
0	0h RO	**Space Type (SPTYP):** Indicates that this BAR is located in memory space.

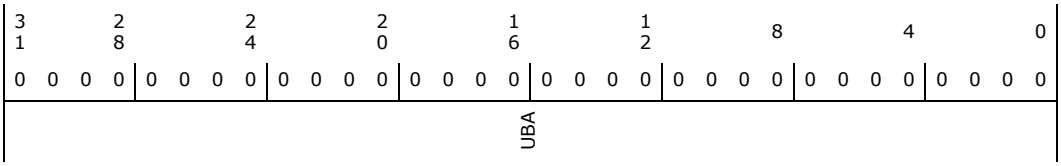
15.8.15 Shadowed PCI Configuration Upper Base Address (SPCUBA)—Offset 1Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/L	**Upper Base Address (UBA):** Upper 32 bits of the Base address for the PCI Configuration register shadowed to memory mapped. Locked when PCICFGCTL0.SPCBAD = 1.

15.8.16 Subsystem Vendor ID (SVID)—Offset 2Ch

This register should be implemented for any function that could be instantiated more than once in a given system, for example, a system with two audio subsystems, one down on the motherboard and the other plugged into a PCI expansion slot, should have the SVID register implemented. The SVID register, in combination with the Subsystem ID register, enables the operating environment to distinguish one audio subsystem from the other(s).

Software (BIOS) will write the value to this register. After that, the value can be read, but writes to the register will have no effect. The write to this register should be combined with the write to the SID to create one 32-bit write. This register is not affected by $D3_{HOT}$ to D0 reset or FLR.

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 3

Default: 0h

Bit Range	Default and Access	Field Name (ID): Description
15:0	0h RW/L	**SVID (SVID):** These RW bits have no functionality. Locked when FNCFG.BCLD = 1.

15.8.17 Subsystem ID (SID)—Offset 2Eh

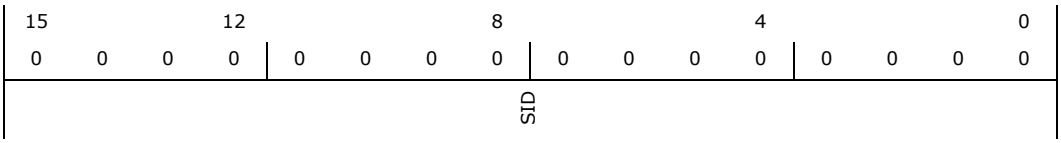
This register should be implemented for any function that could be instantiated more than once in a given system, for example, a system with two audio subsystems, one down on the motherboard and the other plugged into a PCI expansion slot. The SID register, in combination with the Subsystem Vendor ID register make it possible for the operating environment to distinguish one audio subsystem from the other(s). Software (BIOS) will write the value to this register. After that, the value can be read, but writes to the register will have no effect. The write to this register should be combined with the write to the SVID to create one 32-bit write. This register is not affected by $D3_{HOT}$ to D0 reset or FLR.

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:0	0h RW/L	**SID (SID):** These RW bits have no functionality. Locked when FNCFG.BCLD = 1.

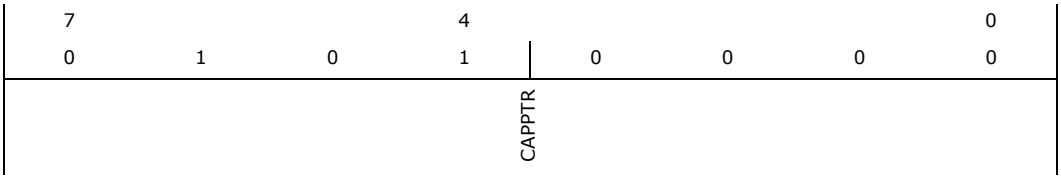
15.8.18 Capability Pointer (CAPPTR)—Offset 30h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 3

Default: 50h



Bit Range	Default and Access	Field Name (ID): Description
7:0	50h RO	**Capability Pointer (CAPPTR):** Indicates that the first capability pointer offset is offset 50h (Power Management Capability).

15.8.19 Interrupt Line (INTLN)—Offset 31h

This register is not affected by FLR.

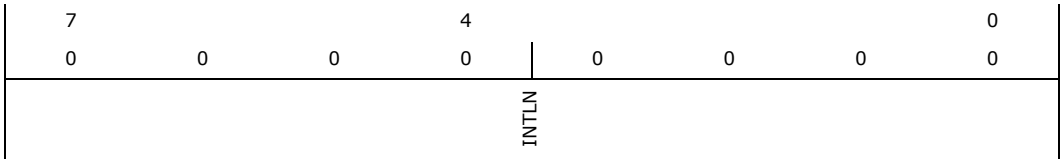
Implementation Notes: Due to legacy implementation of the INTLN field being implemented as reset by $D3_{HOT}$ to D0 Reset but not FLR and already work well with existing SW (despite the FLR section, in the *PCIe Specification* does not have this register in the preservation list), this legacy implementation remains no change and documented here as reset by D3ONLY, which is a deviation from D3RST definition by excluding FLR.

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW	**Interrupt Line (INTLN):** Hardware does not use this field directly. It is used to communicate to software the interrupt line that the interrupt pin is connected to.

15.8.20 Interrupt Pin (INTPN)—Offset 32h

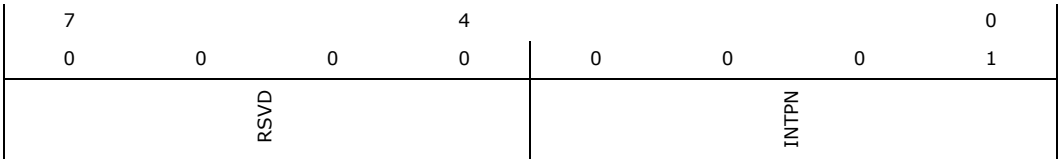
This register is not affected by $D3_{HOT}$ to D0 reset or FLR.

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 3

Default: 1h



Bit Range	Default and Access	Field Name (ID): Description
7:4	0h RO	Reserved
3:0	1h RW/L	**Interrupt Pin (INTPN):** Identifies the interrupt pin the function uses. 0h: No interrupt pin 1h: INTA 2h: INTB 3h: INTC 4h: INTD 5h Fh: reserved Locked when FNCFG.BCLD = 1.

15.8.21 Power Gating Control (PGCTL)—Offset 44h

D3PGD are meant for the Intel HD Audio driver software to control whether the Intel HD Audio subsystem should be power gated or not in D3.

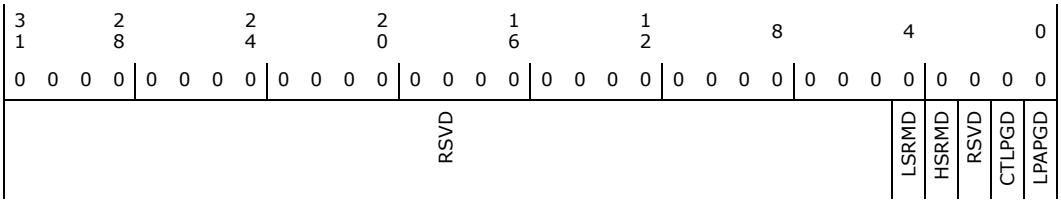
Note: The power gating will only be initiated when out of platform reset, if conditions are met.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:5	0h RO	Reserved
4	0h RW	**LP SRAM Retention Module Disable (LSRMD):** Register is used to disable the LP SRAM retention mode capability of the L2 SRAMs.
3	0h RW	**HP SRAM Retention Module Disable (HSRMD):** Register is used to disable the HP SRAM retention mode capability of the L2 SRAMs.
2	0h RO	Reserved
1	0h RW	**D3 Power Gating Disable (CTLPGD):** Register is used to disable the power gating capability during D3 state.
0	0h RW	**Low Power Audio Power Gating Disable (LPAPGD):** Register is used to disable the power gating capability of the Primary well (gated-controller) domain.

15.8.22 Clock Gating Control (CGCTL)—Offset 48h

The trunk clock gating enable and local clock gating enables are meant for the BIOS or driver to enable or disable the HW capability to detect idle condition and clock gate accordingly. HW should treat these clock gate enable register bits as 0 if FNCFG.CGD = 1 or FUSVAL.CGD = 1.

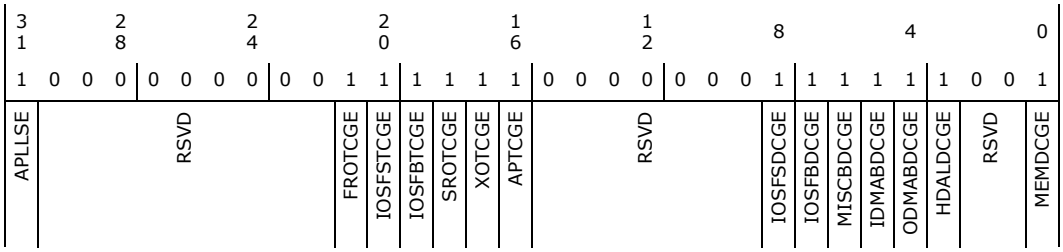
Note: The clock gating will only be initiated when out of platform reset, if conditions are met.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 3

Default: 803F01F9h



Bit Range	Default and Access	Field Name (ID): Description
31	1h RW	**Audio PLL Shutdown Enable (APLLSE):** Set to 1 to enable PLL shutdown after trunk clock gating.
30:22	0h RO	Reserved
21	1h RW	**Fast RING Oscillator Trunk Clock Gating Enable (FROTCGE):** Set to 1 to enable trunk clock gating. If enabled, HW will trunk clock gate when no logic are using this clock (i.e., all local clock gating condition is true).
20	1h RW	**IOSF Sideband Trunk Gate Enable (IOSFSTCGE):** Enable IOSF trunk clock gating functionality on IOSF interface. When set, IOSF Sideband interface clock request can de-assert to allow trunk clock gating.

Bit Range	Default and Access	Field Name (ID): Description
19	1h RW	**IOSF Backbone Trunk Gate Enable (IOSFBTCGE):** Enable IOSF trunk clock gating functionality on IOSF interface. When set, IOSF Primary interface clock request can de-assert to allow trunk clock gating.
18	1h RW	**Slow RING Oscillator Trunk Clock Gating Enable (SROTCGE):** Set to 1 to enable trunk clock gating. If enabled, HW will trunk clock gate when no logic are using this clock (i.e., all local clock gating condition is true).
17	1h RW	**XTAL Oscillator Trunk Clock Gating Enable (XOTCGE):** Set to 1 to enable trunk clock gating. If enabled, HW will trunk clock gate when no logic are using this clock (i.e., all local clock gating condition is true).
16	1h RW	**Audio PLL Trunk Clock Gating Enable (APTCGE):** Set to 1 to enable trunk clock gating. If enabled, HW will trunk clock gate when no logic are using this clock (i.e., all local clock gating condition is true).
15:9	0h RO	Reserved
8	1h RW	**IOSF Sideband Dynamic Clock Gate Enable (IOSFSDCGE):** Enable IOSF dynamic clock gating functionality inside IOSF interface. When set, IOSF Sideband clock gating functionality is enabled.
7	1h RW	**IOSF Backbone Dynamic Clock Gate Enable (IOSFBDCGE):** Enable IOSF dynamic clock gating functionality inside IOSF interface. When set, IOSF Primary clock gating functionality is enabled.
6	1h RW	**Miscellaneous Backbone Dynamic Clock Gating Enable (MISCBDCGE):** This controls dynamic clock gating of backbone (Command/data) clocks to the rest of the Intel HD Audio controller (i.e. other than the IOSF, Input DMA engine, and Output DMA engine). When this bit is asserted, dynamic clock gating logic is enabled for backbone clocks to the rest of the Intel HD Audio controller.
5	1h RW	**IDMA Backbone Dynamic Clock Gating Enable (IDMABDCGE):** This controls dynamic clock gating of backbone (Command/data) clocks to each Input DMA engine. When this bit is asserted, dynamic clock gating logic is enabled for backbone clocks to each Input DMA engine.
4	1h RW	**ODMA Backbone Dynamic Clock Gating Enable (ODMABDCGE):** This controls dynamic clock gating of backbone (Command/data) clocks to each Output DMA engines. When this bit is asserted, dynamic clock gating logic is enabled for backbone clocks to each Output DMA engine.
3	1h RW	**HD Audio Link Dynamic Clock Gating Enable (HDALDCGE):** This controls dynamic clock gating of bitclk to Link Layer and each Input/Output DMA engine. When this bit is asserted, dynamic clock gating logic is enabled for bitclk.
2:1	0h RO	Reserved
0	1h RW	**Memory Dynamic Clock Gating Enable (MEMDCGE):** When set to 1, it allows HW to automatically detect for idle condition and clock gate Memory block. When clear to 0, it disables this HW auto detect idle clock gating.

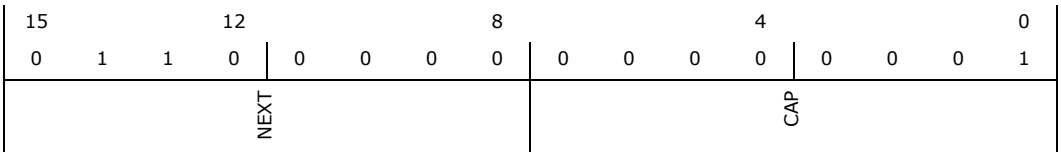
15.8.23 PCI Power Management Capability ID (PID)—Offset 50h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 3

Default: 6001h



Bit Range	Default and Access	Field Name (ID): Description
15:8	60h RW/L	**Next Capability (NEXT):** Points to the next capability structure (MSI). Locked when FNCFG.BCLD = 1.
7:0	1h RO	**Cap ID (CAP):** Indicates that this pointer is a PCI power management capability.

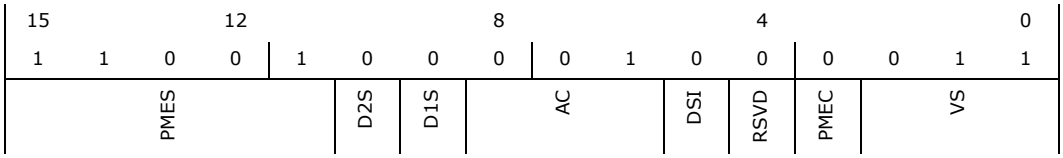
15.8.24 Power Management Capabilities (PC)—Offset 52h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 3

Default: C843h



Bit Range	Default and Access	Field Name (ID): Description
15:11	19h RW/L	**PME_Support (PMES):** Indicates PME# can be generated from D3 and D0 states. Programmable by BIOS for the option to declare SUS well wake is supported or not: 19h (SUS well wake supported) or 09h (SUS well wake not supported). Locked when FNCFG.BCLD = 1.
10	0h RO	**D2_Support (D2S):** The D2 state is not supported.
9	0h RO	**D1_Support (D1S):** The D1 state is not supported.
8:6	1h RW/L	**Aux_Current (AC):** Reports 55 mA maximum Suspend well current required when in the $D3_{COLD}$ state. Programmable by BIOS for the option to declare SUS well wake is supported or not: 001b (SUS well wake supported) or 000b (SUS well wake not supported). Locked when FNCFG.BCLD = 1.
5	0h RO	**Device Specific Initialization (DSI):** Indicates that no device-specific initialization is required.
4	0h RO	Reserved
3	0h RO	**PME Clock (PMEC):** Does not apply. Hardwired to 0.
2:0	3h RW/L	**Version (VS):** Indicates support for Revision 1.2 of the PCI Power Management Specification. Programmable by BIOS to older revision if compatibility issue is found. Locked when FNCFG.BCLD = 1.

15.8.25 Power Management Control And Status (PCS)—Offset 54h

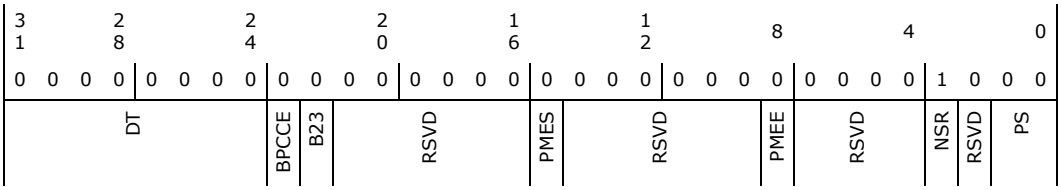
PMES and PMEE bits reside in Resume well, and reset by resume reset.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 3

Default: 8h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	**Data (DT):** Does not apply. Hardwired to 0s.
23	0h RO	**Bus Power/Clock Control Enable (BPCCE):** Does not apply. Hardwired to 0.
22	0h RO	**B2/B3 Support (B23):** Does not apply. Hardwired to 0.
21:16	0h RO	Reserved
15	0h RW/1C/V	**PME Status (PMES):** This bit is set when the Intel HD Audio controller would normally assert the PME# signal independent of the state of the PME bit. This bit is cleared on a power-on reset. Software must not make assumptions about the reset state of this bit and must set it appropriately.
14:9	0h RO	Reserved
8	0h RW	**PME Enable (PMEE):** When set, and if corresponding PMES is also set, the Intel HD Audio subsystem send PME to PMC. This bit is cleared on a power-on reset. Software must not make assumptions about the reset state of this bit and must set it appropriately.
7:4	0h RO	Reserved
3	1h RW/L	**No Soft Reset (NSR):** When set (1), this bit indicates that devices transitioning from $D3_{HOT}$ to D0 because of PowerState commands do not perform an internal reset. Configuration Context is preserved. Upon transition from the $D3_{HOT}$ to the D0 Initialized state, no additional operating system intervention is required to preserve Configuration Context beyond writing the PowerState bits. When clear (0), devices do perform an internal reset upon transitioning from $D3_{HOT}$ to D0 via software control of the PowerState bits. Configuration Context is lost when performing the soft reset. Upon transition from the $D3_{HOT}$ to the D0 state, full reinitialization sequence is needed to return the device to D0 Initialized. Regardless of this bit, devices that transition from $D3_{HOT}$ to D0 by a system or bus segment reset will return to the device state D0 Uninitialized with only PME context preserved if PME is supported and enabled. Locked when FNCFG.BCLD = 1.
2	0h RO	Reserved

Bit Range	Default and Access	Field Name (ID): Description
1:0	0h RW	**Power State (PS):** This field is used both to determine the current power state of the Intel HD Audio subsystem and to set a new power state. The values are: 00 D0 state 11 $D3_{HOT}$ state If software attempts to write a value of 10b or 01b in to this field, the write operation must complete normally, however, the data is discarded and no state change occurs. When in the $D3_{HOT}$ states, the Intel HD Audio subsystem s configuration space is available, but the I/O and memory spaces are not. Additionally, interrupts are blocked. When software changes this value from the $D3_{HOT}$ state to the D0 state, an internal warm (soft) reset is generated, and software must re-initialize the function.

15.8.26 MSI Capability ID (MID)—Offset 60h

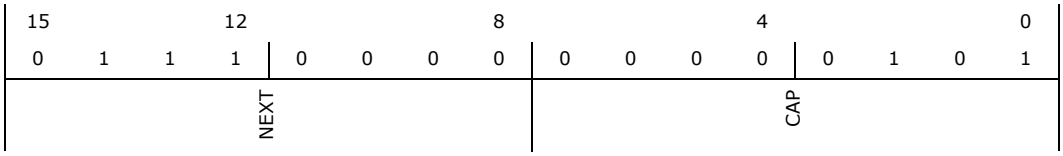
NEXT field is not affected by $D3_{HOT}$ to D0 reset or FLR.

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 3

Default: 7005h



Bit Range	Default and Access	Field Name (ID): Description
15:8	70h RO/V	**Next Capability (NEXT):** Points to the PCI Express capability structure. The value of this field depends on the FNCFG.HDASPCID bit. When FNCFG.HDASPCID is 0, this field has a value of 70h where it points to the PCI Express capability structure. When FNCFG.HDASPCID bit is 1, this field has a value of 00h to indicate that this is the last capability structure in the list.
7:0	5h RO	**Cap ID (CAP):** Indicates that this pointer is a MSI capability.

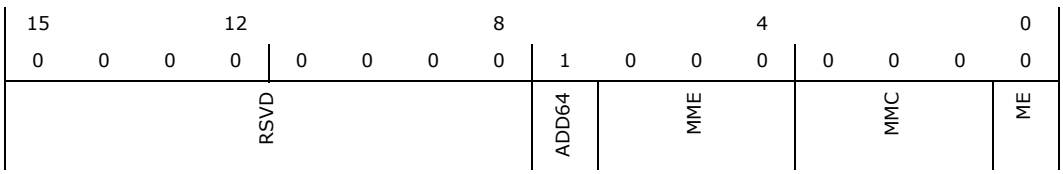
15.8.27 Message Signal Interrupt Message Control (MMC)—Offset 62h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 3

Default: 80h



Bit Range	Default and Access	Field Name (ID): Description
15:8	0h RO	Reserved
7	1h RO	**64b Address Capability (ADD64):** RO. Hardwired to 1. Indicates the ability to generate a 64-bit message address.
6:4	0h RO	**Multiple Message Enable (MME):** Normally this is a R/W register. However, since only 1 message is supported, these bits are hardwired to 000 = 1 message.
3:1	0h RO	**Multiple Message Capable (MMC):** Hardwired to 0 indicating request for 1 message.
0	0h RW	**MSI Enable (ME):** R/W. 0 = An MSI may not be generated. 1 = an MSI will be generated instead of an INTx signal.

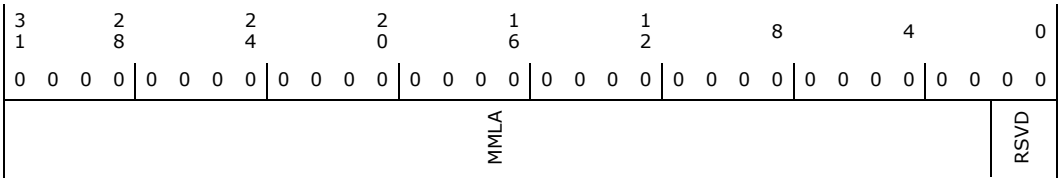
15.8.28 MSI Message Lower Address (MMLA)—Offset 64h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:2	0h RW	**MSI Message Lower Address (MMLA):** Lower Address used for MSI Message.
1:0	0h RO	Reserved

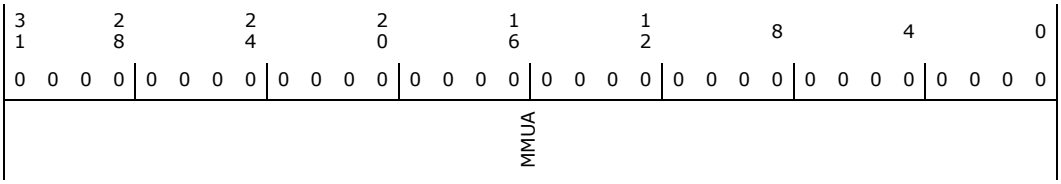
15.8.29 MSI Message Upper Address (MMUA)—Offset 68h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**MSI Message Upper Address (MMUA):** Upper 32 bits of address used for MSI Message.

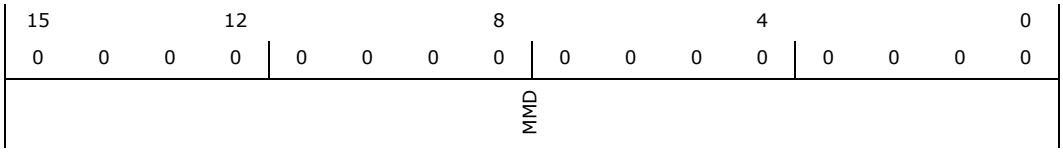
15.8.30 Message Signal Interrupt Capability ID Register (MMD)—Offset 6Ch

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:0	0h RW	**MSI Message Data (MMD):** Data used for MSI Message.

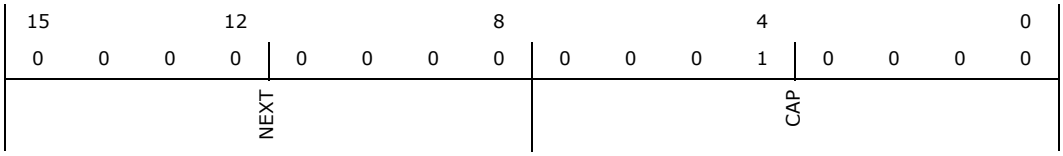
15.8.31 PCI Express Capability ID (PXID)—Offset 70h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 3

Default: 10h



Bit Range	Default and Access	Field Name (ID): Description
15:8	0h RO	**Next Capability (NEXT):** Indicates that this is the last capability structure in the list.
7:0	10h RO	**Cap ID (CAP):** Indicates that this pointer is a PCI Express capability structure.

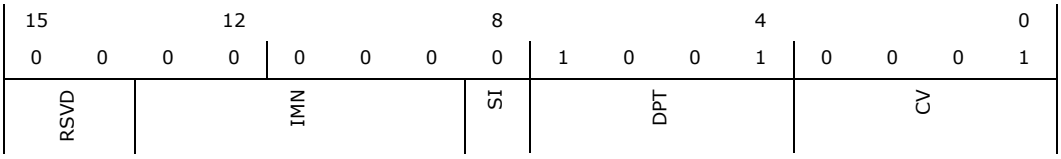
15.8.32 PCI Express Capabilities (PXC)—Offset 72h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 3

Default: 91h



Bit Range	Default and Access	Field Name (ID): Description
15:14	0h RO	Reserved
13:9	0h RO	**Interrupt Message Number (IMN):** Hardwired to 0.
8	0h RO	**Slot Implemented (SI):** Hardwired to 0.
7:4	9h RO	**Device/Port Type (DPT):** Indicates that this is a Root Complex Integrated Endpoint Device.
3:0	1h RO	**Capability Version (CV):** Indicates version #1 PCI Express capability.

15.8.33 Device Capabilities (DEVCAP)—Offset 74h

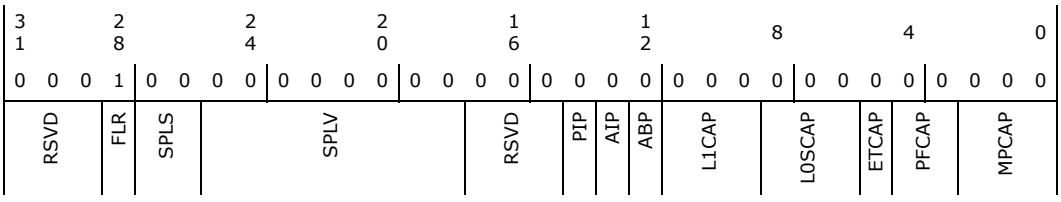
This register is not affected by $D3_{HOT}$ to D0 reset or FLR.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 3

Default: 10000000h



Bit Range	Default and Access	Field Name (ID): Description
31:29	0h RO	Reserved
28	1h RW/L	**Functional Level Reset (FLR):** A 1 indicates that the Intel HD Audio subsystem supports the Function Level Reset capability. Locked when FNCFG.BCLD = 1.
27:26	0h RO	**Captured Slot Power Limit Scale (SPLS):** Hardwired to 0.
25:18	0h RO	**Captured Slot Power Limit Value (SPLV):** Hardwired to 0.
17:15	0h RO	Reserved
14	0h RO	**Power Indicator Present (PIP):** Hardwired to 0.
13	0h RO	**Attention Indicator Present (AIP):** Hardwired to 0.
12	0h RO	**Attention Button Present (ABP):** Hardwired to 0.
11:9	0h RW/L	**Endpoint L1 Acceptable Latency (L1CAP):** This bit field is defined in the PCI Express Specification as RO. At this time it is risky to assign a hardwired value to this bit field. Making it RW would cause a WHQL failure. By making it RW/L it will appear as RO to WHQL testing while allowing the BIOS to write a value at boot that is determined by post silicon system testing. Locked when FNCFG.BCLD = 1.
8:6	0h RW/L	**Endpoint L0s Acceptable Latency (L0SCAP):** This bit field is defined in the PCI Express Specification as RO. At this time it is risky to assign a hardwired value to this bit field. Making it RW would cause a WHQL failure. By making it RW/L it will appear as RO to WHQL testing while allowing the BIOS to write a value at boot that is determined by post silicon system testing. Locked when FNCFG.BCLD = 1.
5	0h RO	**Extended Tag Field Support (ETCAP):** Indicates 5 bit tag supported.
4:3	0h RO	**Phantom Functions Supported (PFCAP):** Indicates phantom functions not supported.
2:0	0h RO	**Max Payload Size Supported (MPCAP):** Indicates 128B maximum payload size capability.

15.8.34 Device Control (DEVC)—Offset 78h

NSNPEN bit is not affected by $D3_{HOT}$ to D0 reset or FLR.

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 3

Default: 2800h

15			12				8				4				0
0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0
IF	MRRS			NSNPEN	AUXPEN	PFEN	ETEN	MAXPAY			ROEN	URREN	FEREN	NFEREN	CEREN

Bit Range	Default and Access	Field Name (ID): Description
15	0h WO	**Initiate FLR (IF):** Used to initiate FLR transition. A write of 1 initiates FLR transition. Since hardware must not respond to any cycles until FLR completion, the read value by software from this bit is 0.
14:12	2h RW	**Max Read Request Size (MRRS):** This field sets the maximum Read Request size for the Function as a Requester. The Function must not generate Read Requests with size exceeding the set value. Defined encodings for this field are: 000: 128 B 001: 256 B 010: 512 B 011: 1024 B 100: 2048 B 101: 4096 B 110 â?? 111: Reserved
11	1h RW	**Enable No Snoop (NSNPEN):** When set to 1 (or EM2.FNSNPEN = 1) the Intel HD Audio controller is permitted to set the No Snoop bit in the Requester Attributes of a bus master transaction. In this case VC0, VCp, or VC1 may be used for isochronous transfers. When set to 0 (and EM2.FNSNPEN = 0) the Intel HD Audio controller will not set the No Snoop bit. In the case isochronous transfers will not use VC1(VCi) even if it is enabled since VC1 is never snooped. Isochronous transfers will use either VCp or VC0. This bit is not affected by $D3_{HOT}$ to D0 reset or FLR.
10	0h RO	**Auxiliary (AUX) Power PM Enable (AUXPEN):** Hardwired to 0 indicating Intel HD Audio device does not draw AUX power.
9	0h RO	**Phantom Functions Enable (PFEN):** Hardwired to 0 disabling phantom functions.
8	0h RO	**Extended Tag Field Enable (ETEN):** Hardwired to 0 enabling 5-bit tag.
7:5	0h RO	**Max Payload Size (MAXPAY):** Hardwired to 000 indicating 128 B.
4	0h RO	**Enable Relaxed Ordering (ROEN):** Hardwired to 0 disabling relaxed ordering.
3	0h RW	**Unsupported Request Reporting Enable (URREN):** Functionality not implemented. This bit is R/W to pass PCIe compliance testing.
2	0h RW	**Fatal Error Reporting Enable (FEREN):** Functionality not implemented. This bit is R/W to pass PCIe compliance testing.
1	0h RW	**Non-Fatal Error Reporting Enable (NFEREN):** Functionality not implemented. This bit is R/W to pass PCIe compliance testing.
0	0h RW	**Correctable Error Reporting Enable (CEREN):** Functionality not implemented. This bit is R/W to pass PCIe compliance testing.

15.8.35 Device Status (DEVS)—Offset 7Ah

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 3

Default: 10h

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
RSVD										TXP	AUXDET	URDET	FEDET	NFEDET	CEDET

Bit Range	Default and Access	Field Name (ID): Description
15:6	0h RO	Reserved
5	0h RO/V	**Transactions Pending (TXP):** A 1 indicates that the Intel HD Audio controller has issued Non-Posted requests which have not been completed. A 0 indicates that Completions for all Non-Posted Requests have been received.
4	1h RW/L	**AUX Power Detected (AUXDET):** Hardwired to 1 indicating the device is connected to Suspend power. Programmable by BIOS for the option to declare SUS well wake is supported or not: 1b (SUS well wake supported) or 0b (SUS well wake not supported). Locked when FNCFG.BCLD = 1.
3	0h RO	**Unsupported Request Detected (URDET):** Not implemented. Hardwired to 0.
2	0h RO	**Fatal Error Detected (FEDET):** Not implemented. Hardwired to 0.
1	0h RO	**Non-Fatal Error Detected (NFEDET):** Not implemented. Hardwired to 0.
0	0h RO	**Correctable Error Detected (CEDET):** Not implemented. Hardwired to 0.

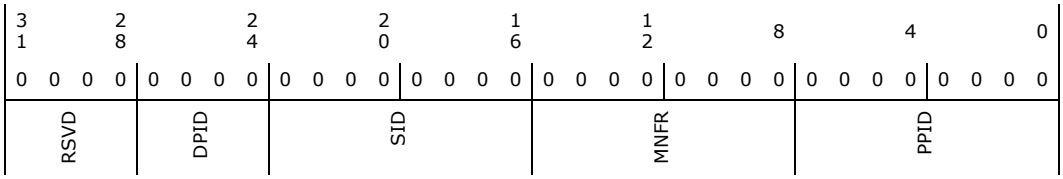
15.8.36 Manufacturing Process (MANID)—Offset F8h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:28	0h RO	Reserved
27:24	0h RO/V	**Dot Portion Process ID (DPID):** Indicating the dot portion of the process ID. IOSF Sideband Set ID Value message initializes this register value.
23:16	0h RO/V	**Stepping ID (SID):** This field is incremented for each stepping of the part. Note that this field can be used by software to differentiate stepping when the Revision ID may not change. IOSF Sideband Set ID Value message initializes this register value.
15:8	0h RO/V	**Manufacturer (MNFR):** Indicating the manufacturer ID. IOSF Sideband Set ID Value message initializes this register value.
7:0	0h RO/V	**Process Portion Process ID (PPID):** Indicating the process portion of the process ID. IOSF Sideband Set ID Value message initializes this register value.

15.8.37 Virtual Channel Enhanced Capability Header (VCCAP)—Offset 100h

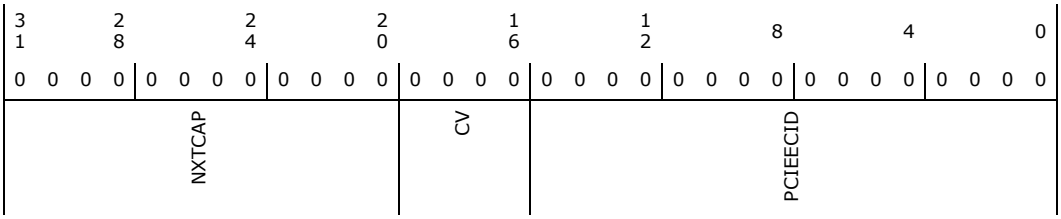
This register is not affected by $D3_{HOT}$ to D0 reset or FLR.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:20	0h RW/L	**Next Capability Offset (NXTCAP):** Points to the next capability header, which is the Root Complex Link Declaration Enhanced Capability Header. This register is RW/L to support removing the Root Complex Topology Capability from the PCI Express Extended Capability List. For systems which support the Root Complex Topology Capability Structure, boot BIOS should write a 130h to this register, otherwise boot BIOS should write a 000h to this register. Locked when FNCFG.BCLD = 1.
19:16	0h RW/L	**Capability Version (CV):** This register is RW/L to support removing the PCI Express Extended Capabilities from the Intel HD Audio subsystem. For systems which support the PCI Express Virtual Channel capability and the Root Complex Topology Capability Structure, boot BIOS should write a 1h to this register, otherwise boot BIOS should write a 0h to this register. Locked when FNCFG.BCLD = 1.
15:0	0h RW/L	**PCI Express Extended Capability ID (PCIEECID):** This register is RW/L to support removing the PCI Express Extended Capabilities from the Intel HD Audio subsysterm. For systems which support the PCI Express Virtual Channel capability and the Root Complex Topology Capability Structure, boot BIOS should write a 0002h to this register, otherwise boot BIOS should write a 0000h to this register. Locked when FNCFG.BCLD = 1.

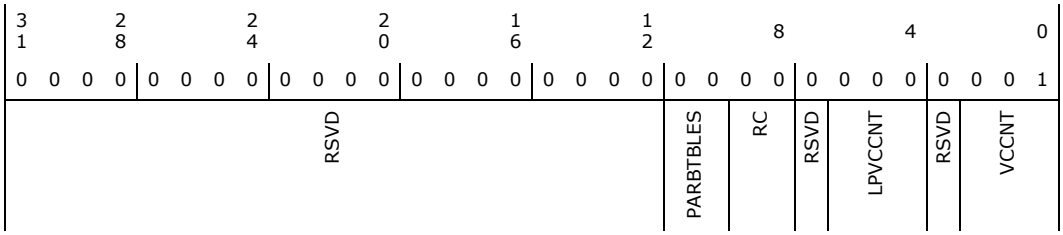
15.8.38 Port VC Capability Register 1 (PVCCAP1)—Offset 104h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 3

Default: 1h



Bit Range	Default and Access	Field Name (ID): Description
31:12	0h RO	Reserved
11:10	0h RO	**Port Arbitration Table Entry Size (PARBTBLES):** This is an endpoint device therefore this field is hardwired to 0s.
9:8	0h RO	**Reference Clock (RC):** This is an endpoint device therefore this field is hardwired to 0s.
7	0h RO	Reserved.
6:4	0h RO	**Low Priority Extended VC Count (LPVCCNT):** Indicates that only VC0 belongs to the low-priority VC group.
3	0h RO	Reserved
2:0	1h RO	**Extended VC Count (VCCNT):** Indicates that one extended VC (in addition to VC0) is supported by the Intel(r) HD Audio controller.

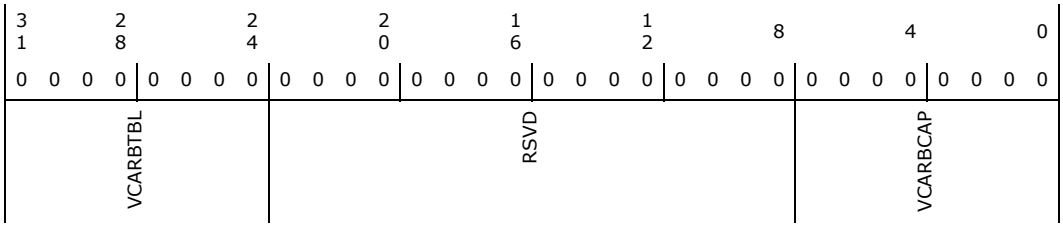
15.8.39 Port VC Capability Register 2 (PVCCAP2)—Offset 108h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	**VC Arbitration Table Offset (VCARBTBL):** Hardwired to 0 indicating that a VC Arbitration Table is not present.
23:8	0h RO	Reserved
7:0	0h RO	**VC Arbitration Capability (VCARBCAP):** Hardwired to 0 s. These bits are not applicable since the Intel HD Audio controller reports a 0 in the Low Priority Extended VC Count field.

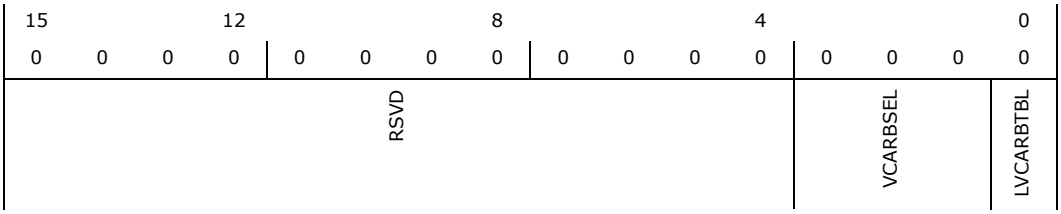
15.8.40 Port VC Control Register (PVCCTL)—Offset 10Ch

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:4	0h RO	Reserved
3:1	0h RO	**VC Arbitration Select (VCARBSEL):** Hardwired to 0s. Normally these bits are RW. But these bits are not applicable since the Intel HD Audio controller reports a 0 in the Low Priority Extended VC Count field.
0	0h RO	**Load VC Arbitration Table (LVCARBTBL):** Hardwired to 0 since an arbitration table is not present.

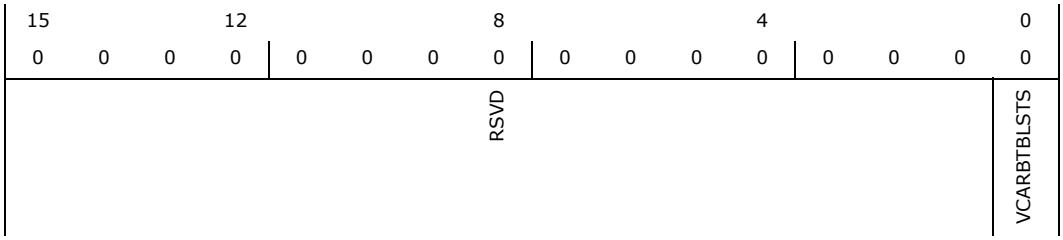
15.8.41 Port VC Status Register (PVCSTS)—Offset 10Eh

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:1	0h RO	Reserved
0	0h RO	**VC Arbitration Table Status (VCARBTBLSTS):** Hardwired to 0 since the VC Arbitration Table is not present.

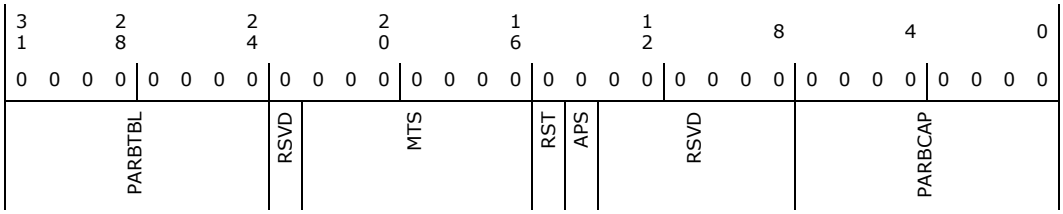
15.8.42 VC0 Resource Capability Register (VC0CAP)—Offset 110h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	**Port Arbitration Table Offset (PARBTBL):** Hardwired to 0 since this field is not valid for endpoint devices.
23	0h RO	Reserved
22:16	0h RO	**Maximum Time Slots (MTS):** Hardwired to 0 since this field is not valid for endpoint devices.
15	0h RO	**Reject Snoop Transactions (RST):** Hardwired to 0 since this field is not valid for endpoint devices.

Bit Range	Default and Access	Field Name (ID): Description
14	0h RO	**Advanced Packet Switching (APS):** Hardwired to 0 since this field is not valid for endpoint devices.
13:8	0h RO	Reserved
7:0	0h RO	**Port Arbitration Capability (PARBCAP):** Hardwired to 0 since this field is not valid for endpoint devices.

15.8.43 VC0 Resource Control Register (VC0CTL)—Offset 114h

VC0MAP(7:1) field is not affected by FLR.

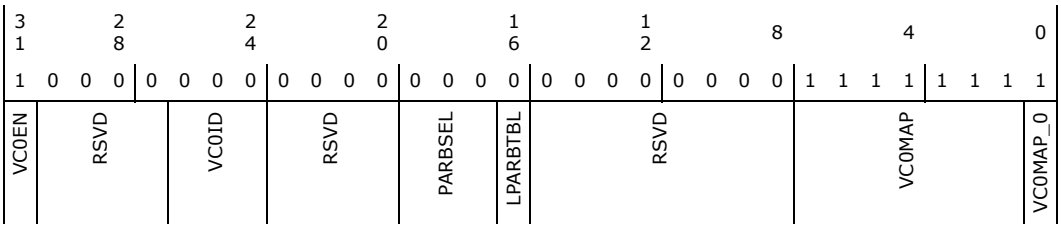
Implementation Notes: Due to legacy implementation of the VC0MAP field being implemented as reset by $D3_{HOT}$ to D0 Reset but not FLR and already work well with existing SW (and there is no clear definition in the PCI and PCIe Specification whether it should be reset by $D3_{HOT}$ to D0 Reset or not), this legacy implementation remains no change and documented here as reset by D3ONLY, which is a deviation from D3RST definition by excluding FLR.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 3

Default: 800000FFh



Bit Range	Default and Access	Field Name (ID): Description
31	1h RO	**VC0 Enable (VC0EN):** Hardwired to 1 for VC0.
30:27	0h RO	Reserved
26:24	0h RO	**VC0 ID (VC0ID):** Hardwired to 0 since the first VC is always assigned as VC0.
23:20	0h RO	Reserved
19:17	0h RO	**Port Arbitration Select (PARBSEL):** Hardwired to 0 since this field is not valid for endpoint devices.
16	0h RO	**Load Port Arbitration Table (LPARBTBL):** Hardwired to 0 since this field is not valid for endpoint devices.

Bit Range	Default and Access	Field Name (ID): Description
15:8	0h RO	Reserved
7:1	7Fh RW	**TC/VC0 Map (VC0MAP):** Bits [7:1] are implemented as R/W bits. This field is not used by the hardware, but it is RW to avoid confusing software.
0	1h RO	**TC/VC0 Map (VC0MAP_0):** Bit 0 is hardwired to 1 since TC0 is always mapped to VC0.

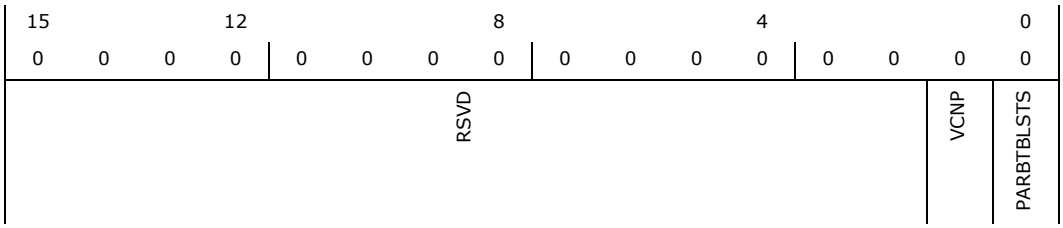
15.8.44 VC0 Resource Status Register (VC0STS)—Offset 11Ah

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:2	0h RO	Reserved
1	0h RO	**VC0 Negotiation Pending (VCNP):** This bit does not apply to the integrated Intel HD Audio device and is therefore hardwired to 0.
0	0h RO	**Port Arbitration Table Status (PARBTBLSTS):** Hardwired to 0 since this field is not valid for endpoint devices.

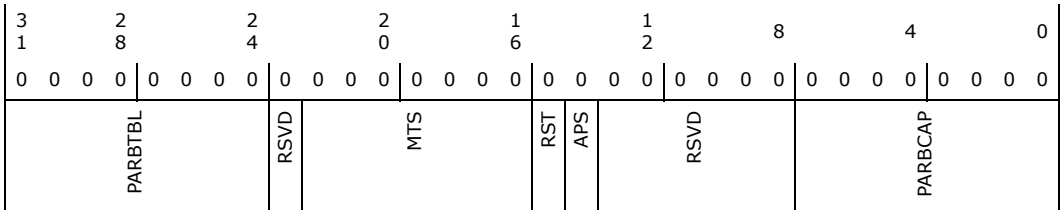
15.8.45 VCi Resource Capability Register (VCiCAP)—Offset 11Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	**Port Arbitration Table Offset (PARBTBL):** Hardwired to 0 since this field is not valid for endpoint devices.
23	0h RO	Reserved
22:16	0h RO	**Maximum Time Slots (MTS):** Hardwired to 0 since this field is not valid for endpoint devices.
15	0h RO	**Reject Snoop Transactions (RST):** Hardwired to 0 since this field is not valid for endpoint devices.
14	0h RO	**Advanced Packet Switching (APS):** Hardwired to 0 since this field is not valid for endpoint devices.
13:8	0h RO	Reserved
7:0	0h RO	**Port Arbitration Capability (PARBCAP):** Hardwired to 0 since this field is not valid for endpoint devices.

15.8.46 VCi Resource Control Register (VCiCTL)—Offset 120h

VCiEN bit and VCiID field are not affected by $D3_{HOT}$ to D0 Reset or FLR. VCiM(7:1) field is not affected by FLR.

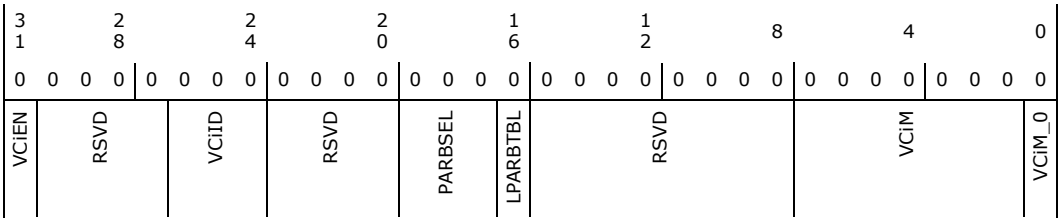
Implementation Notes: Due to legacy implementation of the VCiM field being implemented as reset by $D3_{HOT}$ to D0 Reset but not FLR and already work well with existing SW (and there is no clear definition in the PCI and PCIe Specification whether it should be reset by $D3_{HOT}$ to D0 Reset or not), this legacy implementation remains no change and documented here as reset by D3ONLY, which is a deviation from D3RST definition by excluding FLR.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RW	**VCi Enable (VCiEN):** When set to 1, VCi is enabled. This bit is not affected by $D3_{HOT}$ to D0 reset.
30:27	0h RO	Reserved
26:24	0h RW	**VCi ID (VCiID):** This fields assigns a VC ID to the VCi resource. This field is not used by the PCH hardware, but it is RW to avoid confusing software. These bits are not affected by $D3_{HOT}$ to D0 reset.
23:20	0h RO	Reserved
19:17	0h RO	**Port Arbitration Select (PARBSEL):** Hardwired to 0 since this field is not valid for endpoint devices.
16	0h RO	**Load Port Arbitration Table (LPARBTBL):** Hardwired to 0 since this field is not valid for endpoint devices.
15:8	0h RO	Reserved
7:1	0h RW	**TC/VCi Map (VCiM):** This field indicates the TCs that are mapped to the VCi resource. Bits [7:1] are implemented as RW bits. This field is not used by the hardware, but it is RW to avoid confusing software.
0	0h RO	**TC/VCi Map (VCiM_0):** This field indicates the TCs that are mapped to the VCi resource. Bit 0 is hardwired to 0 indicating it can not be mapped to VCi. This field is not used by the hardware.

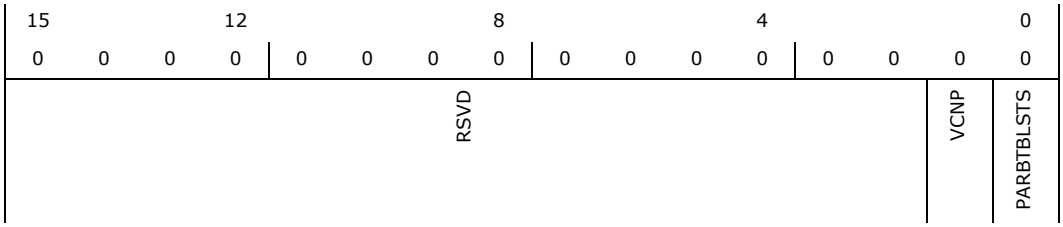
15.8.47 VCi Resource Status Register (VCiSTS)—Offset 126h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:2	0h RO	Reserved
1	0h RO	**VCi Negotiation Pending (VCNP):** This bit does not apply to the integrated Intel HD Audio subsystem and is therefore hardwired to 0.
0	0h RO	**Port Arbitration Table Status (PARBTBLSTS):** Hardwired to 0 since this field is not valid for endpoint devices.

15.8.48 Root Complex Link Declaration Enhanced (RCCAP)—Offset 130h

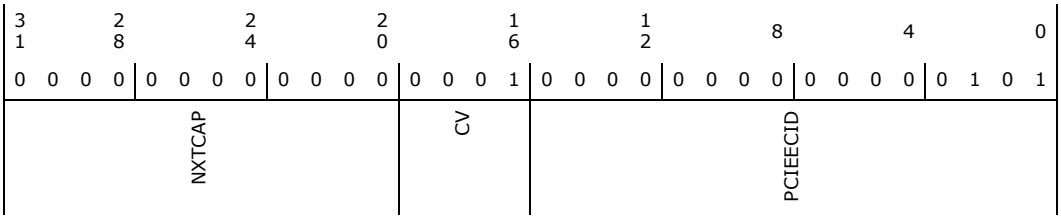
Capability Header

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 3

Default: 10005h



Bit Range	Default and Access	Field Name (ID): Description
31:20	0h RO	**Next Capability Offset (NXTCAP):** Indicates this is the last capability.
19:16	1h RO	**Capability Version (CV)**
15:0	5h RO	**PCI Express Extended Capability ID (PCIEECID)**

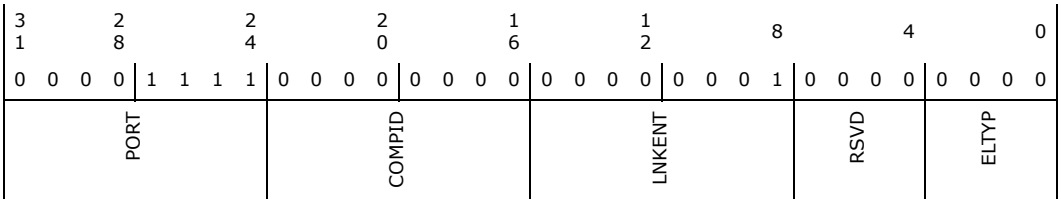
15.8.49 Element Self Description (ESD)—Offset 134h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 3

Default: F000100h



Bit Range	Default and Access	Field Name (ID): Description
31:24	Fh RO	**Port Number (PORT):** Intel HD Audio assigned as Port #15.
23:16	0h RW/L	**Component ID (COMPID):** Indicates the component ID assigned to this element by software. This is written once by platform BIOS and is locked until a platform reset. Locked when FNCFG.BCLD = 1.
15:8	1h RO	**Number of Link Entries (LNKENT):** The Intel HD Audio controller only connects to one device, the SOC egress port. Therefore this field reports a value of 1.
7:4	0h RO	Reserved
3:0	0h RO	**Element Type (ELTYP):** The Intel HD Audio controller is an Integrated Root Complex Device. Therefore this field reports a value of 0h.

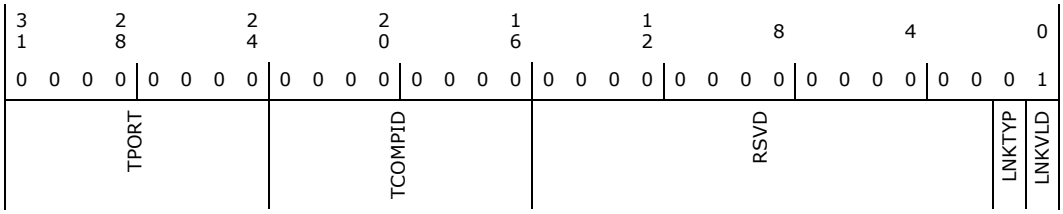
15.8.50 Link 1 Description (L1DESC)—Offset 140h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 3

Default: 1h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	**Target Port Number (TPORT):** The Intel HD Audio controller targets the SOC RCRB egress port, which is port #0.
23:16	0h RW/L	**Target Component ID (TCOMPID):** This field is programmed by platform BIOS to match the component ID of the SOC RCRB that is attached to this RCRB Locked when FNCFG.BCLD = 1.
15:2	0h RO	Reserved
1	0h RO	**Link Type (LNKTYP):** Indicates Type 0.
0	1h RO	**Link Valid (LNKVLD):** Hardwired to 1.

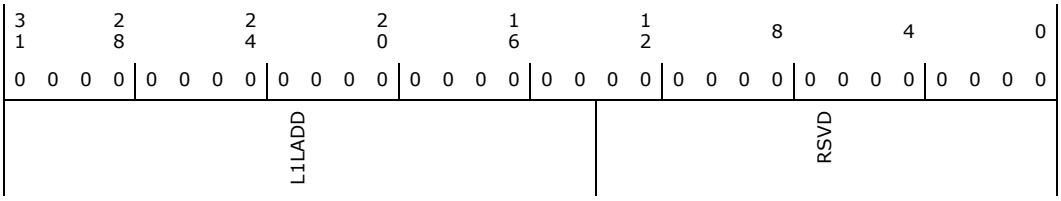
15.8.51 Link 1 Lower Address (L1LADD)—Offset 148h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:14	0h RO/V	**Link 1 Lower Address (L1LADD):** Hardwired to match the RCBA register value in the PCI-LPC bridge.
13:0	0h RO	Reserved

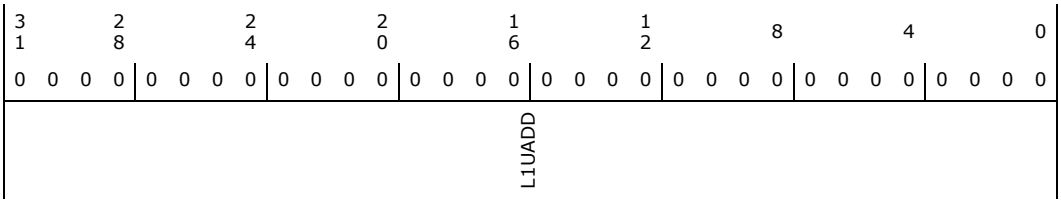
15.8.52 Link 1 Upper Address (L1UADD)—Offset 14Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RO/V	**Link 1 Upper Address (L1UADD):** Hardwired to match the RCBA register value in the PCI-LPC bridge.

§

16 Processor Interface

16.1 Acronyms

Acronyms	Description
PECI	Platform Environmental Control Interface

16.2 Overview

The sideband signals are used for the communication between the processor and PCH.

16.3 Signal Description

Name	Type	Description
PROC_PWRGD	O	Signal to the processor to indicate its primary power is good.
THRMTRIP#	I	Signal from the processor to indicate that thermal overheating has occurred.
PECI	I/O	Single-wire serial bus for accessing processor digital thermometer.
PM_SYNC	O	Power Management Sync: State exchange from the PCH to the Processor
PM_SYNC2	O	Power Management Sync: State exchange from the PCH to the Processor. Use this pin when there are more than 2 sockets.
PLTRST_CPU_N	O	Platform reset to processor.
GPP_E3_CPU_GP0	I	PM_SYNC CPU GPIO Signal
GPP_E7_CPU_GP1	I	PM_SYNC CPU GPIO Signal
GPP_B3_CPU_GP2	I	PM_SYNC CPU GPIO Signal
GPP_B4_CPU_GP3	I	PM_SYNC CPU GPIO Signal

16.4 Integrated Pull-Ups and Pull-Downs

None.

16.5 I/O Signal Planes and States

Signal Name	Power Plane	During Reset	Immediately after Reset	S3/S4/S5	Deep Sx
PROC_PWRGD	Primary	Driven Low	Driven High	Off	Off
THERMTRIP#	Primary	Undriven	Undriven	Off	Off
PECI	Primary	Undriven	Undriven	Off	Off
PM_SYNC	Primary	Driven Low	Driven Low	Off	Off
PM_SYNC2	Primary	Driven Low	Driven Low	Off	Off
PLTRST_CPU_N	Primary	Driven Low	Driven High	Off	Off
CPU_GP[3:0]	Primary	Undriven	Undriven	Undriven	Off

16.6 Functional Description

PROC_PWRGD out to the processor indicates that the primary power is ramped up and stable.

If THERMTRIP# goes active, the CPU is indicating an overheat condition, and the PCH will immediately transition to an S5 state.

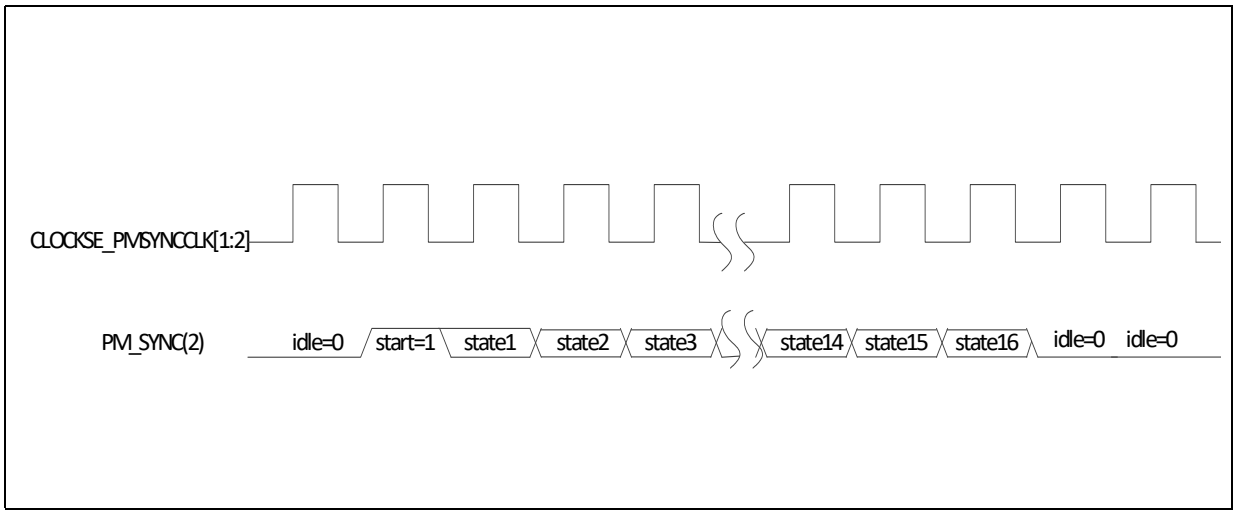
The CPU_GP[3:0] are signals whose states are reflected in the certain bits in the message stream of the PM_SYNC signals connecting the processors. They provide an easy way of transmitting a certain state or information to all CPUs simultaneously. CPU_GP0 can be used to trigger ADR. There are 4 GPIOs in the PM_Sync message. Each CPU_GP can be mapped to 2 of the GPIOS.

PM_SYNC and PM_SYNC2 are used to provide an early warning to the processor that a global reset is in progress and that the memory contents should be saved and placed in self refresh.

The PM_SYNC and PM_SYNC2 signals are just copies of each other. Because of the loading on these pins, they can not drive to more than two CPUs. The same for CLOCKSE_PMSYNCCLK1 and CLOCKSE_PMSYNCCLK2. There is also no relationship between any CLOCKSE_PMSYNCCLK and PM_SYNC signal. CLOCKSE_PMSYNCCLK1 can go with either PM_SYNC or PM_SYNC2, and the same can be said for CLOCKSE_PMSYNCCLK2

Traffic on PM_SYNC is unidirectional, it only goes from the PCH to the CPU. There is no communication from CPU to PCH over PM_SYNC. When the bus is idle, PM_SYNC is low. The start of any communication is having PM_SYNC be clocked in as high. Next are the 16 bits of information. At the end of this, if a "1" is clocked in, another 16 bit of information are coming. If the bus is going to idle, then PM_SYNC needs to be low for 2 clocks before it can go high again.

Figure 16-1. PM_SYNC protocol



PLTRST_CPU_N is the platform reset to the processor. This is a low voltage signal that only drives a single processor.

16.7 Processor Interface Memory Registers Summary

Table 16-1. Summary of Processor Interface Memory Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
61h	61h	NMI Status and Control (NMI_STS_CNT)—Offset 61h	0h
70h	70h	NMI Enable (and Real Time Clock Index) (NMI_EN)—Offset 70h	80h
92h	92h	Init Register (PORT92)—Offset 92h	0h
CF9h	CF9h	Reset Control Register (RST_CNT)—Offset CF9h	0h

16.7.1 NMI Status and Control (NMI_STS_CNT)—Offset 61h

Access Method

Type: I/O Register
(Size: 8 bits)

Device:
Function:

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
SERR_NMI_STS	IOCHK_NMI_STS	TMR2_OUT_STS	RSVD	IOCHK_NMI_EN	PCI_SERR_EN	SPKR_DAT_EN	TIM_CNT2_EN

Bit Range	Default and Access	Field Name (ID): Description
7	0h RO	**SERR# NMI Source Status (SERR_NMI_STS):** This bit is set by any of the sources of the internal SERR on the PCH backbone, this includes SERR assertions forwarded from the secondary PCI bus, error from a PCIe port, Do_SERR or standard PCIe error message from DMI, or internal Bus 0 functions that generate SERR#. Bit 2 must be cleared in this register in order for this bit to be set. This interrupt source is enabled by setting bit 2 to 0. To reset the interrupt, set bit 2 to 1 and then set it to 0. This bit is read-only. When writing to port 61h, this bit must be 0.
6	0h RO	**IOCHK# NMI Source Status (IOCHK_NMI_STS):** This bit is set if an ISA agent (via SERIRQ) asserts IOCHK# and bit 3 is cleared in this register. This interrupt source is enabled by setting bit 3 to 0. To reset the interrupt, set bit 3 to 1 and then set it to 0. This bit is read-only. When writing to port 61h, this bit must be a 0.
5	0h RO	**Timer Counter 2 OUT Status (TMR2_OUT_STS):** This bit reflects the current state of the 8254 counter 2 output. Counter 2 must be programmed following any PCI reset for this bit to have a determinate value. When writing to port 61h, this bit must be a 0.
4	0h RO	Reserved
3	0h RW	**IOCHK# NMI Enable (IOCHK_NMI_EN):** When this bit is a 1, IOCHK# NMIs are disabled and cleared. When this bit is a 0, IOCHK# NMIs are enabled.

Bit Range	Default and Access	Field Name (ID): Description
2	0h RW	**PCI SERR# Enable (PCI_SERR_EN):** When this bit is a 1, the SERR# NMIs are disabled and cleared. When this bit is a 0, SERR# NMIs are enabled.
1	0h RW	**Speaker Data Enable (SPKR_DAT_EN):** When this bit is a 0, the SPKR output is a 0. When this bit is a 1, the SPKR output is equivalent to the Counter 2 OUT signal value.
0	0h RW	**Timer Counter 2 Enable (TIM_CNT2_EN):** When this bit is a 0, Counter 2 counting is disabled. Counting is enabled when this bit is 1.

16.7.2 NMI Enable (and Real Time Clock Index) (NMI_EN)—Offset 70h

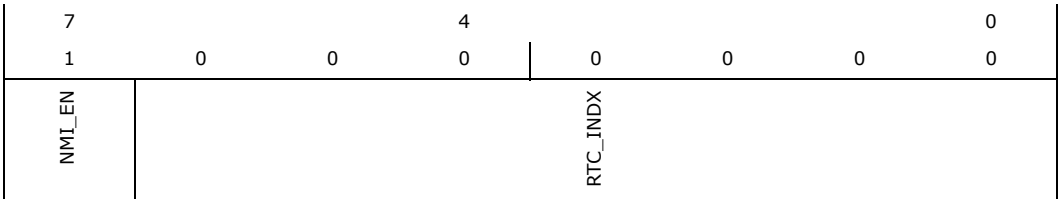
This register is write-only for normal operation. In Alt-Access mode, this register can be read to find the NMI Enable status and the RTC index value. *WO for normal operation, RW if Alternate access mode is enabled. Use RW because there is no equivalent register access attribute in RDL.

Access Method

Type: I/O Register
(Size: 8 bits)

Device:
Function:

Default: 80h



Bit Range	Default and Access	Field Name (ID): Description
7	1h RW	**NMI_EN# (NMI_EN):** When this bit is a 1, all NMI sources are disabled. When this bit is a 0, NMI sources are enabled.
6:0	0h RW	**Real Time Clock Index (Address) (RTC_INDX):** This data goes to the RTC to select which register or CMOS RAM address is being accessed.

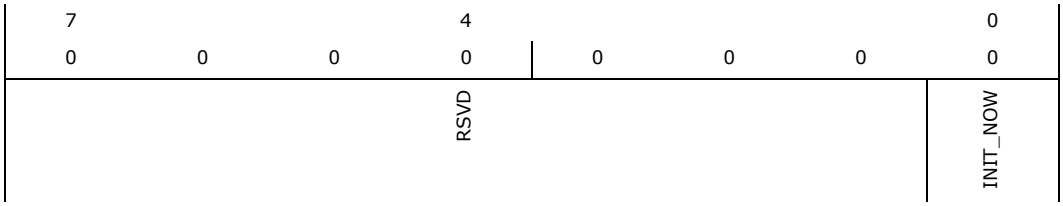
16.7.3 Init Register (PORT92)—Offset 92h

Access Method

Type: I/O Register
(Size: 8 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:1	0h RO	Reserved
0	0h RW	**INIT_NOW (INIT_NOW):** When this bit transitions from a 0 to a 1, the PCH will force INIT# active for 16 PCI clocks.

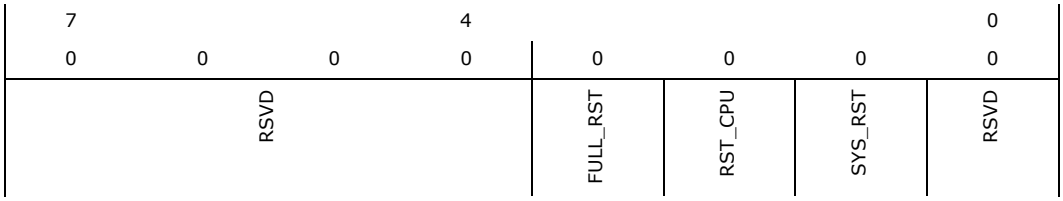
16.7.4 Reset Control Register (RST_CNT)—Offset CF9h

Access Method

Type: I/O Register
(Size: 8 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:4	0h RO	Reserved.
3	0h RW	**Full Reset (FULL_RST):** When this bit is set to 1 and bit 1 is set to 1 (indicating Hard Reset, not Soft Reset), and the RST_CPU bit (bit 2) is written from 0 to 1, the PCH will do a full reset, including driving SLP_S3#, SLP_S4# and SLP_S5# active (low) for at least 3 (and no more than 5) seconds. When this bit is set, it also causes the full power cycle (SLP_S3/4/5# assertion) in response to SYSRESET#, PWROK#, and Watchdog timer reset sources.

Bit Range	Default and Access	Field Name (ID): Description
2	0h RW	**Reset CPU (RST_CPU):** This bit will cause either a hard or soft reset to the CPU depending on the state of the SYS_RST bit (bit 1 in this same register). The software will cause the reset by setting bit 2 from a 0 to a 1.
1	0h RW	**System Reset (SYS_RST):** This bit determines the type of reset caused via RST_CPU (bit 2 of this register). If SYS_RST is 0 when RST_CPU goes from 0 to 1, then the PCH will force INIT# active for 16 PCI clocks. If SYS_RST is 1 when RST_CPU goes from 0 to 1, then the PCH will force PCI reset active for about 1 ms, however the SLP_S3#, SLP_S4# and SLP_S5# signals assertion is dependent on the value of the FULL_RST (bit3 of this register).
0	0h RO	Reserved

§

17 Enhanced Serial Peripheral Interface

17.1 Acronyms

Acronyms	Description
eSPI	Enhanced Serial Peripheral Interface
EC	Embedded Controller
MAFS	Master Attached Flash Sharing
MAFCC	Master Attached Flash Channel Controller
SAFS	Slave Attached Flash Sharing
SAFCC	Slave Attached Flash Channel Controller
SAFS	Slave Attached Flash Sharing
OOB	Out-of-Band
TAR	Turn-around cycle

17.2 References

Specification	Location
Enhanced Serial Peripheral Interface Specifications	https://downloadcenter.intel.com/Detail_Desc.aspx?agr=Y&DwnldID=22112

17.3 Overview

The PCH provides the eSPI to support connection of a BMC/EC/SIO to the platform.

The interface supports 1.8V only and connects to a maximum of two eSPI devices. This interface is not shared and distinct from the SPI bus interface used for flash device and TPM.

The LPC and eSPI interfaces are mutually exclusive. Both use the same pins, but on power-up, a HW strap determines if the eSPI or the LPC bus is operational. Once selected, it’s not possible to change to the other interface.

17.4 Signal Description

Name	Type	Description
GPP_A9_CLKOUT_LPC0_ESPI_CLK	O	**ESPI Clock:** eSPI clock output from the PCH to slave device.
GPP_A1_LAD0_ESPI_IO0	I/O	**ESPI Data Signal 0:** Bi-directional pin used to transfer data between the PCH and eSPI slave device.
GPP_A2_LAD1_ESPI_IO1	I/O	**ESPI Data Signal 1:** Bi-directional pin used to transfer data between the PCH and eSPI slave device
GPP_A3_LAD2_ESPI_IO2	I/O	**ESPI Data Signal 2:** Bi-directional pin used to transfer data between the PCH and eSPI slave device
GPP_A4_LAD3_ESPI_IO3	I/O	**ESPI Data Signal 3:** Bi-directional pin used to transfer data between the PCH and eSPI slave device
GPP_A5_LFRAME#_ESPI_CS_N	O	**ESPI Chip Select 0:** Driving CS# signal low to select eSPI slave for the transaction.
GPP_A6_SERIRQ_ESPI_CS1_N	O	**ESPI Chip Select 1:** Driving CS# signal low to select eSPI slave for the transaction.
GPP_A7_PIRQA#_ESPI_ALERT0_N	I	**ESPI Alert 0:** Alert signal from eSPI slave to the PCH. ***Note:*** If only a single Slave is connected, the eSPI Compatibility Specification requires that the Slave must operate with in-band Alert# signaling in order to free up the GPIO pin required for the discrete Alert# pin.
GPP_A0_RCIN#_ESPI_ALERT1_N	I	**ESPI Alert 1:** Alert signal from eSPI slave to the PCH. ***Note:*** If only a single Slave is connected, the eSPI Compatibility Specification requires that the Slave must operate with in-band Alert# signaling in order to free up the GPIO pin required for the discrete Alert# pin.
GPP_A14_ESPI_RESET_N	O	**ESPI Reset:** Reset signal from the PCH to eSPI slave.

17.5 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
ESPI_CLK	Pull-down	9K - 50K	
ESPI_IO[3:0]	Pull-up	15K - 40K	
ESPI_ CS [1:0]#	Pull-up	15K - 40K	
ESPI_ALERT [1:0]#	Pull-up	15K - 40K	
Note: Integrated pull-ups and pull-downs will not be valid until all the voltages have reached a valid level.			

17.6 I/O Signal Planes and States

Signal Name	Power Plane	During Reset	Immediately After Reset	S3/S4/S5	Deep Sx
ESPI_CLK	Primary	Internal Pull-down	Driven Low	Driven Low	Off
ESPI_IO [3:0]	Primary	Internal Pull-up	Internal Pull-up	Internal Pull-up	Off
ESPI_ CS [1:0]#	Primary	Internal Pull-up	Driven High	Driven High	Off
ESPI_ALERT [1:0]#	Primary	Internal Pull-up	Driven High	Driven High	Off
ESPI_ RESET#	Primary	Driven Low	Driven High	Driven High	Off

17.7 Functional Description

17.7.1 Features

The PCH eSPI controller supports the following features:

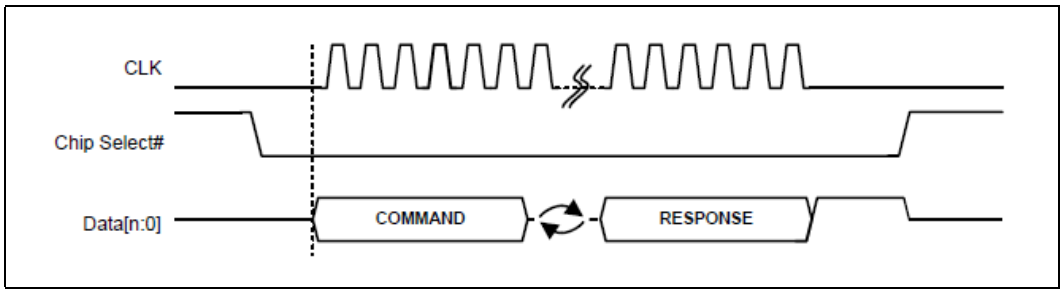
- Master mode only, allowing two slave devices to be connected to the PCH
- Master attached flash sharing and slave attached flash sharing.
- 1.8V support only
- Up to quad mode support
- In-band messages for communication between the PCH and slave devices to eliminate side-band signals
- Real time SPI flash sharing, allowing real time operational access by the PCH and eSPI device
- Transmitting RTC time/date to the slave device upon request.

he PCH eSPI supports two ALERT# pins to support alerts from two supported slave devices to the PCH. However, if only a single slave is connected, the eSPI Compatibility Specification requires that the Slave must operate with in-band Alert# signaling in order to free up the GPIO pin required for the discrete Alert# pin.

17.7.2 Protocols

Below is an overview of the basic eSPI protocol. Refer to the latest eSPI Specification for more detail (see Section 17.2).

Figure 17-1. Basic eSPI Protocol



An eSPI transaction consists of a Command phase driven by the master, a TAR phase, and a Response phase driven by the slave.

A transaction is initiated by the PCH through the assertion of CS#, starting the clock and driving the command onto the data bus. The clock remains toggling until the complete response phase has been received from the slave.

The serial clock must be low at the assertion edge of the CS# while ESPI_RESET# has been de-asserted. The first data is driven out from the PCH while the serial clock is still low and sampled on the rising edge of the clock by the slave. Subsequent data is driven on the falling edge of the clock from the PCH and sampled on the rising edge of the clock by the slave. Data from the slave is driven out on the falling edge of the clock and is sampled on a falling edge of the clock by the PCH.

All transactions on eSPI are in eight bit multiples.

17.7.3 WAIT States from eSPI Slave

There are situations when the slave cannot predict the length of the command packet from the master (PCH). For non-posted transactions, the slave is allowed to respond with a limited number of WAIT states.

A WAIT state is a one byte response code. They must be the first set of response bytes from the slave after the TAR cycles.

17.7.4 In-Band Link Reset

In case the eSPI link may end up in an undefined state (for example when a CRC error is received from the slave in a response to a Set_Configuration command), the PCH issues an In-Band Reset command which resets the eSPI link to the default configuration. This allows the controller to re-initialize the link and reconfigure the slave.

17.7.5 Slave Discovery

The PCH eSPI interface is enabled via a hard pin strap, ESPI_EN. If this strap is asserted (high) at RSMRST# deassertion, the eSPI controller is enabled and assumes that a slave is connected to the interface. The controller does not perform any other discovery to confirm the presence of the slave connection. There is a soft strap to indicate whether a second slave device is present or not.

If the ESPI_EN HW strap is de-asserted (low), the eSPI controller will gate all its clocks and put itself to sleep.

17.7.6 Channels and Supported Transactions

An eSPI channel provides a means to allow multiple independent flows of traffic to share the same physical bus. Refer to the eSPI specification for more detail.

Each of the channels has its dedicated resources such as queue and flow control. There is no ordering requirement between traffic from different channels.

The number of types of channels supported by a particular eSPI slave is discovered through the GET_CONFIGURATION command issued by the PCH to the eSPI slave during initialization.

Table 17-1 summarizes the eSPI channels and supported transactions.

Table 17-1. eSPI Channels and Supported Transactions

CH #	Channel	Posted Cycles Supported	Non-Posted Cycles Supported
0	Peripheral	Memory Write, Completions	Memory Read, I/O Read/Write
1	Virtual Wire	Virtual Wire GET/PUT	N/A
2	Out-of-Band Message	SMBus Packet GET/PUT	N/A
3	Flash Access	N/A	Flash Read, Write, Erase
N/A	General	Register Accesses	N/A

17.7.6.1 Peripheral Channel (Channel 0) Overview

The Peripheral channel performs the following Functions:

- Target for PCI Device D31:F0: The eSPI controller duplicates the legacy LPC PCI Configuration space registers. These registers are mostly accessed via the BIOS, though some are accessed via the OS as well.
- Tunnel all Host to eSPI slave (EC/SIO) debug device accesses: these are the accesses that used to go over the LPC bus. These include various programmable and fixed I/O ranges as well as programmable Memory ranges. The programmable ranges and their enables reside in the PCI Configuration space.
- Tunnel all accesses from the eSPI slave to the Host. These include Memory Reads and Writes.

17.7.6.2 Virtual Wire Channel (Channel 1) Overview

The Virtual Wire channel uses a standard message format to communicate several types of signals between the components on the platform.

- Sideband and GPIO Pins: System events and other dedicated signals between the PCH and eSPI slave. These signals are tunneled between the two components over eSPI.
- Serial IRQ Interrupts: Interrupts are tunneled from the eSPI slave to the PCH. Both edge and triggered interrupts are supported.

17.7.6.2.1 eSPI Virtual Wires (VW)

Table 17-2 summarizes the behaviors of the PCH virtual wires in eSPI mode.

Table 17-2. eSPI Virtual Wires (VW) (Sheet 1 of 2)

Virtual Wire	PCH Pin Direction	Comments
SUS_PWRDN_ACK	Output	PCH pin is a GPIO when eSPI is enabled. eSPI controller receives as VW message.
SUS_WARN#	Output	PCH pin is a GPIO when eSPI is enabled. eSPI controller receives as VW message.
SUS_ACK#	Input	PCH pin is a GPIO when eSPI is enabled. eSPI controller receives as VW message.
PME#	Input	PCH pin is a GPIO when eSPI is enabled. PCH received as virtual VW from eSPI
WAKE#	Input	PCH pin is a GPIO when eSPI is enabled. PCH received as Assert _SCI or Deassert_SCI from eSPI. It will be WAKE enabled SCI if received in Sx.
SMI#	Input	PCH pin is a GPIO when eSPI is enabled. PCH receives as Assert_SMI or Deassert_SMI message from eSPI.
SCI#	N/A	Not a PCH pin. PCH receives as Assert_SCI or Deassert_SCI message from eSPI.
SLP_A#	Output	PCH pins are retained (native mode) even when eSPI is enabled. When these signals need to change state to indicate the system state, they are first sent as VW to eSPI which transmits an eSPI VW packet to the slave.

Table 17-2. eSPI Virtual Wires (VW) (Sheet 2 of 2)

Virtual Wire	PCH Pin Direction	Comments
SLP_S3# / SLP_S4# / SLP_S5# / SLP_GBE#	Output	PCH pins are retained (native mode) even when eSPI is enabled. When these signals need to change state to indicate the system state, they are first sent as VW to eSPI which transmits an eSPI VW packet to the slave. ***Note:*** These signals are reset on DPWOK which is a deeper reset than ESPI_RESET#. As a result, these signals can be High or Low when RSMRST# and ESPI_RESET# assert. However, they do not change state once in Deep Sx (RSMRST# asserted). The slave must maintain the states of these wires, which stay static, while the PCH is in Deep Sx.
PLTRST#	Output	Platform Reset: Comment to indicate Platform Reset Assertion
Reset CPU INIT# (RCIN)	Input	Keyboard Controller CPU Reset: Sent to request a CPU reset on behalf of the keyboard controller.
ERROR_NONFATAL	Input	Sent when a non-fatal error is detected.
ERROR_FATAL	Input	Sent when a fatal error is detected.
SLAVE_BOOT_LOAD_DONE	Input	Sent when the BMC has completed it's boot process as an indication to eSPI-MC to continue with the G3 to S0 exit. The eSPI Master waits for the assertion of this virtual wire before proceeding with the SLP_S5# deassertion. The intent is that it is never changed except on a G3 exit - it is reset on a G3 entry.
SLAVE_BOOT_LOAD_STATUS	Input	Sent upon completion of the Slave Boot Load from the attached flash. A stat of 1 indicates that the boot code load was successful and that the integrity of the image is intact.
HOST_RESET_WARN	Output	Sent from the MC just before the Host is about to enter reset. Upon receiving, the BMC must flush and quiesce its upstream Peripheral Channel request queues and assert HOST_RESET_ACK VWire. The MC subsequently completes any outstanding posted transactions or completions and then disables the Peripheral Channel via a write to the Slave's Configuration Register.
HOST_REST_ACT	Input	ACK for the HOST_RESET_WARN message
OOB_RESET_WARN	Output	Sent from the MC just before the OOB processor is about to enter reset. Upon receiving, the BMC must flush and quiesce its OOB Channel upstream request queues and assert OOB_RESET_ACK VWire. The-MC subsequently completes any outstanding posted transactions or completions and then disables the OOB Channel via a write to the Slave's Configuration Register.
OOB_RESET_ACK	Input	ACT for OOB_RESET_ACK
HOST_C10	Output	Indicates that the host CPU has entered deep power down state C10 or deeper.
IRQ [0 – NIRQMAX]	Input	Serial Interrupt Request: This message conveys the serial interrupt protocol.
SMIOUT#	Output	Sent by the master as indication that an SMI# event has occurred. The 0 to 1 on this virtual wire corresponds to the assertion and deassertion of the SMI# to the CPU, respectively.
NMIOUT#	Output	Sent by the master as indication that an NMI# event has occurred. The 0 to 1 on this virtual wire corresponds to the assertion and deassertion of the NMI# to the CPU, respectively.

17.7.6.2.2 Interrupt Events

The eSPI supports both level and edge-triggered interrupt. Refer to the eSPI Specification for details on the theory of operation for interrupts over eSPI.

The PCH eSPI controller will issue a message to the PCH interrupt controller when it receives an IRQ group in its VW packet, indicating a state change for that IRQ line number.

The eSPI slave can send multiple VW IRQ index groups in a single eSPI packet, up to the Operating Maximum VW Count programmed in its Virtual Wire Capabilities and Configuration Channel.

The eSPI controller acts only as a transport for all interrupt events generated from the slave. It does not maintain interrupt state, polarity or enable for any of the interrupt events.

17.7.6.3 Out-of-Band Channel (Channel 2) Overview

The Out-of-Band channel performs the following Functions:

- Tunnel MCTP Packets between the Intel ME and eSPI slave device: The Intel ME communicates MCTP messages to/from the device by embedding those packets over the eSPI protocol. This eliminates the SMBus connection between the PCH and the slave device which was used to communicate the MCTP messages in prior PCH generations. The eSPI controller simply acts as a message transport and forwards the packets between the Intel ME and eSPI device.
- Tunnel PCH Temperature Data to the eSPI slave: The eSPI controller stores the PCH temperature data internally and sends it to the slave using a posted OOB message when a request is made to a specific destination address.
- Tunnel PCH RTC Time and Date Bytes to the eSPI slave: the eSPI controller captures this data internally at periodic intervals from the PCH RTC controller and sends it to the slave device using a posted OOB message when a request is made to a specific destination address.

17.7.6.3.1 PCH Temperature Data Over eSPI OOB Channel

The eSPI controller supports the transmitting of PCH thermal data to the eSPI slave. The thermal data consists of one byte of PCH temperature data which is transmitted periodically (~1 ms) from the thermal sensor unit.

The packet formats for the temperature request from the eSPI slave and the PCH response back are shown in Figure 17-2 and Figure 17-3.

Figure 17-2. eSPI Slave Request to PCH for PCH Temperature

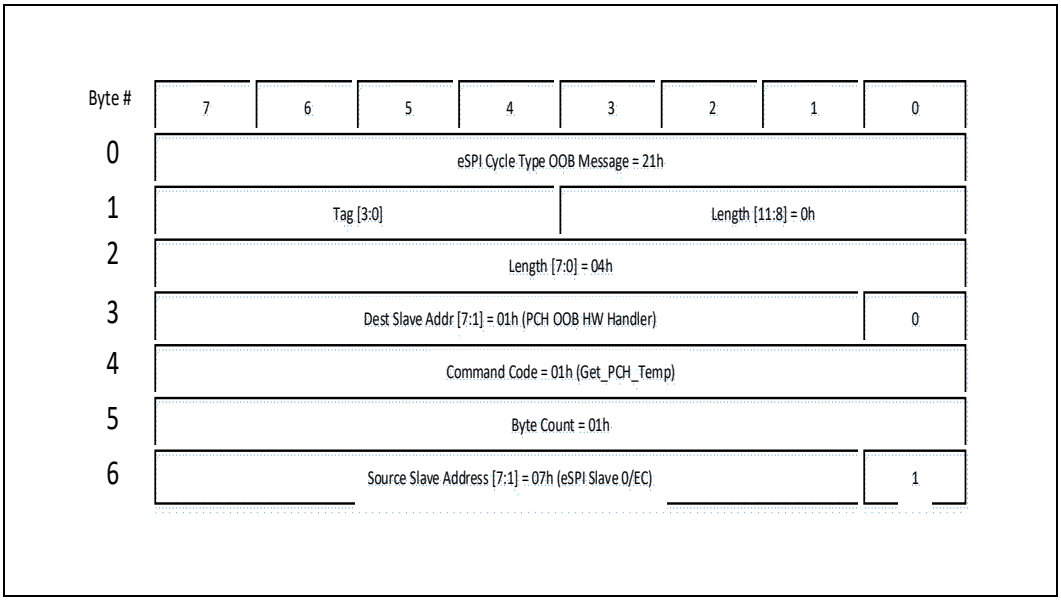
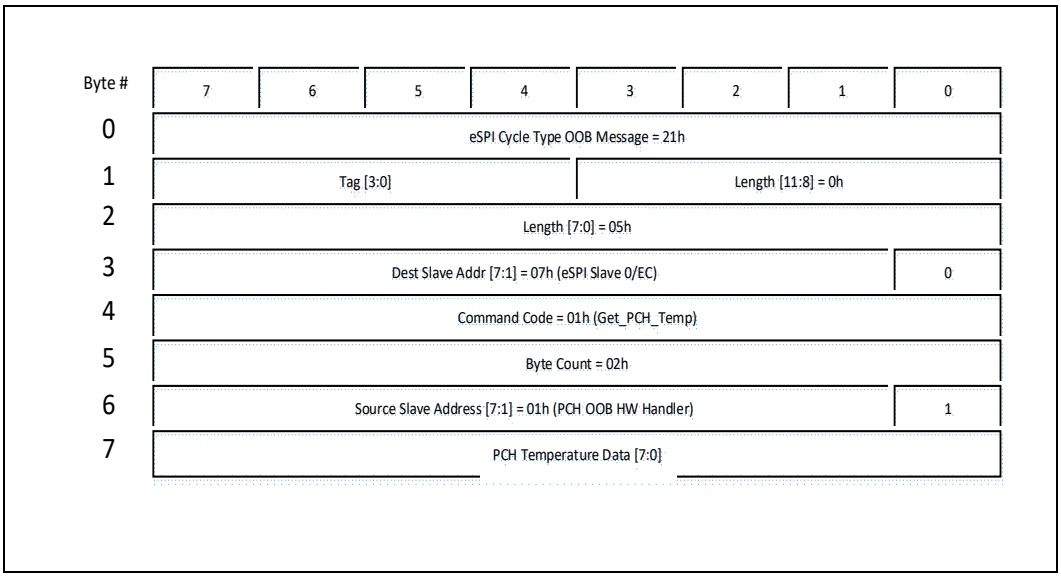


Figure 17-3. PCH Response to eSPI Slave with PCH Temperature



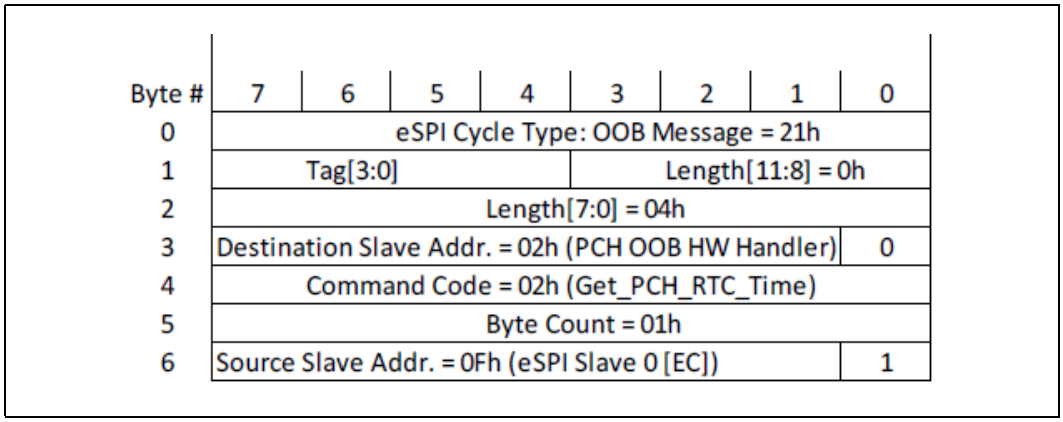
17.7.6.3.2 PCH RTC Time/Date to EC Over eSPI OOB Channel

The PCH eSPI controller supports the transmitting of PCH RTC time/date to the eSPI slave. This allows the eSPI slave to synchronize with the PCH RTC system time. Moreover, using the OOB message channel allows reading of the internal time when the system is in Sx states.

The RTC time consists of seven bytes: seconds, minutes, hours, day of week, day of month, month and year. The controller provides all the time/date bytes together in a single OOB message packet. This avoids the boundary condition of possible roll over on the RTC time bytes if each of the hours, minutes, and seconds bytes is read separately.

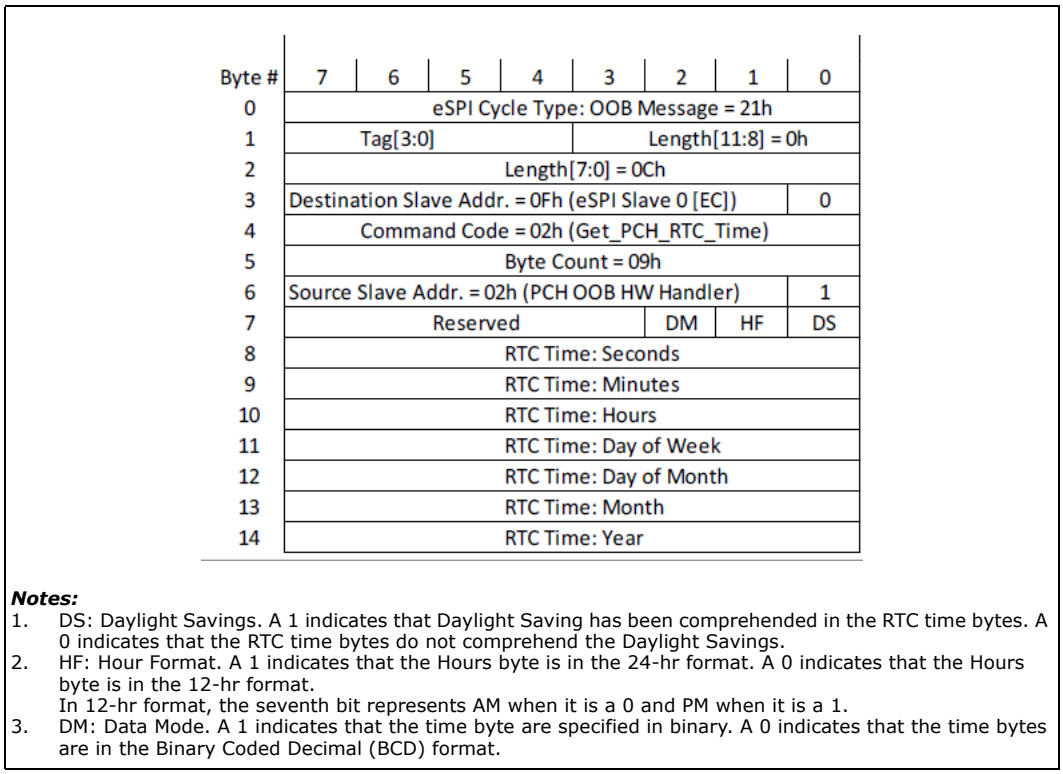
The packet formats for the RTC time/date request from the eSPI slave and the PCH response back to the device are shown in Figure 17-4 and Figure 17-5.

Figure 17-4. eSPI Slave Request to PCH for PCH RTC Time



Byte #	7	6	5	4	3	2	1	0
0	eSPI Cycle Type: OOB Message = 21h							
1	Tag[3:0]				Length[11:8] = 0h			
2	Length[7:0] = 04h							
3	Destination Slave Addr. = 02h (PCH OOB HW Handler)							0
4	Command Code = 02h (Get_PCH_RTC_Time)							
5	Byte Count = 01h							
6	Source Slave Addr. = 0Fh (eSPI Slave 0 [EC])							1

Figure 17-5. PCH Response to eSPI Slave with RTC Time



Byte #	7	6	5	4	3	2	1	0
0	eSPI Cycle Type: OOB Message = 21h							
1	Tag[3:0]				Length[11:8] = 0h			
2	Length[7:0] = 0Ch							
3	Destination Slave Addr. = 0Fh (eSPI Slave 0 [EC])							0
4	Command Code = 02h (Get_PCH_RTC_Time)							
5	Byte Count = 09h							
6	Source Slave Addr. = 02h (PCH OOB HW Handler)							1
7	Reserved					DM	HF	DS
8	RTC Time: Seconds							
9	RTC Time: Minutes							
10	RTC Time: Hours							
11	RTC Time: Day of Week							
12	RTC Time: Day of Month							
13	RTC Time: Month							
14	RTC Time: Year							

Notes:
1. DS: Daylight Savings. A 1 indicates that Daylight Saving has been comprehended in the RTC time bytes. A 0 indicates that the RTC time bytes do not comprehend the Daylight Savings.
2. HF: Hour Format. A 1 indicates that the Hours byte is in the 24-hr format. A 0 indicates that the Hours byte is in the 12-hr format.
   In 12-hr format, the seventh bit represents AM when it is a 0 and PM when it is a 1.
3. DM: Data Mode. A 1 indicates that the time byte are specified in binary. A 0 indicates that the time bytes are in the Binary Coded Decimal (BCD) format.

17.7.6.4 Flash Access Channel (Channel 3) Overview

Note: Slave Attached Flash Sharing (SAFS) is platform dependent on whether it's supported or not. Check your platform supported feature list.

The MAFCC tunnels flash accesses from eSPI slave to the PCH flash controller. The MAFCC simply provides Flash Cycle Type, Address, Length, Payload (for writes) to the flash controller. The flash controller is responsible for all the low level flash operations to perform the requested command and provides a return data/status back to the MAFCC, which then tunnels it back to the eSPI slave in a separate completion packet.

17.7.6.4.1 Master Attached Flash Channel Controller (MAFCC) Flash Operations and Addressing

The EC is allocated a dedicated region within the eSPI Master-Attached flash device. The EC has default read, write, and erase access to this region.

The EC can also access any other flash region as permitted by the Flash Descriptor settings. As such, the EC uses linear addresses, valid up to the maximum supported flash size, to access the flash.

The MAFCC supports flash read, write, and erase operations only.

17.8 Enhanced SPI (eSPI) Configuration Registers Summary

Table 17-3. Summary of Enhanced SPI (eSPI) Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	Identifiers (ESPI_DID_VID)—Offset 0h	8086h
4h	7h	Device Status and Command (ESPI_STS_CMD)—Offset 4h	403h
8h	Bh	Class Code and Revision ID (ESPI_CC_RID)—Offset 8h	6010000h
Ch	Fh	BIST, Header Type, Primary Latency Timer, Cache Line Size (ESPI_BIST_HTYPE_PLT_CLS)—Offset Ch	800000h
2Ch	2Fh	Sub System Identifiers (ESPI_SS)—Offset 2Ch	0h
34h	37h	Capability List Pointer (ESPI_CAPP)—Offset 34h	E0h
80h	83h	I/O Decode Ranges and I/O Enables (ESPI_IOD_IOE)—Offset 80h	0h
84h	87h	LPC Generic I/O Range #1 (ESPI_LGIR1)—Offset 84h	0h
88h	8Bh	LPC Generic I/O Range #2 (ESPI_LGIR2)—Offset 88h	0h
8Ch	8Fh	LPC Generic I/O Range #3 (ESPI_LGIR3)—Offset 8Ch	0h
90h	93h	LPC Generic I/O Range #4 (ESPI_LGIR4)—Offset 90h	0h
94h	97h	USB Legacy Keyboard/Mouse Control (ESPI_ULKMC)—Offset 94h	0h
98h	9Bh	LPC Generic Memory Range (ESPI_LGMR)—Offset 98h	0h
A0h	A3h	eSPI CS1# I/O Routing enables (PCCS1IORE)- Offset A0h	0h
A4h	A7h	eSPI CS1# Generic I/O Range #1 (PCCS1GIR1) - Offset A4h	0h
A8h	ABh	eSPI CS1# Generic Memory Range (PCCS1GMR1) - Offset A8h	0h
D0h	D3h	eSPI CS1# I/O Routing Enables (PCCS1IORE) - Offset A0h	0h
D4h	D7h	FWH ID Select #2 (ESPI_FS2)—Offset D4h	0h
D8h	DBh	BIOS Decode Enable (ESPI_BDE)—Offset D8h	FFCFh
DCh	DFh	BIOS Control (ESPI_BC)—Offset DCh	20h

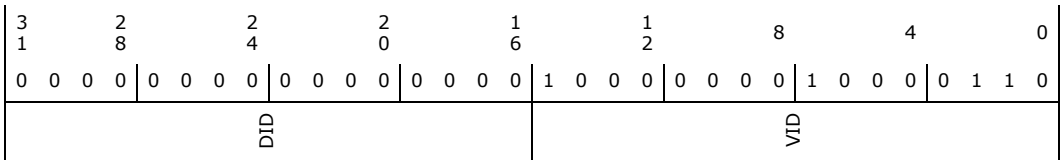
17.8.1 Identifiers (ESPI_DID_VID)—Offset 0h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 8086h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO/V	**Device Identification (DID):** Indicates the Device ID of the controller.
15:0	8086h RO	**Vendor Identification (VID):** Indicates Intel.

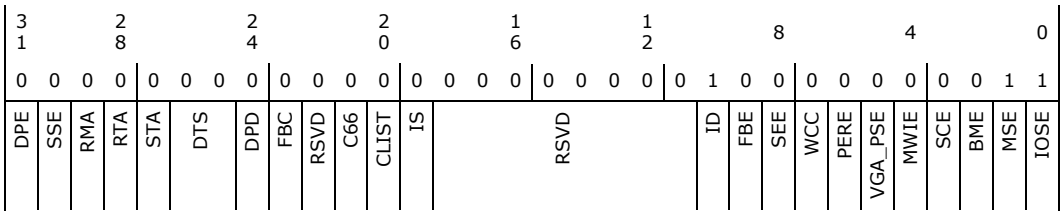
17.8.2 Device Status and Command (ESPI_STS_CMD)—Offset 4h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 403h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RW/1C/V	**Detected Parity Error (DPE):** Set when the bridge detects a parity error on the internal backbone. This bit gets set even if CMD.PERE is not set.
30	0h RW/1C/V	**Signaled System Error (SSE):** Set when the LPC bridge signals a system error to the internal SERR# logic.
29	0h RW/1C/V	**Received Master Abort (RMA):** Set when the bridge receives a completion with unsupported request status.
28	0h RW/1C/V	**Received Target Abort (RTA):** Set when the bridge receives a completion with completer abort status.
27	0h RW/1C/V	**Signaled Target Abort (STA):** Set when the bridge generates a completion packet with target abort status.
26:25	0h RO	**DEVSEL# Timing Status (DTS):** Indicates medium timing, although this has no meaning on the HW.
24	0h RW/1C/V	**Data Parity Error Detected (DPD):** Set when the bridge receives a completion packet from the backbone from a previous request, and detects a parity error, and CMD.PERE is set.
23	0h RO	**Fast Back to Back Capable (FBC):** Reserved - bit has no meaning on the HW.
22	0h RO	Reserved

Bit Range	Default and Access	Field Name (ID): Description
21	0h RO	**66 MHz Capable (C66):** Reserved - bit has no meaning on the HW.
20	0h RO	**Capabilities List (CLIST):** There is a capabilities list in the LPC bridge.
19	0h RO	**Interrupt Status (IS):** The LPC bridge does not generate interrupts.
18:11	0h RO	Reserved
10	1h RO	**Interrupt Disable (ID):** The LPC bridge has no interrupts to disable.
9	0h RO	**Fast Back to Back Enable (FBE):** Reserved as 0 per PCI Express Specification.
8	0h RW	**SERR# Enable (SEE):** The LPC bridge generates SERR# if this bit is set.
7	0h RO	**Wait Cycle Control (WCC):** Reserved as 0 per PCI Express Specification.
6	0h RW	**Parity Error Response Enable (PERE):** When this bit is set to 1, it enables the LPC bridge to response to parity errors detected.
5	0h RO	**VGA Palette Snoop (VGA_PSE):** Reserved as 0 per the *PCI Express Specification*.
4	0h RO	**Memory Write and Invalidate Enable (MWIE):** Reserved as 0 per the *PCI Express Specification*.
3	0h RO	**Special Cycle Enable (SCE):** Reserved as 0 per the *PCI Express Specification*.
2	0h RW	**Bus Master Enable (BME):** Bus Masters cannot be disabled.
1	1h RO	**Memory Space Enable (MSE):** Memory space cannot be disabled on LPC.
0	1h RO	**I/O Space Enable (IOSE):** I/O space cannot be disabled on LPC.

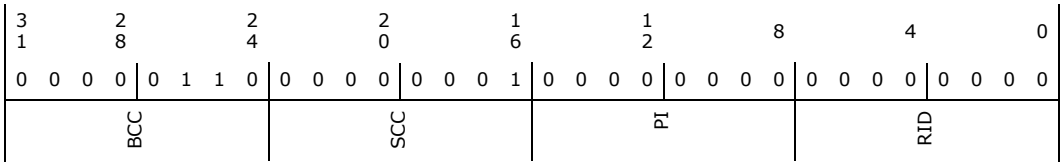
17.8.3 Class Code and Revision ID (ESPI_CC_RID)—Offset 8h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 6010000h



Bit Range	Default and Access	Field Name (ID): Description
31:24	6h RO	**Base Class Code (BCC):** Indicates the device is a bridge device.
23:16	1h RO	**Sub-Class Code (SCC):** Indicates the device a PCI to ISA bridge.
15:8	0h RO	**Programming Interface (PI):** The LPC bridge has no programming interface.
7:0	0h RO/V	**Revision ID (RID):** Indicates the part revision.

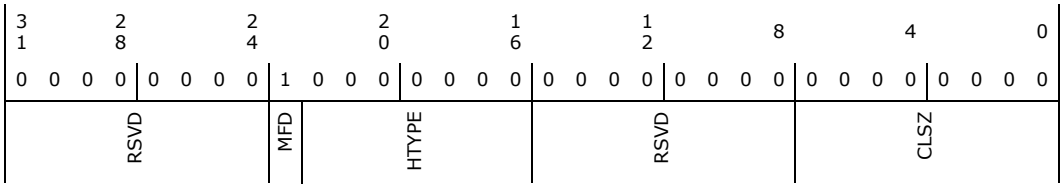
17.8.4 BIST, Header Type, Primary Latency Timer, Cache Line Size (ESPI_BIST_HTYPE_PLT_CLS)—Offset Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 800000h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	1h RO	**Multi-function Device (MFD):** This bit is 1 to indicate a multifunction device.

Bit Range	Default and Access	Field Name (ID): Description
22:16	0h RO	**Header Type (HTYPE):** Identifies the header layout of the configuration space, which is a generic device.
15:8	0h RO	Reserved
7:0	0h RO	**Cacheline Size (CLSZ):** Cacheline Size

17.8.5 Sub System Identifiers (ESPI_SS)—Offset 2Ch

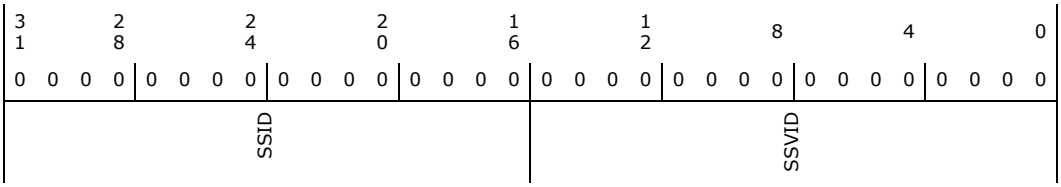
This register is initialized to logic 0 by the assertion of PLTRST#. This register can be written only once after PLTRST# de-assertion.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RW/O	**Subsystem ID (SSID):** This is written by the BIOS. No hardware action taken on this value.
15:0	0h RW/O	**Subsystem Vendor ID (SSVID):** This is written by the BIOS. No hardware action taken on this value.

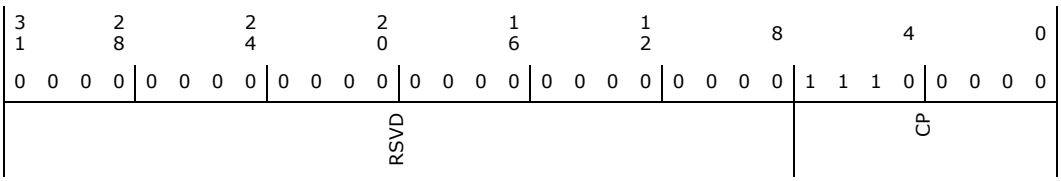
17.8.6 Capability List Pointer (ESPI_CAPP)—Offset 34h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: E0h



Bit Range	Default and Access	Field Name (ID): Description
31:8	0h RO	Reserved
7:0	E0h RO	**Capability Pointer (CP):** Indicates the offset of the first Capability Item.

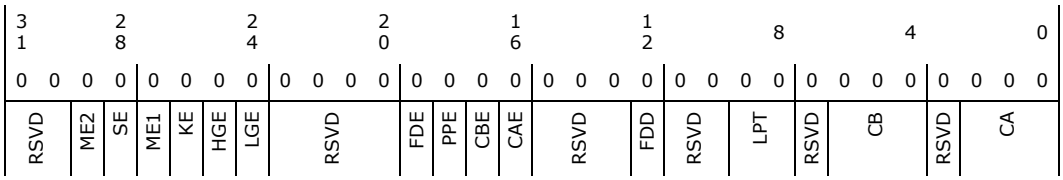
17.8.7 I/O Decode Ranges and I/O Enables (ESPI_IOD_IOE)—Offset 80h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:30	0h RO	Reserved
29	0h RW	**Microcontroller Enable #2 (ME2):** Enables decoding of I/O locations 4Eh and 4Fh to LPC.
28	0h RW	**Super I/O Enable (SE):** Enables decoding of I/O locations 2Eh and 2Fh to LPC.
27	0h RW	**Microcontroller Enable #1 (ME1):** Enables decoding of I/O locations 62h and 66h to LPC.
26	0h RW	**Keyboard Enable (KE):** Enables decoding of the keyboard I/O locations 60h and 64h to LPC.
25	0h RW	**High Gameport Enable (HGE):** Enables decoding of the I/O locations 208h to 20Fh to LPC.
24	0h RW	**Low Gameport Enable (LGE):** Enables decoding of the I/O locations 200h to 207h to LPC.
23:20	0h RO	Reserved
19	0h RW	**Floppy Drive Enable (FDE):** Enables decoding of the FDD range to LPC. Range is selected by LIOD.FDE.
18	0h RW	**Parallel Port Enable (PPE):** Enables decoding of the LPT range to LPC. Range is selected by LIOD.LPT.
17	0h RW	**Com Port B Enable (CBE):** Enables decoding of the COMB range to LPC. Range is selected by LIOD.CB.
16	0h RW	**Com Port A Enable (CAE):** Enables decoding of the COMA range to LPC. Range is selected by LIOD.CA.

Bit Range	Default and Access	Field Name (ID): Description
15:13	0h RO	Reserved
12	0h RW	**FDD Range (FDD):** The following table describes which range to decode for the FDD Port Bits Decode Range: 0 3F0h - 3F5h, 3F7h (Primary) 1 370h - 375h, 377h (Secondary)
11:10	0h RO	Reserved
9:8	0h RW	**LPT Range (LPT):** The following table describes which range to decode for the LPT Port: Bits Decode Range: 00 378h - 37Fh and 778h - 77Fh 01 278h - 27Fh (port 279h is read only) and 678h - 67Fh 10 3BCh - 3BEh and 7BCh - 7BEh 11 Reserved
7	0h RO	Reserved
6:4	0h RW	**ComB Range (CB):** The following table describes which range to decode for the COMB Port: Bits Decode Range 000 3F8h - 3FFh (COM 1) 001 2F8h - 2FFh (COM 2) 010 220h - 227h 011 228h - 22Fh 100 238h - 23Fh 101 2E8h - 2EFh (COM 4) 110 338h - 33Fh 111 3E8h - 3EFh (COM 3)
3	0h RO	Reserved
2:0	0h RW	**ComA Range (CA):** The following table describes which range to decode for the COMA Port Bits Decode Range: 000 3F8h - 3FFh (COM 1) 001 2F8h - 2FFh (COM 2) 010 220h - 227h 011 228h - 22Fh 100 238h - 23Fh 101 2E8h - 2EFh (COM 4) 110 338h - 33Fh 111 3E8h - 3EFh (COM 3)

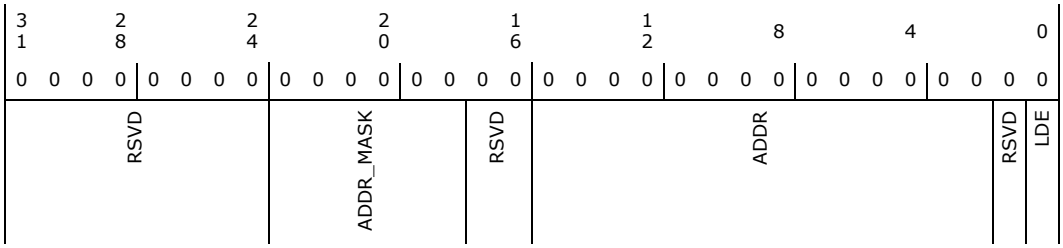
17.8.8 LPC Generic I/O Range #1 (ESPI_LGIR1)—Offset 84h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved
23:18	0h RW	**Address[7:2] Mask (ADDR_MASK):** A 1 in any bit position indicates that any value in the corresponding address bit in a received cycle will be treated as a match. The corresponding bit in the Address field, below, is ignored. The mask is only provided for the lower 6 bits of the DWord address, allowing for decoding blocks up to 256 bytes in size.
17:16	0h RO	Reserved
15:2	0h RW	**Address[15:2] (ADDR):** DWord-aligned address. Note that PCH does not provide decode down to the word or byte level.
1	0h RO	Reserved
0	0h RW	**LPC Decode Enable (LDE):** When this bit is set to 1, then the range specified in this register is enabled for decoding to LPC.

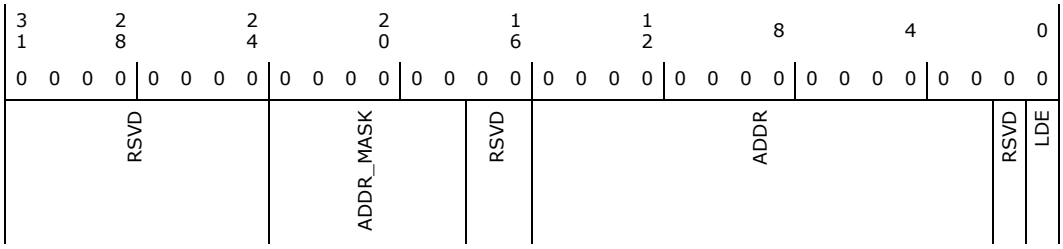
17.8.9 LPC Generic I/O Range #2 (ESPI_LGIR2)—Offset 88h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved
23:18	0h RW	**Address[7:2] Mask (ADDR_MASK):** A 1 in any bit position indicates that any value in the corresponding address bit in a received cycle will be treated as a match. The corresponding bit in the Address field, below, is ignored. The mask is only provided for the lower 6 bits of the DWord address, allowing for decoding blocks up to 256 bytes in size.
17:16	0h RO	Reserved
15:2	0h RW	**Address[15:2] (ADDR):** DWord-aligned address. Note that PCH does not provide decode down to the word or byte level.
1	0h RO	Reserved
0	0h RW	**LPC Decode Enable (LDE):** When this bit is set to 1, then the range specified in this register is enabled for decoding to LPC.

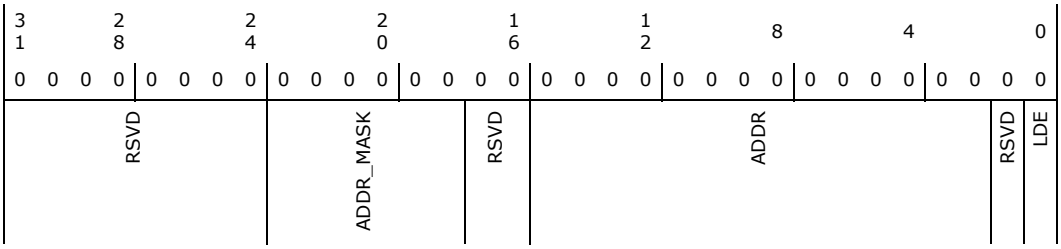
17.8.10 LPC Generic I/O Range #3 (ESPI_LGIR3)—Offset 8Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved
23:18	0h RW	**Address[7:2] Mask (ADDR_MASK):** A 1 in any bit position indicates that any value in the corresponding address bit in a received cycle will be treated as a match. The corresponding bit in the Address field, below, is ignored. The mask is only provided for the lower 6 bits of the DWord address, allowing for decoding blocks up to 256 bytes in size.
17:16	0h RO	Reserved
15:2	0h RW	**Address[15:2] (ADDR):** DWord-aligned address. Note that PCH does not provide decode down to the word or byte level.
1	0h RO	Reserved
0	0h RW	**LPC Decode Enable (LDE):** When this bit is set to 1, then the range specified in this register is enabled for decoding to LPC.

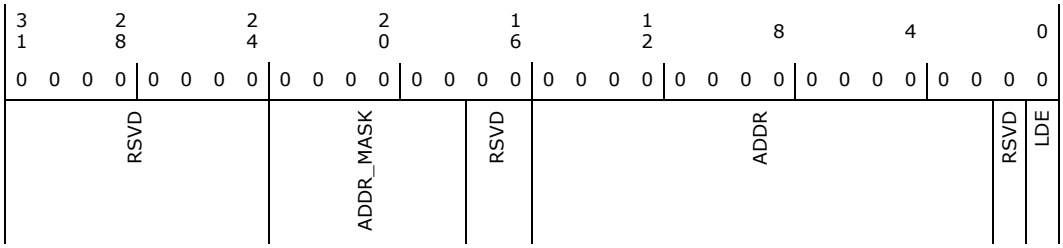
17.8.11 LPC Generic I/O Range #4 (ESPI_LGIR4)—Offset 90h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved
23:18	0h RW	**Address[7:2] Mask (ADDR_MASK):** A 1 in any bit position indicates that any value in the corresponding address bit in a received cycle will be treated as a match. The corresponding bit in the Address field, below, is ignored. The mask is only provided for the lower 6 bits of the DWord address, allowing for decoding blocks up to 256 bytes in size.
17:16	0h RO	Reserved
15:2	0h RW	**Address[15:2] (ADDR):** DWord-aligned address. Note that PCH does not provide decode down to the word or byte level.
1	0h RO	Reserved
0	0h RW	**LPC Decode Enable (LDE):** When this bit is set to 1, then the range specified in this register is enabled for decoding to LPC.

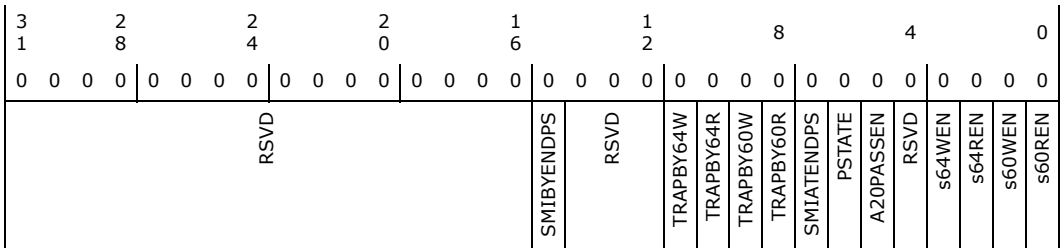
17.8.12 USB Legacy Keyboard/Mouse Control (ESPI_ULKMC)—Offset 94h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved
15	0h RW/1C/V	**SMI Caused by End of Pass-through (SMIBYENDPS):** Indicates if the event occurred. Note that even if the corresponding enable bit is not set in the bit 7, then this bit will still be active. It is up to the SMM code to use the enable bit to determine the exact cause of the SMI#. Writing a 1 to this bit will clear the latch.
14:12	0h RO	Reserved
11	0h RW/1C/V	**SMI Caused by Port 64 Write (TRAPBY64W):** Indicates if the event occurred. Note that even if the corresponding enable bit is not set in the bit 3, then this bit will still be active. It is up to the SMM code to use the enable bit to determine the exact cause of the SMI#. Writing a 1 to this bit will clear the latch. Note that the A20Gate Pass-Through Logic allows specific port 64h Writes to complete without setting this bit.
10	0h RW/1C/V	**SMI Caused by Port 64 Read (TRAPBY64R):** Indicates if the event occurred. Note that even if the corresponding enable bit is not set in the bit 2, then this bit will still be active. It is up to the SMM code to use the enable bit to determine the exact cause of the SMI#. Writing a 1 to this bit will clear the latch.
9	0h RW/1C/V	**SMI Caused by Port 60 Write (TRAPBY60W):** Indicates if the event occurred. Note that even if the corresponding enable bit is not set in the bit 1, then this bit will still be active. It is up to the SMM code to use the enable bit to determine the exact cause of the SMI#. Writing a 1 to this bit will clear the latch. Note that the A20Gate Pass-Through Logic allows specific port 60h Writes to complete without setting this bit.
8	0h RW/1C/V	**SMI Caused by Port 60 Read (TRAPBY60R):** Indicates if the event occurred. Note that even if the corresponding enable bit is not set in the bit 0, then this bit will still be active. It is up to the SMM code to use the enable bit to determine the exact cause of the SMI#. Writing a 1 to this bit will clear the latch.
7	0h RW	**SMI at End of Pass-through Enable (SMIATENDPS):** May need to cause SMI at the end of a pass-through. Can occur if an SMI is generated in the middle of a pass through, and needs to be serviced later.
6	0h RO/V	**Pass Through State (PSTATE):** This read-only bit indicates that the state machine is in the middle of an A20GATE pass-through sequence. If software needs to reset this bit, it should set Bit 5 0.
5	0h RW	**A20Gate Pass-Through Enable (A20PASSEN):** When enabled, allows A20GATE sequence Pass-Through function. When enabled, a specific cycle sequence involving writes to port 60h and port 64h does not result in the setting of the SMI status bits.SMI# will not be generated, even if the various enable bits are set.
4	0h RO	Reserved
3	0h RW	**SMI on Port 64 Writes Enable (s64WEN):** When set, a 1 in bit 11 will cause an SMI event.
2	0h RW	**SMI on Port 64 Reads Enable (s64REN):** When set, a 1 in bit 10 will cause an SMI event.
1	0h RW	**SMI on Port 60 Writes Enable (s60WEN):** When set, a 1 in bit 9 will cause an SMI event.
0	0h RW	**SMI on Port 60 Reads Enable (s60REN):** When set, a 1 in bit 8 will cause an SMI event.

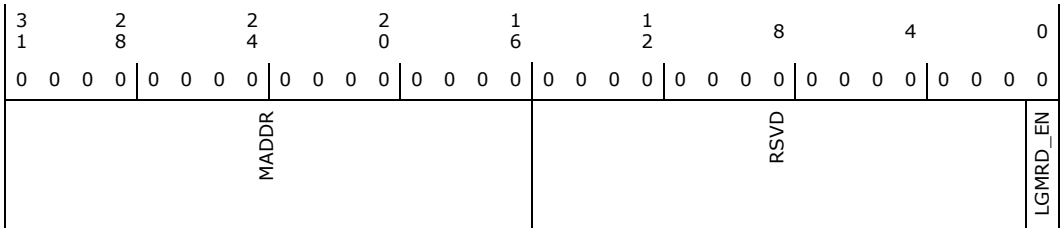
17.8.13 LPC Generic Memory Range (ESPI_LGMR)—Offset 98h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RW	**Memory Address[31:16] (MADDR):** This field specifies a 64 KB memory block anywhere in the 4 GB memory space that will be decoded to LPC as standard LPC Memory Cycle if enabled.
15:1	0h RO	Reserved
0	0h RW	**LPC Memory Range Decode Enable (LGMRD_EN):** When this bit is set to 1, then the range specified in this register is enabled for decoding to LPC.

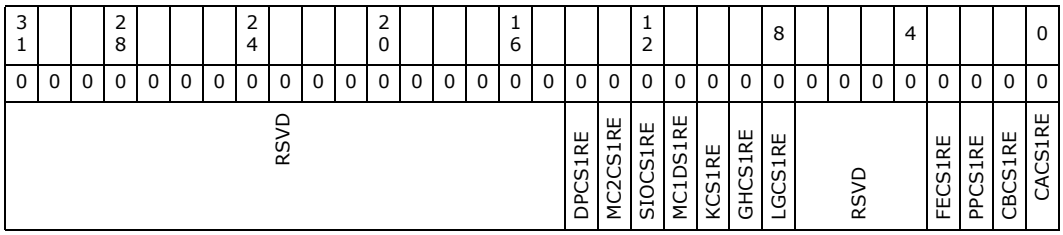
17.8.14 eSPI CS1# I/O Routing Enables (PCCS1IORE) - Offset A0h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:15	0h RO	**Reserved**
14	0b RW/L	**Debug Port CS#1 Routing enable #2 (DPCS1RE)**: Enables the routing of I/O locations 80h, 84-86h, 88h, 8C-8Eh, 90h, 94-96h, 98h, 9C-9Eh to eSPI CS#1.
13	0h RW	**Microcontroller #2 CS#1 Routing Enable (MC2CS1RE)**: Enables the routing of I/O locations 4Eh and 4Fh to eSPI CS#1.
12	0b RW/L	**Super I/O CS#1 Routing Enabling (SIOCS1RE)**: Enables the routing of I/O locations 2Eh and 2Fh to eSPI CS#1.
11	0b RW/L	**Microcontroller #1 CS#1 Routing Enable (MC1CS1RE)**: Enables the routing of I/O locations 62h and 66h to eSPI CS#1.
10	0b RW/L	**Keyboard CS#1 Routing Enable (KCS1RE):** Enables the routing of I/O locations 60h and 64h to eSPI CS#1.
9	0b RW/L	**High Gameport CS#1 Routing Enable (HGCS1RE)**: Enables the routing of I/O locations 208h-20Fh to eSPI CS#1.
8	0b RW/L	**Low Gameport CS#1 Routing Enable (LGCS1RE)**: Enables the routing of I/O locations 200h-207h to eSPI CS#1.
7:4	0h RO	**Reserved**
3	0b RW/L	**Floppy Drive CS1# Routing Enable (FDCS1RE)**: Enables the routing of the FDD range to eSPI CS#1. The range is selected by PCIODR.FDE.
2	0b RW/L	**Parallel PortCS1# Routing Enable (PPCS1RE)**: Enables the routing of the LPT range to eSPI CS#1. The range is selected by PCIODR.LPT.
1	0b RW/L	**Com Port B CS1# Routing Enable (CBCS1RE)**: Enables the routing of the COMB range to eSPI CS#1. The range is selected by PCIODR.CB.
0	0b RW/L	**Com Port A CS1# Routing Enable (CACS1RE)**: Enables the routing of the COMA range to eSPI CS#1. The range is selected by PCIODR.C.

Note: PCIODE (Offset 82h) enables a given range. If that is zero, all access to that range are U/R'ed, irrespective of the settings for that range in this register. So PCIODE should be set/checked first. If a given range is enabled in PCIODE, then this register (PCCS1IORE) is used to select which eSPI slave device to route the transactions to. If a second eSPI slave device is not present, then PCCS1IORE is ignored and all I/O transactions are targeted to any range enabled in PCIODE are routed to CS0#.

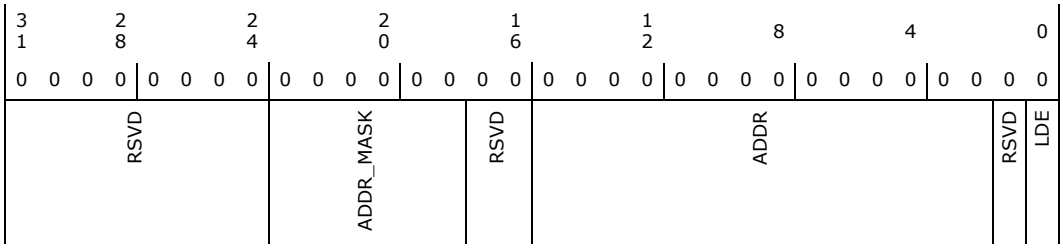
17.8.15 eSPI CS #1 Generic I/O Range (PCCS1GIR1)—Offset A4h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved
23:18	0h RW	**Address[7:2] Mask (ADDR_MASK):** A 1 in any bit position indicates that any value in the corresponding address bit in a received cycle will be treated as a match. The corresponding bit in the Address field, below, is ignored. The mask is only provided for the lower 6 bits of the DWord address, allowing for decoding blocks up to 256 bytes in size.
17:16	0h RO	Reserved
15:2	0h RW	**Address[15:2] (ADDR):** DWord-aligned address. Note that PCH does not provide decode down to the word or byte level.
1	0h RO	Reserved
0	0h RW	**LPC (eSPI) Decode Enable (LDE):** When this bit is set to 1, then the range specified in this register is enabled for decoding to eSPI.

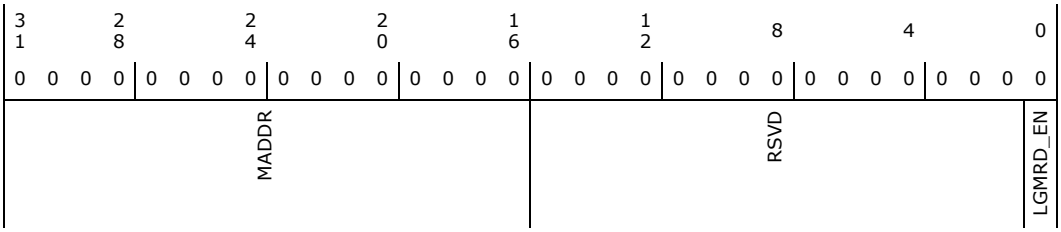
17.8.16 eSPI CS#1 Generic Memory Range (PCCS1GMR1)—Offset A8h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RW	**Memory Address[31:16] (MADDR):** This field specifies a 64 KB memory block anywhere in the 4 GB memory space that will be decoded to eSPI as standard LPC Memory Cycle if enabled.
15:1	0h RO	Reserved
0	0h RW	**LPC (eSPI) Memory Range Decode Enable (LGMRD_EN):** When this bit is set to 1, then the range specified in this register is enabled for decoding to eSPI.

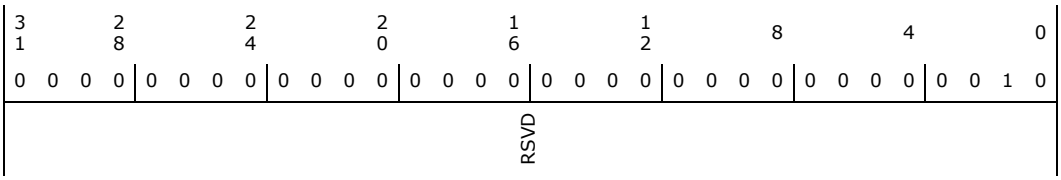
17.8.17 FWH ID Select #1 (ESPI_FS1)—Offset D0h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 00000000h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RO	**Reserved:** eSPI Peripheral Channel does not support Legacy LPC FWH.

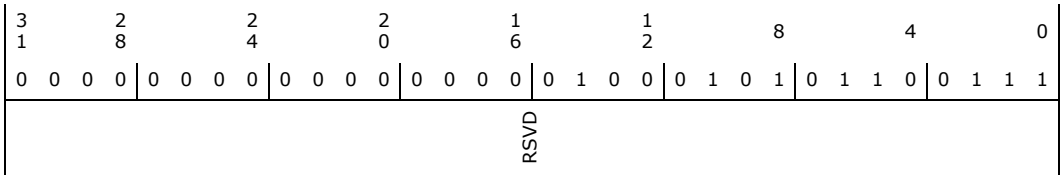
17.8.18 FWH ID Select #2 (ESPI_FS2)—Offset D4h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 000000000h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RO	Reserved. eSPI Peripheral channel does not support Legacy LPC FWH.

17.8.19 BIOS Decode Enable (ESPI_BDE)—Offset D8h

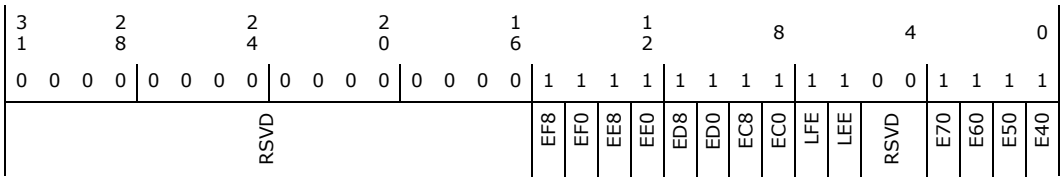
Note: This register effects the BIOS decode regardless of whether the BIOS is resident on LPC or SPI. The concept of Feature Space does not apply to SPI-based flash. PCH simply decodes these ranges as memory accesses when enabled for the SPI flash interface.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: FFCFh



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved
15	1h RO	**F8-FF Enable (EF8):** Enables decoding of 512K of the following BIOS range: Data space: FFF80000h - FFFFFFFFh Feature space: FFB80000h - FFBFFFFFh.
14	1h RW	**F0-F8 Enable (EF0):** Enables decoding of 512K of the following BIOS range: Data space: FFF00000h - FFF7FFFFh Feature space: FFB00000h - FFB7FFFFh.
13	1h RW	**E8-EF Enable (EE8):** Enables decoding of 512K of the following BIOS range: Data space: FFE80000h - FFEFFFFFh Feature space: FFA80000h - FFAFFFFFh.

Bit Range	Default and Access	Field Name (ID): Description
12	1h RW	**E0-E8 Enable (EE0):** Enables decoding of 512K of the following BIOS range: Data space: FFE00000h - FFE7FFFFh Feature Space: FFA00000h - FFA7FFFFh.
11	1h RW	**D8-DF Enable (ED8):** Enables decoding of 512K of the following BIOS range: Data space: FFD80000h - FFDFFFFFh Feature space: FF980000h - FF9FFFFFh.
10	1h RW	**D0-D7 Enable (ED0):** Enables decoding of 512K of the following BIOS range: Data space: FFD00000h - FFD7FFFFh Feature space: FF900000h - FF97FFFFh.
9	1h RW	**C8-CF Enable (EC8):** Enables decoding of 512K of the following BIOS range: Data space: FFC80000h - FFCFFFFFh Feature space: FF880000h - FF8FFFFFh.
8	1h RW	**C0-C7 Enable (EC0):** Enables decoding of 512K of the following BIOS range: Data space: FFC00000h - FFC7FFFFh Feature space: FF800000h - FF87FFFFh.
7	1h RW	**Legacy F Segment Enable (LFE):** This enables the decoding of the legacy 64 KB range at F0000h - FFFFFh. Note that decode for the BIOS legacy F segment is enabled by the LFE bit only, it is not affected by the GEN_PMCON_1.iA64_EN bit.
6	1h RW	**Legacy E Segment Enable (LEE):** This enables the decoding of the legacy 64 KB range at E0000h - EFFFFh. Note that decode for the BIOS legacy E segment is enabled by the LEE bit only, it is not affected by the GEN_PMCON_1.iA64_EN bit.
5:4	0h RO	Reserved
3	1h RW	**70-7F Enable (E70):** Enables decoding of 1 MB of the following BIOS range: Data space: FF700000h - FF7FFFFFh Feature space: FF300000h - FF3FFFFFh.
2	1h RW	**60-6F Enable (E60):** Enables decoding of 1 MB of the following BIOS range: Data space: FF600000h - FF6FFFFFh Feature Space: FF200000h - FF2FFFFFh.
1	1h RW	**50-5F Enable (E50):** Enables decoding of 1 MB of the following BIOS range: Data space: FF500000h - FF5FFFFFh Feature space: FF100000h - FF1FFFFFh.
0	1h RW	**40-4F Enable (E40):** Enables decoding of 1 MB of the following BIOS range: Data space: FF400000h - FF4FFFFFh Feature space: FF000000h - FF0FFFFFh.

17.8.20 BIOS Control (ESPI_BC)—Offset DCh

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 20h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
RSVD																BWRE	BWRS	RSVD	BWPDS	BILD	BBS	EISS	TS	RSVD	ESPI	LE	WPD				

Bit Range	Default and Access	Field Name (ID): Description
31:12	0h RO	Reserved
11	0h RW/L	**BIOS Write Reporting (Async-SMI) Enable (BWRE):** 1'b0: Disable reporting of BIOS Write event. 1'b1: Enable reporting of BIOS Write event (PCBC.BWRS = 1) using Async-SMI.
10	0h RW/1C/V	**BIOS Write Status (BWRS):** HW sets this bit if a memory write access is detected to a protected BIOS range. 1'b0: Memory write to the BIOS region not attempted or attempted with PCBC.WPD = 1. 1'b1: A memory write transaction to the BIOS region has been received with PCBC.WPD = 0. ***Notes:*** An Async-SMI message is generated to report this event if PCBC.BWRE is set. SW must write a 1 to this bit to clear it, which will also deassert the Async-SMI, if PCBC.BWRE is set.
9	0h RO	Reserved
8	0h RW/1C/V	**BIOS Write Protect Disable Status (BWPDS):** When set, prevents BC.TS and BC.BBS from being changed. This bit can only be written from 0 to 1 once. BIOS. ***Note:*** This bit is not backed up in the RTC well. This bit should also be set in the BUC register in the RTC device to record the last state of this value following a cold reset.
7	0h RW/L	**BIOS Interface Lock-Down (BILD):** When set, prevents BC.TS and BC.BBS from being changed. This bit can only be written from 0 to 1 once. BIOS. ***Note:*** This bit is not backed up in the RTC well. This bit should also be set in the BUC register in the RTC device to record the last state of this value following a cold reset.
6	0h RW/V/L	**Boot BIOS Strap (BBS):** This field determines the destination of accesses to the BIOS memory range. For the default, see the Functional Strap section of Signal Description chapter for details. Bits Description 0 SPI 1 LPC When SPI or LPC is selected, the range that is decoded is further qualified by other configuration bits described in the respective sections. The value in this field can be overwritten by software as long as the BIOS Interface Lock-Down is not set.
5	1h RW/L	**Enable InSMM.STS (EISS):** When this bit is set, the BIOS region is not writable until SMM sets the InSMM.STS bit. Today BIOS Flash is writable if WPD is a 1. If this bit (5) is set, then WPD must be a 1 and InSMM.STS (0xFED3_0880(0)) must be 1 also. If this bit (5) is clear, then BIOS is writable based only on WPD = 1 and the InSMM.STS is a don't care.
4	0h RO/V	**Top Swap (TS):** When set, PCH will invert either A16, A17, or A18 for cycles going to the BIOS space (but not the Feature space) in the FWH. When cleared, PCH will not invert A16. If booting from LPC (FWH) or eSPI, then the Boot Block Size is fixed at 64KB and A16 is inverted if Top Swap is enabled. If booting from SPI, then the BOOT_BLOCK_SIZE soft strap determines if A16, A17, or A18 should be inverted if Top Swap is enabled. If the Top-Swap strap is asserted, then this bit cannot be cleared by software. The strap jumper should be removed and the system rebooted. ***Note:*** This bit provides a read-only path to view the state of the Top Swap strap. It is backed up and driven from the RTC well. The BIOS will need to program the corresponding register in the RTC well, which will be reflected in this register.
3	0h RO	Reserved

Bit Range	Default and Access	Field Name (ID): Description
2	0h RO/V	**eSPI Enable Pin Strap (ESPI):** This field determines the destination of accesses to the D31:F0 and related Fixed and Variable I/O and Memory decode ranges, including BIOS memory range. 0 = LPC is the D31:F0 target. 1 = eSPI is the D31:F0 target. ***Notes:*** This field cannot be overwritten by software (unlike the PCBC.BBS field). This bit is also reflected in the LPC (D31:F0) and SPI Flash (D31:F5) PCI Configuration register Offset DCh.
1	0h RW/L	**Lock Enable (LE):** When set, setting the WP bit will cause SMI. When cleared, setting the WP bit will not cause SMI. Once set, this bit can only be cleared by a PLTRST#. When this bit is set, EISS - bit (5) of this register is locked down.
0	0h RW	**Write Protect Disable (WPD):** When set, access to the BIOS space is enabled for both read and write cycles to BIOS. When cleared, only read cycles are permitted to the FWH or SPI flash. When this bit is written from a 0 to a 1 and the LE bit is also set, an SMI# is generated. This ensures that only SMM code can update the BIOS.

17.9 eSPI PCR Registers Summary

These registers are within the PCH Private Configuration Space which is accessible through the PCH Sideband Interface. They can be accessed via (SBREG_BAR + PortID + Register Offset).

Table 17-4. Summary of eSPI Additional Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
4000h	4003h	eSPI Slave Configuration and Link Control (SLV_CFG_REG_CTL)—Offset 4000h	0h
4004h	4007h	eSPI Slave Configuration Register Data (SLV_CFG_REG_DATA)—Offset 4004h	0h
4020h	4023h	Peripheral Channel Error for Slave 0 (PCERR_SLV0)—Offset 4020h	0h
4030h	4033h	Virtual Wire Channel Error for Channel 0 (VWERR_SLV0)—Offset 4030h	0h
4040h	4043h	Flash Access Channel Error for Slave 0 (FCERR_SLV0)—Offset 4040h	0h
4050h	4053h	Link Error for Slave 0 (LNKERR_SL0)—Offset 4050h	FF00h

17.9.1 eSPI Slave Configuration and Link Control (SLV_CFG_REG_CTL)—Offset 4000h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SCRE	SCRS			SBLCL	RSVD						SID		RSVD	SCRT	RSVD					SCRA											

Bit Range	Default and Access	Field Name (ID): Description
31	0h RW/1S/V	**Slave Configuration Register Access Enable (SCRE):** Writing a 1 to this field triggers an access (SCRT) to a Slave Config Register ('Go'). ***Notes:*** Hardware clears this bit to 0 (and sets the SCRS field) when the transaction has completed on the eSPI bus. In the case of a configuration/status register read, the data is valid only after this bit has been cleared by HW. The SCRE is effective only if SCRS is clear.
30:28	0h RW/1C/V	**Slave Configuration Register Access Status (SCRS):** This field is set by upon the completion of a configuration register access (SCRE). Software must clear this field by writing all 1s before initiating another Slave configuration register access (SCRE). 0h: Status not valid 1h: Slave No_Response 2h: Slave Response CRC Error 3h: Slave Response Fatal Error 4h: Slave Response Non-Fatal Error 5h – 6h: Reserved 7h: No errors (transaction completed successfully)
27	0h RW/1S	**SB eSPI Link Configuration Lock (SBLCL):** When set, eSPI controller prevents writes (i.e., SET_CONFIGURATION) to any eSPI Specification defined Slave Capabilities and Configuration registers in the reserved register address range (0h – 7FFh). Access to Slave implementation specific configuration registers outside this range are not impacted by this lock bit and are always available – access protections to such registers are Slave implementation dependent. ***Notes:*** This bit cannot be written to 0 once it has been set to 1. It can only be cleared by PLTRST# assertion. The lock is automatically disabled if and while the LNKERR_SLV0.SLCRR register bit is asserted (upon an eSPI link Fatal Error condition) to allow BIOS (or another SW agent) to attempt to recover the link. This bit has no effect when PLTRST# is asserted. **BIOS Note:** The BIOS must ensure that this bit is set to 1 after initial eSPI link configuration is over to prevent any further (unintentional or malicious) changes to the eSPI link configuration.
26:21	0h RO	Reserved
20:19	0h RW	**Slave ID (SID):** eSPI Slave ID (CS#) to which the Slave Configuration Register Access (SCRT) is directed. 00: eSPI Slave 0 (EC/BMC) 01: eSPI Slave 1 (for Server SKU only and when a when a second eSPI Slave device is present) 10 – 2'b11: Reserved
18	0h RO	Reserved
17:16	0h RW	**Slave Configuration Register Access Type (SCRT):** 00: Slave Configuration register read from address SCRA[11:0] (GET_CONFIG)(br) 01: Slave Configuration register write to address SCRA[11:0] (SET_CONFIG)(br) 10: Slave Status register read (GET_STATUS)(br) 11: In-Band Reset. ***Notes:*** Writes to Slave Configuration registers in the reserved address range (0h – 7FFh) are gated by the SBLCL bit. Setting this field to 10 triggers a Get_Status command to the Slave. In this case, the SCRA field is ignored and only the lower 16-bits of the returned data (SLV_CFG_REG_DATA[15:0]) are valid. Setting this field to 11 triggers an In-Band Reset command to the Slave. In this case, the SCRA field is ignored and no data is returned. This command resets the link for the targeted Slave to a default configuration. Software is responsible for reinitializing the link to optimized (higher performance) settings using these registers.
15:12	0h RO	Reserved
11:0	0h RW	**Slave Configuration Register Address (SCRA)**

17.9.2 eSPI Slave Configuration Register Data (SLV_CFG_REG_DATA)—Offset 4004h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SCRD																															

Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/V	**Slave Configuration Register for Read and Write data (SCRD):** Configuration register Write data from software or read data from the Slave. For writes, this register must be programmed before the CTL register. For reads, data in this register is valid after the CTL.SCRE bit has been cleared by HW and the CTL.SCRS field indicates a successful transaction.

17.9.3 Peripheral Channel Error for Slave 0 (PCERR_SLV0)—Offset 4020h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD			SLV_HOST_RST_ACK_OVRD		PCRMTARE		PCURRE	PCURD	RSVD									PCNFEE	PCNFES	PCNFEC				RSVD		PCFEE	PCFES	PCFEC				

Bit Range	Default and Access	Field Name (ID): Description
31:29	0h RO	Reserved
28	0h RW	**SLV_HOST_RST_ACK_OVRD:** A 1 in this bit will cause the eSPI-MC to not wait for the Slave HOST_RESET_ACK Virtual Wire before (immediately) asserting the ResetPrepAck (Host space, GenPrep). The Host_Reset_Warn VW will be transmitted to the Slave independent of the setting for this bit.
27:26	0h RW	**Peripheral Channel Received Master or Target Abort Reporting Enable (PCRMTARE):** 00: Disable RMA or RTA Reporting 01: Reserved 10: Enable RMA or RTA Reporting as SERR 11: Enable RMA or RTA Reporting as SMI ***Notes:*** SERR enable is also qualified by PCCMD.SEE (Offset 04h) when PLTRST# is deasserted. SERR also sets PCSTS.SSE (Offset 06h) when PLTRST# is deasserted.
25	0h RW	**Peripheral Channel Unsupported Request Reporting Enable (PCURRE):** If set to 1 by software, it allows reporting of an Unsupported Request (UR) as a System Error (SERR). If eSPI controller decodes a Posted transaction that is not supported, it sets the PCURD bit. If PCCMD.SEE (SERR enable) is also set to 1, then eSPIMC sets the PCSTS.SSE (Signaled System Error) bit and sends a Do_SErr message. ***Note:*** If the transaction was a Non-Posted request, then the agent handles the transaction as an Advisory Non-Fatal error, and no error logging or signaling is done. The Completion with UR Completion Status serves the purpose of error reporting.
24	0h RW/1C/V	**Peripheral Channel Unsupported Request Detected (PCURD):** Set to 1 by hardware upon detecting an Unsupported Request (UR) that is not considered an Advisory Non- Fatal error and PCERR.PCURRE is set. Cleared to 0 when software writes a 1 to this register.
23:15	0h RO	Reserved
14:13	0h RW	**Peripheral Channel Non-Fatal Error Reporting Enable (PCNFEE):** 00: Disable Non-Fatal Error Reporting 01: Reserved 10: Enable Non-Fatal Error Reporting as SERR 11: Enable Non-Fatal Error Reporting as SMI ***Notes:*** SERR enable is also qualified by PCCMD.SEE (Offset 04h) when PLTRST# is deasserted. SERR also sets PCSTS.SSE (Offset 06h) when PLTRST# is deasserted. SMI/SERR messages are not generated if the Host is in reset (PLTRST# asserted).
12	0h RW/1C/V	**Peripheral Channel Non-Fatal Status (PCNFES):** This field is set by hardware if a Non-Fatal Error condition is detected on the Peripheral Channel. Software must clear this bit. 0: No Non-Fatal Error detected 1: Non-Fatal Error detected (PCNFEC has a non-zero value) ***Notes:*** Clearing this unlocks the PCNFEC field and triggers a SB Deassert_SMI message if PCNFEE is set to SMI. Setting of this bit is independent of the enable to generate a SMI/SERR (PCNFEE).
11:8	0h RO/V	**Peripheral Channel Non-Fatal Status (PCNFEC):** 0h: No error 1h: Slave Response Code: NONFATAL_ERROR 2h: Slave Response Code: Unsuccessful Completion 3h: Unexpected completion received from Slave (i.e., completion without non-posted request or completion with invalid tag or completion with invalid length) 4h: Unsupported Cycle Type (w.r.t. Command) 5h: Unsupported Message Code 6h: Unsupported Address/Length alignment (upstream only): Memory: Address + Length > 64 B (aligned) [for both Posted and Non-Posted transactions] 7h: Unsupported Address/Length alignment (upstream only): Memory: 64-bit Address with Addr[63:32] = 0h [for both Posted and Non-Posted transactions] 8h – Fh: Reserved ***Note:*** This field is updated after a Peripheral channel transaction is completed if the PCNFES bit is not set.
7	0h RO	Reserved

Bit Range	Default and Access	Field Name (ID): Description
6:5	0h RW	**Peripheral Channel Fatal Error Reporting (PCFEE):** 00: Disable Fatal Error Reporting 01: Reserved 10: Enable Fatal Error Reporting as SERR (IOSF-SB Do_SErr message) 11: Enable Fatal Error Reporting as SMI (IOSF-SB Assert_SMI message) ***Notes:*** SERR enable is also qualified by PCCMD.SEE (Offset 04h) when PLTRST# is deasserted. SERR also sets PCSTS.SSE (Offset 06h) when PLTRST# is deasserted. SMI/SERR messages are not generated if the Host is in reset (PLTRST# asserted).
4	0h RW/1C/V	**Peripheral Channel Fatal Error Reporting (PCFES):** This field is set by hardware if a Fatal Error condition is detected on the Peripheral Channel. Software must clear this bit by writing a 1 to it. 0: No Fatal Error detected 1: Fatal Error Type 2 detected (PCFEC has a non-zero value) ***Notes:*** Clearing this unlocks the PCFEC field and triggers an SB Deassert_SMI message if PCFEE is set to SMI. Setting of this bit is independent of the enable to generate a SMI/SERR (PCFEE).
3:0	0h RO/V	**Peripheral Channel Fatal Error Cause (PCFEC):** 0h: No error 1h – 7h: Reserved 8h: Malformed Slave Response Payload: Payload length > Max. Payload Size (aligned) [Type 2] 9h: Malformed Slave Response Payload: Read request size > Max. Read Request Size (aligned) [Type 2] Ah: Malformed Slave Response Payload: Address + Length > 4 KB (aligned) [Type 2] Bh – Fh: Reserved ***Note:*** This field is updated after a Peripheral channel transaction is completed if the PCFES bit is not set.

17.9.4 Virtual Wire Channel Error for Channel 0 (VWERR_SLV0)—Offset 4030h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																		VWNFEE	VWNFES	VWNFEC				RSVD	VWFEE	VWFES	VWFEC				

Bit Range	Default and Access	Field Name (ID): Description
31:15	0h RO	Reserved
14:13	0h RW	**Virtual Wire Channel Non-Fatal Error Reporting Enable (VWNFEE):** 00: Disable Non-Fatal Error Reporting 01: Reserved 10: Enable Non-Fatal Error Reporting as SERR (SB Do_SErr message) 11: Enable Non-Fatal Error Reporting as SMI (SB Assert_SMI message) ***Note:*** SERR enable is also qualified by PCCMD.SEE (Offset 04h) when PLTRST# is deasserted. SERR also sets PCSTS.SSE (Offset 06h) when PLTRST# is deasserted. SMI/SERR messages are not generated if the Host is in reset (PLTRST# asserted).
12	0h RW/1C/V	**Virtual Wire Channel Non-Fatal Error Status (VWNFES):** This field is set by hardware if a Non-Fatal Error condition is detected on the Virtual Wire Channel. Software must clear this bit. 0: No Non-Fatal Error detected 1: Non-Fatal Error detected (VWNFEC has a non-zero value) ***Note:*** Clearing this unlocks the VWNFEC field and triggers an SB Deassert_SMI message if VWNFEE is set to SMI. Setting of this bit is independent of the enable to generate a SMI/SERR (VWNFEE).
11:8	0h RO/V	**Virtual Wire Channel Non-Fatal Error Cause (VWNFEC):** 0h: No error 1h: Slave Response Code: NONFATAL_ERROR 2h – Dh: Reserved Eh: Slave Virtual Wire: NON_FATAL_ERROR: 0 to 1 transition (1 to 0 transition on this VW is ignored) Fh: Reserved ***Note:*** This field is updated after a Virtual Wire Channel transaction is completed if the VWNFES bit is not set.
7	0h RO	Reserved
6:5	0h RW	**Virtual Wire Channel Fatal Error Reporting Enable (VWFEE):** 00: Disable Fatal Error Reporting 01: Reserved 10: Enable Fatal Error Reporting as SERR (IOSF-SB Do_SErr message) 11: Enable Fatal Error Reporting as SMI (SB Assert_SMI message) ***Notes:*** SERR enable is also qualified by PCCMD.SEE (Offset 04h) when PLTRST# is deasserted. SERR also sets PCSTS.SSE (Offset 06h) when PLTRST# is deasserted. SMI/SERR messages are not generated if the Host is in reset (PLTRST# asserted).
4	0h RW/1C/V	**Virtual Wire Channel Fatal Error Status (VWFES):** This field is set by hardware if a Fatal Error condition is detected on the Virtual Wire Channel. Software must clear this bit by writing all 1s to it. 0: No Fatal Error detected 1: Fatal Error Type 2 detected (VWFEC has a non-zero value) ***Notes:*** Clearing this unlocks the VWFEC field and triggers an SB Deassert_SMI message if VWFEE is set to SMI. Setting of this bit is independent of the enable to generate a SMI/SERR (VWFEE).
3:0	0h RO/V	**Virtual Wire Channel Fatal Error Cause (VWFEC):** 0h: No error 1h – 7h: Reserved 8h: Malformed Slave Response Payload: VW Count > Max. VW Count [Type 2] 9h – 4'hD: Reserved Eh: Slave Virtual Wire: FATAL_ERROR 0 to 1 transition (1 to 0 transition on this VW is ignored) [Type 2] Fh: Reserved ***Note:*** This field is updated after a Virtual Wire Channel transaction is completed if the VWFES bit is not set.

17.9.5 Flash Access Channel Error for Slave 0 (FCERR_SLV0)—Offset 4040h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																		FCNFEE		FCNFES	FCNFEC				RSVD		FCFEE		FCFES	FCFEC		

Bit Range	Default and Access	Field Name (ID): Description
31:15	0h RO	Reserved
14:13	0h RW	**Flash Access Channel Non-Fatal Error Reporting Enable (FCNFEE):** 00: Disable Non-Fatal Error Reporting. 01: Reserved 10: Enable Non-Fatal Error Reporting as SERR (SB Do_SErr message). 11: Enable Non-Fatal Error Reporting as SMI (SB Assert_SMI message). ***Notes:*** SERR enable is also qualified by PCCMD.SEE (Offset 04h) when PLTRST# is deasserted. SERR also sets PCSTS.SSE (Offset 06h) when PLTRST# is deasserted. SMI/SERR messages are not generated if the Host is in reset (PLTRST# asserted).
12	0h RW/1C/V	**Flash Access Channel Non-Fatal Error Status (FCNFES):** This field is set by hardware if a Non-Fatal Error condition is detected on the Flash Access Channel. Software must clear this bit. 0: No Non-Fatal Error detected 1: Non-Fatal Error detected (FCNFEC has a non-zero value) ***Notes:*** Clearing this unlocks the FCNFEC field and triggers an SB Deassert_SMI message if FCNFEE is set to SMI. Setting of this bit is independent of the enable to generate a SMI/SERR (FCNFEE).
11:8	0h RO/V	**Flash Access Channel Non-Fatal Error Cause (FCNFEC):** 0h: No error 1h: Slave Response Code: NONFATAL_ERROR 2h: Slave Response Code: Unsuccessful Completion [for Slave-Attached Flash accesses only] 3h: Unexpected completion received from Slave (i.e., completion without non-posted request or completion with invalid tag or completion with invalid length) [for Slave-Attached Flash accesses only] 2h – 3h: Reserved 4h: Unsupported Cycle Type (w.r.t. Command) [for SPT-LP Master Attached Flash this should only be unsupported Erase Block Size] 5h: Reserved 6h: Unsupported Address (i.e., address > Flash linear address range) [for Master-Attached Flash accesses only] ? set to Flash Access Error 7h: Reserved 8h – Fh: Reserved ***Note:*** This field is updated after a Flash Access Channel transaction is completed if the FCNFES bit is not set
7	0h RO	Reserved

Bit Range	Default and Access	Field Name (ID): Description
6:5	0h RW	**Flash Access Channel Fatal Error Reporting Enable (FCFEE):** 00: Disable Fatal Error Reporting 01: Reserved 10: Enable Fatal Error Reporting as SERR (SB Do_SErr message). 11: Enable Fatal Error Reporting as SMI (SB Assert_SMI message). ***Notes:*** SERR enable is also qualified by PCCMD.SEE (Offset 04h) when PLTRST# is deasserted. SERR also sets PCSTS.SSE (Offset 06h) when PLTRST# is deasserted. SMI/SERR messages are not generated if the Host is in reset (PLTRST# asserted).
4	0h RW/1C/V	**Flash Access Channel Fatal Error Status (FCFES):** This field is set by hardware if a Fatal Error condition is detected on the Flash Access Channel. Software must clear this bit by writing a 1 to it. 0: No Fatal Error detected 1: Fatal Error Type 2 detected (FCFEC has a non-zero value) ***Notes:*** Clearing this unlocks the FCFEC field and triggers an IOSF-SB Deassert_SMI message if FCFEE is set to SMI. Setting of this bit is independent of the enable to generate a SMI/SERR (FCFEE).
3:0	0h RO/V	**Flash Access Channel Fatal Error Cause (FCFEC):** 0h: No error 1h – 7h: Reserved 8h: Malformed Slave Response Payload: Payload length > Max. Payload Size [Type 2] 9h: Malformed Slave Response Payload: Read request size > Max. Read Request Size [for Master-Attached Flash accesses only] [Type 2] Ah – Fh: Reserved ***Note:*** This field is updated after a Flash Access Channel transaction is completed if the FCFES bit is not set.

17.9.6 Link Error for Slave 0 (LNKERR_SL0)—Offset 4050h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: FF00h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
SLCRR	RSVD									LFET1E	LFET1S	LFET1C					LFET1CTYP								LFET1CMD							

Bit Range	Default and Access	Field Name (ID): Description
31	0h RW/1C/V	**eSPI Link and Slave Channel Recovery Required (SLCRR):** HW sets this bit when it has detected a Type 1 Fatal Error condition, for any channel (LFET1C is non-zero). Setting of this bit will trigger an error handling sequence by the eSPI controller followed by the suspension of all HW initiated transactions on the eSPI link with the Slave. SW must clear this bit (by writing a 1 to it) after it has taken all necessary actions to recover the link. This indicates the eSPI controller to resume HW initiated transactions with the Slave. If this bit is set, it impedes transactions from going on in either CS0 or CS1. So an error on CS0 can stop transactions on CS1 and vice Versa
30:23	0h RO	Reserved
22:21	0h RW	**Fatal Error Type 1 Reporting Enable (LFET1E):** 00: Disable Fatal Error Type 1 Reporting 01: Reserved 10: Enable Fatal Error Type 1 Reporting as SERR (SB Do_SErr message). 11: Enable Fatal Error Type 1 Reporting as SMI (SB Assert_SMI message). ***Notes:*** SERR enable is also qualified by PCCMD.SEE (Offset 04h) when PLTRST# is deasserted. SERR also sets PCSTS.SSE (Offset 06h) when PLTRST# is deasserted. SMI/SERR messages are not generated if the Host is in reset (PLTRST# asserted). When this error is reported, SW must also inspect and handle the SLCRR field.
20	0h RW/1C/V	**Fatal Error Type 1 Reporting Status (LFET1S):** This field is set by hardware if a Link Fatal Error Type 1 condition is detected on the eSPI link (any transaction). Software must clear this bit by writing a 1 to it. 0: No Link Fatal Error Type 1 detected 1: Fatal Error Type 1 detected (LFET1C has a non-zero value. ***Notes:*** Clearing this unlocks the LFET1C field and triggers an SB Deassert_SMI message if LFET1E is set to SMI. Setting of this bit is independent of the enable to generate a SMI/SERR (LFET1E).
19:16	0h RO/V	**Link Fatal Type 1 cause (LFET1C):** 0h: No error 1h: Slave Response Code: NO_RESPONSE [Type 1] 2h: Slave Response Code: FATAL_ERROR [Type 1] 3h: Slave Response Code: CRC_ERROR [Type 1] 4h: Invalid Slave Response Code (w.r.t. to Command) [Type 1] 5h: Invalid Slave Cycle Type (w.r.t. to Command) [Type 1] 6h – Fh: Reserved ***Notes:*** This field is updated after a transaction (any channel) is completed if the LFET1S bit is clear. A non-zero value in this field also causes the SLCRR bit to be set.
15:8	FFh RO/V	**Link Fatal Error Type 1 Cycle Type (LFET1CTYP):** When LFET1C is set, this field reflects the Cycle Type for the transaction that encountered the Fatal Error Type 1. If no valid Cycle Type exists w.r.t. the Command (LFET1CMD), this field is set to FFh to indicate that it should be ignored. ***Note:*** This field is updated after a transaction (any channel) is completed if the LFET1S bit is clear.
7:0	0h RO/V	**Link Fatal Error Type 1 Command (LFET1CMD):** When LFET1C is set, this field reflects the Command for the transaction that encountered the Fatal Error Type 1. ***Note:*** This field is updated after a transaction (any channel) is completed if the LFET1S bit is clear.

§

18 General Purpose Input and Output

18.1 Acronyms

Acronyms	Description
GPI	General Purpose Input
GPO	General Purpose Output
GPIO	General Purpose Input/Output
GPP	General Purpose I/O in Primary Well
GPD	General Purpose I/O in Deep Sleep Well
DSW	Deep Sleep Well

18.2 References

None.

18.3 Overview Integrated Pull-Ups and Pull-Downs

GPIOs have weak internal pull-up/pull-down resistors which are off by default, with the exception of those noted. The internal PU/PD can be programmed (PU/PD/None) by BIOS after reset. In Endpoint Mode, with no BIOS to reconfigure the GPIOs, they stay at their defaults.

Note: Integrated pull-ups and pull-downs will not be valid until all the voltages have reached a valid level.

18.4 Overview

The PCH General Purpose Input/Output signals are grouped into multiple groups (e.g., GPP_A, GPP_B, etc.) and are powered by either the PCH Primary well or Deep Sleep well. Most of these pin groups has a dedicated power pin that can be set to either 1.8V or 3.3V with the exception of GPP_I, GPP_L and GPD. GPP_L is powered by VCCPRIM_1P05 due to the need for the power to the native testability pins. GPP_I has 10GbE native functions on it, so these need to be 3.3V. GPD is powered by 3.3V DSW. All pins within the same group (including the native functionality that is multiplexed with the GPIO) operate at the same voltage determined by the power supplied to the power pins.

All PCH GPIOs can be configured as input or output signals. Many GPIOs are multiplexed with other functions. With FITC, you determine whether IE, Intel ME or the Host owns the GPIO. If the Host owns it, then the native vs GPIO selection is done via BIOS. The host configuration registers are displayed further on down in this chapter.

SCI and IOxAPIC interrupt capability is available on all GPIOs. NMI and SMI capability is available on selected GPIOs only.

Table 18-1. GPIO Group Summary

GPIO Group	Power Pins	Voltage
Primary Well Group A (GPP_A)	VCCPGPPA	1.8V or 3.3V
Primary Well Group B (GPP_B)	VCCPGPPB	1.8V or 3.3V
Primary Well Group C (GPP_C)	VCCPGPPC	1.8V or 3.3V
Primary Well Group D (GPP_D)	VCCPGPPD	1.8V or 3.3V
Primary Well Group E (GPP_E)	VCCPGPPE	1.8V or 3.3V
Primary Well Group F (GPP_F)	VCCPGPPF	1.8V or 3.3V
Primary Well Group G (GPP_G)	VCCPGPPG	1.8V or 3.3V
Primary Well Group H(GPP_H)	VCCPGPPH	1.8V or 3.3V
Primary Well Group I (GPP_I)	VCC_3P3	3.3V
Primary Well Group J (GPP_J)	VCCPGPPD	1.8V or 3.3V
Primary Well Group K (GPP_K)	VCCPGPPD	1.8V or 3.3V
Primary Well Group L (GPP_L)	VCCPRIM_1P05	1.05v
Deep Sleep Well Group (GPD)	VCCDSW_3P3	3.3V

18.5 Signal Description

Table 16-1 summarizes the GPIO implementation in the PCH.

Table 18-2. General Purpose I/O Signals (Sheet 1 of 10)

Name	Internal Pull-Up/ Pull-Down (Note 1)	De-Glitch (Note 2)		Multiplexed With	Default	NMI or SMI Capable	Note
		Input	Output				
Group A GPIO - Primary Power Well (1.8V or 3.3V)							
GPP_A0	NO	NO	YES	RCIN# (Mode1) ESPI_ALERT#(Mode3)	RCIN# ESPI_ALERT#	No	Default depends on whether eSPI is selected by hardstrap.
GPP_A1	NO	NO	YES	LAD0 (Mode1) ESPI_IO0 (Mode3)	LAD0 ESPI_IO0	No	Default depends on whether eSPI is selected by hardstrap.
GPP_A2	NO	NO	YES	LAD1 (Mode1) ESPI_IO1(Mode3)	LAD1 ESPI_IO1	No	Default depends on whether eSPI is selected by hardstrap.
GPP_A3	NO	NO	YES	LAD2(Mode1) ESPI_IO2(Mode3)	LAD2 ESPI_IO2	No	Default depends on whether eSPI is selected by hardstrap.
GPP_A4	NO	NO	YES	LAD3 (Mode1) ESPI_IO3(Mode3)	LAD3 ESPI_IO3	No	Default depends on whether eSPI is selected by hardstrap.
GPP_A5	NO	NO	YES	LFRAME#(Mode1) ESPI_CS0#(Mode3)	LFRAME# ESPI_CS0#	No	Default depends on whether eSPI is selected by hardstrap.
GPP_A6	NO	NO	YES	SERIRQ(Mode1) ESPI_CS1#(Mode3)	SERIRQ ESPI_CS1#	No	Default depends on whether eSPI is selected by hardstrap.

Table 18-2. General Purpose I/O Signals (Sheet 2 of 10)

Name	Internal Pull-Up/ Pull-Down (Note 1)	De-Glitch (Note 2)		Multiplexed With	Default	NMI or SMI Capable	Note
		Input	Output				
GPP_A7	NO	NO	YES	PIRQA#(Mode1) ESPI_ALERT0#(Mode3)	PIRQA# ESPI_ALERT0#	No	Default depends on whether eSPI is selected by hardstrap.
GPP_A8	NO	NO	YES	CLKRUN#(Mode1)	ClKRUN#	No	
GPP_A9	20KPD	NO	YES	CLKOUT_LCP0 (Mode1) ESPI_CLK(Mode3)	CLKOUT_LCP0 ESPI_CLK	No	Default depends on whether eSPI is selected by hardstrap. **This pin can never be a GPIO pin, only native function clocks.**
GPP_A10	NO	NO	YES	CLKOUT_LPC1(Mode1)	CLKOUT_LPC1	No	
GPP_A11	NO	NO	YES	PME#(Mode1)	GPP_A11 PME#	No	Default depends on whether eSPI is selected by hardstrap.
GPP_A12	NO	NO	YES	BMBUSY#(Mode1) SXEXITHLDOFF#(Mode3)	GPP_A12	No	
GPP_A13	NO	NO	YES	SUSWARN#_SUSPWRDNACK (Mode1)	SUSWARN#_SUSPWRDNACK GPP_A13	No	If eSPI is enabled, then GPP_A13.
GPP_A14	NO	NO	YES	ESPI_RESET#(Mode3)	ESPI_RESET# GPP_A14	No	If eSPI is enabled, then ESPI_RESET#.
GPP_A15	NO	NO	YES	SUSACK#(Mode1)	SUSACK# GPP_A15	No	If eSPI is enabled, then GPP_A15.
GPP_A16	NO	NO	YES	CLKOUT_LPC2(Mode1)	CLKOUT_LPC2 GPP_A16	No	If eSPI is enabled, then GPP_A16.
GPP_A17	NO	NO	YES	NONE	GPI	No	
GPP_A18	NO	NO	YES	NONE	GPI	No	
GPP_A19	NO	NO	YES	NONE	GPI	No	
GPP_A20	NO	NO	YES	NONE	GPO	No	
GPP_A21	NO	NO	YES	NONE	GPO	No	
GPP_A22	NO	NO	YES	NONE	GPO	No	
GPP_A23	NO	NO	YES	NONE	GPO	No	
Group B GPIO - Primary Power Well (1.8V or 3.3V)							
GPP_B0	20K PU	NO	YES	CORE_VID0 (Mode1)	Native	No	
GPP_B1	20K PU	NO	YES	CORE_VID1 (Mode1)	Native	No	
GPP_B2	NO	NO	YES	NONE	GPI	No	
GPP_B3	NO	NO	YES	CPU_GP2 (Mode1)	GPI	No	
GPP_B4	NO	NO	YES	CPU_GP3 (Mode1)	GPI	No	
GPP_B5	NO	NO	YES	SRCCLKREQ0#(Mode1)	GPI	No	
GPP_B6	NO	NO	YES	SRCCLKREQ1#(Mode1)	GPI	No	
GPP_B7	NO	NO	YES	SRCCLKREQ2#(Mode1)	GPI	No	
GPP_B8	NO	NO	YES	SRCCLKREQ3#(Mode1)	GPI	No	
GPP_B9	NO	NO	YES	SRCCLKREQ4#(Mode1)	GPI	No	
GPP_B10	NO	NO	YES	SRCCLKREQ5#(Mode1)	GPI	No	
GPP_B11	NO	NO	YES	NONE	GPO	No	

Table 18-2. General Purpose I/O Signals (Sheet 3 of 10)

Name	Internal Pull-Up/ Pull-Down (Note 1)	De-Glitch (Note 2)		Multiplexed With	Default	NMI or SMI Capable	Note
		Input	Output				
GPP_B12	NO	NO	YES	GLB_RST_WARN_N#(Mode1)	GLB_RST_WARN_N#	No	
GPP_B13	NO	NO	YES	PLTRST#(Mode1)	PLTRST#	No	
GPP_B14	20K PD	NO	YES	SPKR(Mode1)	GPO	NMI/SMI	Top swap override strap
GPP_B15	NO	NO	YES	NONE	GPI	No	
GPP_B16	NO	NO	YES	NONE	GPI	No	
GPP_B17	NO	NO	YES	NONE	GPI	No	
GPP_B18	20K PD	NO	YES	NONE	GPO	No	No reboot strap
GPP_B19	NO	NO	YES	NONE	GPI	No	
GPP_B20	NO	NO	YES	NONE	GPI	NMI/SMI	
GPP_B21	NO	NO	YES	NONE	GPI	No	
GPP_B22	20k PD	NO	YES	NONE	GPO	No	Boot BIOS strap
GPP_B23	20K PD	YES	YES	SML1ALRT# (Mode1) PHOT#(Mode2) MEIE_SML1ALRT#(Mode3)	GPO	NMI/SMI	DCI disable pin strap. Needs to be low on rising edge of RSMRST# to enable DCI.
Group C GPIO - Primary Power Well (1.8V or 3.3V)							
GPP_C0	NO	YES	YES	SMBCLK (Mode1)	SMBCLK	No	
GPP_C1	NO	YES	YES	SMBDATA (Mode1)	SMBDATA	No	
GPP_C2	20K PD	YES	YES	SMBALERT# (Mode1)	GPO	No	TLS confidentiality strap
GPP_C3	NO	YES	YES	SML0CLK# (Mode1) SML0CLK_IE (Mode2)	SML0CLK_IE	No	
GPP_C4	NO	YES	YES	SML0DATA (Mode1) SML0DATA_IE (Mode2)	SML0DATA_IE	No	
GPP_C5	20K PD	YES	YES	SML0ALERT# (Mode1) SML0ALERT_IE# (Mode2)	GPO	No	eSPI selection strap
GPP_C6	NO	YES	YES	SML1CLK (Mode1) SML1CLK_IE (Mode2)	GPI	No	
GPP_C7	NO	YES	YES	SML1DATA (Mode1) SML1DATA_IE (Mode2)	GPI	No	
GPP_C8	NO	NO	YES	NONE	GPI	No	
GPP_C9	NO	NO	YES	NONE	GPI	No	
GPP_C10	NO	NO	YES	NONE	GPI	No	
GPP_C11	NO	NO	YES	NONE	GPI	No	
GPP_C12	NO	NO	YES	NONE	GPI	No	
GPP_C13	NO	NO	YES	NONE	GPI	No	
GPP_C14	NO	NO	YES	NONE	GPI	No	
GPP_C15	NO	NO	YES	NONE	GPI	No	
GPP_C16	NO	NO	YES	NONE	GPI	No	
GPP_C17	NO	NO	YES	NONE	GPI	No	
GPP_C18	NO	NO	YES	NONE	GPI	No	
GPP_C19	NO	NO	YES	NONE	GPI	No	
GPP_C20	NO	NO	YES	NONE	GPI	No	
GPP_C21	NO	NO	YES	NONE	GPI	No	

Table 18-2. General Purpose I/O Signals (Sheet 4 of 10)

Name	Internal Pull-Up/ Pull-Down (Note 1)	De-Glitch (Note 2)		Multiplexed With	Default	NMI or SMI Capable	Note
		Input	Output				
GPP_C22	NO	NO	YES	NONE	GPI	NMI/SMI	
GPP_C23	NO	NO	YES	NONE	GPI	NMI/SMI	
Group D GPIO in Primary Power Well (1.8V or 3.3V)							
GPP_D0	NO	NO	YES	Serial Blink (Mode3)	GPI	NMI/SMI	
GPP_D1	NO	NO	YES	Serial Blink (Mode3)	GPI	NMI/SMI	
GPP_D2	NO	NO	YES	Serial Blink (Mode3)	GPI	NMI/SMI	
GPP_D3	NO	NO	YES	Serial Blink (Mode3)	GPI	NMI/SMI	
GPP_D4	NO	YES	YES	Serial Blink (Mode3)	GPI	NMI/SMI	
GPP_D5	NO	NO	YES	None	GPI	No	
GPP_D6	NO	NO	YES	None	GPI	No	
GPP_D7	NO	NO	YES	None	GPI	No	
GPP_D8	NO	NO	YES	None	GPI	No	
GPP_D9	NO	NO	YES	SSATA_DEVSLP3 (Mode3)	GPI	No	
GPP_D10	NO	NO	YES	SSATA_DEVSLP4 (Mode3)	GPI	No	
GPP_D11	NO	NO	YES	SSATA_DEVSLP5 (Mode3)	GPI	No	
GPP_D12	NO	NO	YES	SSATA_SDATAOUT1 (Mode3)	GPI	No	
GPP_D13	NO	NO	YES	SML0BCLK (Mode 2) SML0BCLK_IE (Mode 3)	GPI	No	
GPP_D14	NO	NO	YES	SML0BDATA (Mode 2) SML0BDATA_IE (Mode3)	GPI	No	
GPP_D15	NO	NO	YES	SSATA_SDATAOUT0 (Mode3)	GPI	No	
GPP_D16	NO	NO	YES	SML0BALERT# (Mode 2) SML0BALERT_IE# (Mode 3)	GPI	No	
GPP_D17	NO	NO	YES	None	GPI	No	
GPP_D18	NO	NO	YES	None	GPI	No	
GPP_D19	NO	NO	YES	None	GPI	No	
GPP_D20	NO	NO	YES	None	GPI	No	
GPP_D21	NO	NO	YES	IE_UART_RX (Mode3)	GPI	No	
GPP_D22	NO	NO	YES	IE_UART_TX (Mode3)	GPI	No	
GPP_D23	NO	YES	YES	NONE	GPI	No	
Group E GPIO - Primary Power Well (1.8V or 3.3V)							
GPP_E0	NO	NO	YES	SATAXPCIE0 (Mode1) SATAGP0 (Mode2)	SATAXPCIE0 GPP_E0	SMI/NMI	Default depends on soft strap.
GPP_E1	NO	NO	YES	SATAXPCIE1 (Mode1) SATAGP1 (Mode2)	STATAXPCIE1 GPP_E1	SMI/NMI	Default depends on soft strap.
GPP_E2	NO	NO	YES	SATAXPCIE2 (Mode1) SATAGP2 (Mode2)	STATAXPCIE2 GPP_E2	SMI/NMI	Default depends on soft strap.
GPP_E3	NO	NO	YES	CPU_GP0 (Mode1)	GPI	SMI/NMI	
GPP_E4	NO	NO	YES	SATA_DEVSLP0 (Mode1)	GPI	SMI/NMI	
GPP_E5	NO	NO	YES	SATA_DEVSLP1 (Mode1)	GPI	SMI/NMI	
GPP_E6	NO	NO	YES	SATA_DEVSLP2 (Mode1)	GPI	SMI/NMI	

Table 18-2. General Purpose I/O Signals (Sheet 5 of 10)

Name	Internal Pull-Up/ Pull-Down (Note 1)	De-Glitch (Note 2)		Multiplexed With	Default	NMI or SMI Capable	Note
		Input	Output				
GPP_E7	NO	NO	YES	CPU_GP1 (Mode1)	GPI	SMI/NMI	
GPP_E8	NO	NO	YES	SATA_LED# (Mode1)	GPI	SMI/NMI	
GPP_E9	NO	NO	YES	USB2_OC0# (Mode1)	GPI	No	
GPP_E10	NO	NO	YES	USB2_OC1# (Mode1)	GPI	No	
GPP_E11	NO	NO	YES	USB2_OC2# (Mode1)	GPI	No	
GPP_E12	NO	NO	YES	USB2_OC3# (Mode1)	GPI	No	
Group F GPIO - Primary Power Well (1.8V or 3.3V)							
GPP_F0	NO	NO	YES	SATAXPCIE3 (Mode1) SATAGP3 (Mode3)	SATAXPCIE3 GPI	No	Default depends on soft strap.
GPP_F1	NO	NO	YES	SATATXPCIE4 (Mode1) SATAGP4 (Mode3)	SATAXPCIE4 GPI	No	Default depends on soft strap.
GPP_F2	NO	NO	YES	SATAXPCIE5 (Mode1) SATAGP5 (Mode3)	SATAXPCIE5 GPI	No	Default depends on soft strap.
GPP_F3	NO	NO	YES	SATAXPCIE6 (Mode1) SATAGP6 (Mode 3)	SATAXPCIE6 GPI	No	Default depends on soft strap.
GPP_F4	NO	NO	YES	SATAXPCIE7 (Mode1) SATAGP7 (Mode3)	SATAXPCIE7 GPI	No	Default depends on soft strap.
GPP_F5	NO	NO	YES	SATA_DEVSLP3 (Mode1)	GPI	No	
GPP_F6	NO	NO	YES	SATA_DEVSLP4 (Mode1)	GPI	No	
GPP_F7	NO	NO	YES	SATA_DEVSLP5 (Mode1)	GPI	No	
GPP_F8	NO	NO	YES	SATA_DEVSLP6 (Mode1)	GPI	No	
GPP_F9	NO	NO	YES	SATA_DEVSLP7 (Mode1)	GPI	No	
GPP_F10	NO	NO	YES	SATA_SCLOCK (Mode1)	GPI	No	
GPP_F11	NO	NO	YES	SATA_SLOAD (Mode1)	GPI	No	
GPP_F12	NO	NO	YES	SATA_SDATAOUT1 (Mode1)	GPI	No	
GPP_F13	NO	NO	YES	SATA_SDATAOUT0 (Mode1)	GPI	No	
GPP_F14	NO	NO	YES	SSATA_LED# (Mode3)	GPI	No	
GPP_F15	NO	NO	YES	USB2_OC4# (Mode1)	GPI	No	
GPP_F16	NO	NO	YES	USB2_OC5# (Mode1)	GPI	No	
GPP_F17	NO	NO	YES	USB2_OC6# (Mode1)	GPI	No	
GPP_F18	NO	NO	YES	USB2_OC7# (Mode1)	GPI	No	
GPP_F19	NO	NO	YES	LAN_SMBCLK (Mode1)	LAN_SMBCLK	No	
GPP_F20	NO	NO	YES	LAN_SMBDATA (Mode1)	LAN_SMBDATA	No	
GPP_F21	NO	NO	YES	LAN_SMBALRT# (Mode1)	LAN_SMBALRT#	No	
GPP_F22	NO	NO	YES	SSATA_SCLOCK (Mode3)	GPI	No	
GPP_F23	NO	NO	YES	SSATA_SLOAD (Mode3)	GPI	No	
Group G GPIO - Primary Power Well (1.8V or 3.3V)							
GPP_G0	NO	YES	YES	FANTACH0 (Mode1) FANTACH0IE (Mode2)	GPI	No	
GPP_G1	NO	YES	YES	FANTACH1 (Mode1) FANTACH1IE (Mode2)	GPI	No	
GPP_G2	NO	YES	YES	FANTACH2 (Mode1) FANTACH2IE (Mode2)	GPI	No	

Table 18-2. General Purpose I/O Signals (Sheet 6 of 10)

Name	Internal Pull-Up/ Pull-Down (Note 1)	De-Glitch (Note 2)		Multiplexed With	Default	NMI or SMI Capable	Note
		Input	Output				
GPP_G3	NO	YES	YES	FANTACH3 (Mode1) FANTACH3IE (Mode2)	GPI	No	
GPP_G4	NO	YES	YES	FANTACH4 (Mode1) FANTACH4IE (Mode2)	GPI	No	
GPP_G5	NO	YES	YES	FANTACH5 (Mode1) FANTACH5IE (Mode2)	GPI	No	
GPP_G6	NO	YES	YES	FANTACH6 (Mode1) FANTACH6IE (Mode2)	GPI	No	
GPP_G7	NO	YES	YES	FANTACH7 (Mode1) FANTACH7IE (Mode2)	GPI	No	
GPP_G8	NO	NO	YES	FANPWM0 (Mode1) FANPWM0IE (Mode2)	FANPWM0_ FANPWM0IE	No	
GPP_G9	NO	NO	YES	FANPWM1 (Mode1) FANPWM1IE (Mode2)	FANPWM1_ FANPWM1IE	No	
GPP_G10	NO	NO	YES	FANPWM2 (Mode1) FANPWM2IE (Mode2)	FANPWM2_ FANPWM2IE	No	
GPP_G11	NO	NO	YES	FANPWM3 (Mode1) FANPWM3IE (Mode2)	FANPWM3_ FANPWM3IE	No	
GPP_G12	NO	NO	YES	NONE	GPI	No	
GPP_G13	NO	NO	YES	NONE	GPI	No	
GPP_G14	NO	NO	YES	NONE	GPI	No	
GPP_G15	NO	NO	YES	NONE	GPI	No	
GPP_G16	NO	NO	YES	NONE	GPI	No	
GPP_G17	NO	NO	YES	ADR_COMPLETE (mode1)	GPI	No	
GPP_G18	NO	NO	YES	NMI# (Mode1)	GPI	No	
GPP_G19	NO	NO	YES	SMI# (Mode1)	GPI	No	
GPP_G20	NO	NO	YES	SSATA_DEVSLP0 (Mode2)	GPI	No	
GPP_G21	NO	NO	YES	SSATA_DEVSLP1 (Mode2)	GPI	No	
GPP_G22	NO	NO	YES	SSATA_DEVSLP2 (Mode2)	GPI	No	
GPP_G23	NO	NO	YES	SSATAXPCIE0 (Mode2) SSATAGP0 (Mode3)	GPI	No	
Group H GPIO - Primary Power Well (1.8V or 3.3V)							
GPP_H0	NO	NO	YES	SRCCLKREQ6# (Mode1)	GPI	No	
GPP_H1	NO	NO	YES	SRCCLKREQ7# (Mode1)	GPI	No	
GPP_H2	NO	NO	YES	SRCCLKREQ8# (Mode1)	GPI	No	
GPP_H3	NO	NO	YES	SRCCLKREQ9# (Mode1)	GPI	No	
GPP_H4	NO	NO	YES	SRCCLKREQ10# (Mode1)	GPI	No	
GPP_H5	NO	NO	YES	SRCCLKREQ11# (Mode1)	GPI	No	
GPP_H6	NO	NO	YES	SRCCLKREQ12# (Mode1)	GPI	No	
GPP_H7	NO	NO	YES	SRCCLKREQ13# (Mode1)	GPI	No	
GPP_H8	NO	NO	YES	SRCCLKREQ14# (Mode1)	GPI	No	

Table 18-2. General Purpose I/O Signals (Sheet 7 of 10)

Name	Internal Pull-Up/ Pull-Down (Note 1)	De-Glitch (Note 2)		Multiplexed With	Default	NMI or SMI Capable	Note
		Input	Output				
GPP_H9	NO	NO	YES	SRCCLKREQ15# (Mode1)	GPI	No	
GPP_H10	NO	YES	YES	SML2CLK (Mode1) SML2CLK_IE (Mode2)	GPI	No	
GPP_H11	NO	YES	YES	SML2DATA (Mode1) SML2DATA_IE (Mode2)	GPI	No	
GPP_H12	20K PD	YES	YES	SML2ALERT# (Mode1) SML2ALERT#_IE# (Mode2)	GPO	No	Slave Attached Flash strap
GPP_H13	NO	YES	YES	SML3CLK (Mode1) SML3CLK_IE (Mode2)	GPI	No	
GPP_H14	NO	YES	YES	SML3DATA (Mode1) SML3DATA_IE (Mode2)	GPI	No	
GPP_H15	20K PU	YES	YES	SML3ALERT# (Mode1) SML3ALERT#_IE# (Mode2)	GPI	No	ADR Timer hold-off strap
GPP_H16	NO	YES	YES	SML4CLK (Mode1) SML4CLK_IE (Mode2)	GPI	No	
GPP_H17	NO	YES	YES	SML4DATA (Mode1) SML4DATA_IE (Mode2)	GPI	No	
GPP_H18	20K PU	YES	YES	SML4ALERT# (Mode1) SML4ALERT#_IE# (Mode2)	GPI	No	Reserved pin strap. See Table 3-1.
GPP_H19	NO	YES	YES	SSATAXPCIE1 (Mode2) SSATAGP1 (Mode3)	SSATAXPCIEI GPI	No	Default depends on soft strap.
GPP_H20	NO	YES	YES	SSATAXPCIE2 (Mode2) SSATAGP2 (Mode3)	SSATAXPCIE2 GPI	No	Default depends on soft strap.
GPP_H21	NO	YES	YES	SSATAXPCIE3 (Mode2) SSATAGP3 (Mode3)	SSATAXPCIE3 GPI	No	Default depends on soft strap.
GPP_H22	NO	YES	YES	SSATAXPCIE4 (Mode2) SSATAGP4 (Mode3	SSATAXPCIE4 GPI	No	Default depends on soft strap.
GPP_H23	NO	NO	YES	SSATAXPCIE5 (Mode2) SSATAGP5 (Mode3)	SSATAXPCIE5 GPI	No	Default depends on soft strap.
Group I GPIO - Primary Power Well (3.3V Only)							
GPP_I0	NO	NO	YES	LAN_TDO (Mode2)	LAN_TDO	SMI/NMI	
GPP_I1	NO	NO	YES	LAN_TCK (Mode2)	LAN_TCK	SMI/NMI	
GPP_I2	NO	NO	YES	LAN_TMS (Mode2)	LAN_TMS	SMI/NMI	
GPP_I3	NO	NO	YES	LAN_TDI (Mode2)	LAN_TDI	SMI/NMI	
GPP_I4	NO	NO	YES	RESET_IN# (Mode2)	GPI	No	
GPP_I5	NO	YES	YES	RESET_OUT# (Mode2)	GPI	No	
GPP_I6	NO	YES	YES	RESET_DONE (Mode2)	GPI	No	
GPP_I7	NO	YES	YES	LAN_TRST_IN(Mode2)	LAN_TRST_IN	No	
GPP_I8	NO	YES	YES	PCI_DIS (Mode2)	PCI_DIS	No	
GPP_I9	NO	YES	YES	LAN_DIS (Mode2)	LAN_DIS	No	
GPP_I10	NO	YES	YES	NONE	GPI	No	
Group J GPIO - Primary Power Well (1.8V or 3.3V Only)							
GPP_J0	NO	YES	YES	LAN_LED_P0_0 (Mode1)	LAN_LED_P0_0	No	Default state is output.
GPP_J1	NO	YES	YES	LAN_LED_P0_1 (Mode1)	LAN_LED_P0_1	No	Default state is output.

Table 18-2. General Purpose I/O Signals (Sheet 8 of 10)

Name	Internal Pull-Up/ Pull-Down (Note 1)	De-Glitch (Note 2)		Multiplexed With	Default	NMI or SMI Capable	Note
		Input	Output				
GP_J2	NO	YES	YES	LAN_LED_P1_0 (Mode1)	LAN_LED_P1_0	No	Default state is output.
GPP_J3	NO	YES	YES	LAN_LED_P1_1 (Mode1)	LAN_LED_P1_1	No	Default state is output.
GPP_J4	NO	YES	YES	LAN_LED_P2_0 (Mode1)	LAN_LED_P2_0	No	Default state is output.
GPP_J5	NO	YES	YES	LAN_LED_P2_1 (Mode1)	LAN_LED_P2_1	No	Default state is output.
GPP_J6	NO	YES	YES	LAN_LED_P3_0 (Mode1)	LAN_LED_P3_0	No	Default state is output.
GPP_J7	NO	YES	YES	LAN_LED_P3_1 (Mode1)	LAN_LED_P3_1	No	Default state is output.
GPP_J8	NO	YES	YES	LAN_I2C_SCL_MDC_P0 (Mode1)	LAN_I2C_SCL_ MDC_P0	No	Default state is OD.
GPP_J9	NO	YES	YES	LAN_I2C_SDA_MDIO_P0 (Mode1)	LAN_I2C_SDA_ MDIO_P0	No	Default state is OD.
GPP_J10	NO	YES	YES	LAN_I2C_SCL_MDC_P1 (Mode1)	LAN_I2C_SCL_ MDC_P1	No	Default state is OD.
GPP_J11	NO	YES	YES	LAN_I2C_SDA_MDIO_P1 (Mode1)	LAN_I2C_SDA_ MDIO_P1	No	Default state is OD.
GPP_J12	NO	YES	YES	LAN_I2C_SCL_MDC_P2 (Mode1)	LAN_I2C_SCL_ MDC_P2	No	Default state is OD.
GPP_J13	NO	YES	YES	LAN_I2C_SDA_MDIO_P2 (Mode1)	LAN_I2C_SDA_ MDIO_P2	No	Default state is OD.
GPP_J14	NO	YES	YES	LAN_I2C_SCL_MDC_P3 (Mode1)	LAN_I2C_SCL_ MDC_P3	No	Default state is OD.
GPP_J15	NO	YES	YES	LAN_I2C_SDA_MDIO_P3 (Mode1)	LAN_I2C_SDA_ MDIO_P3	No	Default state is OD.
GPP_J16	NO	YES	YES	LAN_SDP_P0_0 (Mode1)	LAN_SDP_P0_0	No	
GPP_J17	NO	YES	YES	LAN_SDP_P0_1 (Mode1)	LAN_SDP_P0_1	No	
GPP_J18	NO	YES	YES	LAN_SDP_P1_0 (Mode1)	LAN_SDP_P1_0	No	
GPP_J19	NO	YES	YES	LAN_SDP_P1_1 (Mode1)	LAN_SDP_P1_1	No	
GPP_J20	NO	YES	YES	LAN_SDP_P2_0 (Mode1)	LAN_SDP_P2_0	No	
GPP_J21	NO	YES	YES	LAN_SDP_P2_2 (Mode1)	LAN_SDP_P2_2	No	
GPP_J22	NO	YES	YES	LAN_SDP_P3_0 (Mode1)	LAN_SDP_P3_0	No	
GPP_J23	NO	YES	YES	LAN_SDP_P3_1 (Mode1)	LAN_SDP_P3_1	No	
Group K GPIO - Primary Power Well (1.8V or 3.3V Only)							
GPP_K0	NO	YES	YES	LAN_NCSI_CLK_IN (Mode1)	LAN_NCSI_CLK_ IN	No	Default state is input.
GPP_K1	NO	YES	YES	LAN_NCSI_TXD0 (Mode1)	LAN_NCSI_TXD0	No	Default state is input.
GPP_K2	NO	YES	YES	LAN_NCSI_TXD1 (Mode1)	LAN_NCSI_TXD1	No	Default state is input.
GPP_K3	NO	YES	YES	LAN_NCSI_TX_EN (Mode1)	LAN_NCSI_TX_E N	No	Default state is input.
GPP_K4	NO	YES	YES	LAN_NCSI_CRS_DV (Mode1)	LAN_NCSI_CRS_ DV	No	Default state is output.
GPP_K5	NO	YES	YES	LAN_NCSI_RXD0(Mode1)	LAN_NCSI_RXD0	No	Default state is output.
GPP_K6	NO	YES	YES	LAN_NCSI_RXD1 (Mode1)	LAN_NCSI_RXD1	No	Default state is output.

Table 18-2. General Purpose I/O Signals (Sheet 9 of 10)

Name	Internal Pull-Up/ Pull-Down (Note 1)	De-Glitch (Note 2)		Multiplexed With	Default	NMI or SMI Capable	Note
		Input	Output				
GPP_K7	20K PD	YES	YES	RSVD (Mode1)	RSVD	No	Debug strap. Needs to be low on rising edge of RSMRST#.
GPP_K8	NO	YES	YES	LAN_NCSI_ARB_IN (Mode1)	LAN_NCSI_ARB_IN	No	
GPP_K9	NO	YES	YES	LAN_NCSI_ARB_OUT (Mode1)	LAN_NCSI_ARB_OUT	No	
GPP_K10	NO	YES	YES	PE_RST# (Mode1)	PE_RST#	No	
Group L GPIO - Primary Power Well (1.05 Only)							
GPP_L2	NO	NO	YES	TESTCH0_D0 (Mode1)	TESTCH0_D0	No	
GPP_L3	NO	NO	YES	TESTCH0_D1 (Mode1)	TESTCH0_D1	No	
GPP_L4	NO	NO	YES	TESTCH0_D2 (Mode1)	TESTCH0_D2	No	
GPP_L5	NO	NO	YES	TESTCH0_D3 (Mode1)	TESTCH0_D3	No	
GPP_L6	NO	NO	YES	TESTCH0_D4 (Mode1)	TESTCH0_D4	No	
GPP_L7	NO	NO	YES	TESTCH0_D5 (Mode1)	TESTCH0_D5	No	
GPP_L8	NO	NO	YES	TESTCH0_D6 (Mode1)	TESTCH0_D6	No	
GPP_L9	NO	NO	YES	TESTCH0_D7 (Mode1)	TESTCH0_D7	No	
GPP_L10	NO	NO	YES	TESTCH0_CLK (Mode1)	TESTCH0_CLK	No	
GPP_L11	NO	NO	YES	TESTCH1_D0 (Mode1)	TESTCH1_D0	No	
GPP_L12	NO	NO	YES	TESTCH1_D1 (Mode1)	TESTCH1_D1	No	
GPP_L13	NO	NO	YES	TESTCH1_D2 (Mode1)	TESTCH1_D2	No	
GPP_L14	NO	NO	YES	TESTCH1_D3 (Mode1)	TESTCH1_D3	No	
GPP_L15	NO	NO	YES	TESTCH1_D4 (Mode1)	TESTCH1_D4	No	
GPP_L16	NO	NO	YES	TESTCH1_D5 (Mode1)	TESTCH1_D5	No	
GPP_L17	NO	NO	YES	TESTCH1_D6 (Mode1)	TESTCH1_D6	No	
GPP_L18	NO	NO	YES	TESTCH1_D7 (Mode1)	TESTCH1_D7	No	
GPP_L19	NO	NO	YES	TESTCH1_CLK (Mode1)	TESTCH1_CLK	No	
GPIO In Deep Sleep Power Well (3.3V Only)							
GPD0	NO	NO	YES	POWER_DEBUG_N	POWER_DEBUG_N	No	
GPD1	NO	NO	YES	ACPRESENT (Mode1)	ACPRESENT	No	
GPD2	NO	NO	YES	GBE_WAKE# (Mode1)	GBE_WAKE#	No	
GPD3	20K PU	YES	YES	PWRBTN# (Mode1)	PWRBTN#	No	Bit 28 (PB_DIS) of the PM_CFG2 register must be set to 1b in order for this pin to be used as a GPIO.
GPD4	NO	NO	YES	SLP_S3# (Mode1)	SLP_S3#	No	
GPD5	NO	NO	YES	SLP_S4# (Mode1)	SLP_S4#	No	
GPD6	NO	NO	YES	SLP_A# (Mode1)	SLP_A#	No	
GPD7	NO	NO	YES	RSVD (Mode1)	RSVD	No	Default is 1.
GPD8	NO	NO	YES	SUSCLK (Mode1)	SUSCLK	No	
GPD9	NO	NO	YES	reserved	reserved	No	Default is 0.
GPD10	NO	NO	YES	SLP_S5# (Mode1)	SLP_S5#	No	
GPD11	NO	NO	YES	GBEPHY (Mode1)	GBEPHY	No	
GPIO RCOMP Pins							

Table 18-2. General Purpose I/O Signals (Sheet 10 of 10)

Name	Internal Pull-Up/ Pull-Down (Note 1)	De-Glitch (Note 2)		Multiplexed With	Default	NMI or SMI Capable	Note
		Input	Output				
GPIO_RCOMP3_3	NA	NA	NA	NA	NA	NA	RCOMP for 3.3V. This is pin BY25.
GPIO_RCOMP_1P8_3_3	NA	NA	NA	NA	NA	NA	RCOMP for 1.8 and 3.3V. This is pin AM4.

Notes:
1. All GPIOs have weak internal pull-up or pull-down resistors but the resistors are off by default unless noted in this column.
2. De-glitch signal has a 20K Ohm pull-down during the pin power sequencing with the exception of the GPP_B0_CORE_VID0, and GPP_B1_CORE_VID1 which have 20K pull-ups on them.
3. LPC mode and eSPI mode are determined by HW ESPI Enable Strap. See the pin strap section for more detail.
4. There are up to 3 non-GPIO Modes possible. BIOS refers to these as Mode1, Mode2, Mode3 and the use of the terms here is to equate which function is selected when BIOS selects a mode for the GPIO.
5. For GPIOS whose default is a RSVD function, the BIOS should reconfigure these pins to GPIOs or have them as no connects.

18.6 GPIO Community 0 Registers Summary

Community 0 Registers are for GPP_A, GPP_B and GPP_F groups.

Table 18-3. Summary of GPIO Community 0 Registers (Sheet 1 of 5)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
Ch	Fh	Pad Base Address (PADBAR)—Offset Ch	00000400h
10h	13h	Miscellaneous Configuration (MISCCFG)—Offset 10h	00051000h
20h	23h	Pad Ownership (PAD_OWN_GPP_A_0)—Offset 20h	00000000h
24h	27h	Pad Ownership (PAD_OWN_GPP_A_1)—Offset 24h	00000000h
28h	2Bh	Pad Ownership (PAD_OWN_GPP_A_2)—Offset 28h	00000000h
2C	2F	Pad Ownership (PAD_OWN_GPP_B_0)—Offset 2Ch	00000000h
30	33	Pad Ownership (PAD_OWN_GPP_B_1)—Offset 30h	00000000h
34	37	Pad Ownership (PAD_OWN_GPP_B_2)—Offset 34h	00000000h
38	3B	Pad Ownership (PAD_OWN_GPP_F_0)—Offset 38h	00000000h
3C	3F	Pad Ownership (PAD_OWN_GPP_F_1)—Offset 3Ch	00000000h
40	43	Pad Ownership (PAD_OWN_GPP_F_2)—Offset 40h	00000000h
60h	63	Pad Configuration Lock (PADCFGLOCK_GPP_A)—Offset 60h	00000000h
64	67	Pad Configuration Lock (PADCFGLOCKTX_GPP_A)—Offset 64h	00000000h
68	6B	Pad Configuration Lock (PADCFGLOCK_GPP_B)—Offset 68h	00000000h
6C	6F	Pad Configuration Lock (PADCFGLOCKTX_GPP_B)—Offset 6Ch	00000000h
70	73	Pad Configuration Lock (PADCFGLOCK_GPP_F)—Offset 70h	00000000h
74	77	Pad Configuration Lock (PADCFGLOCKTX_GPP_F)—Offset 74h	00000000h
80h	83h	Host Software Pad Ownership (HOSTSW_OWN_GPP_A)—Offset 80h	00000000h
84h	87h	Host Software Pad Ownership (HOSTSW_OWN_GPP_B)—Offset 84h	00000000h
88	8Bh	Host Software Pad Ownership (HOSTSW_OWN_GPP_F)—Offset 88h	00000000h
100h	103h	GPI Interrupt Status (GPI_IS_GPP_A)—Offset 100h	00000000h
104h	107h	GPI Interrupt Status (GPI_IS_GPP_B)—Offset 104h	00000000h
108h	10Bh	GPI Interrupt Status (GPI_IS_GPP_F)—Offset 108h	00000000h
110h	113h	GPI Interrupt Enable (GPI_IE_GPP_A)—Offset 110h	00000000h
114h	117h	GPI Interrupt Enable (GPI_IE_GPP_B)—Offset 114h	00000000h

Table 18-3. Summary of GPIO Community 0 Registers (Sheet 2 of 5)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
118h	11Ch	GPI Interrupt Enable (GPI_IE_GPP_F)—Offset 118h	00000000h
120h	123h	GPI General Purpose Events Status (GPI_GPE_STS_GPP_A)—Offset 120h	00000000h
124h	127h	GPI General Purpose Events Status (GPI_GPE_STS_GPP_B)—Offset 124h	00000000h
128h	12Ch	GPI General Purpose Events Status (GPI_GPE_STS_GPP_F)—Offset 128h	00000000h
130h	133h	GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A)—Offset 130h	00000000h
134h	137h	GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B)—Offset 134h	00000000h
138h	13Bh	GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F)—Offset 133h	00000000h
144h	147h	SMI Status (GPI_SMI_STS_GPP_B)—Offset 144h	00000000h
154h	157h	SMI Enable (GPI_SMI_EN_GPP_B)—Offset 154h	00000000h
164h	167h	NMI Status (GPI_NMI_STS_GPP_B)—Offset 164h	00000000h
174h	177h	NMI Enable (GPI_NMI_EN_GPP_B)—Offset 174h	00000000h
400h	403h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_0)—Offset 400h	44000700h
404h	407h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_0)—Offset 404h	00000018h
408h	40Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_1)—Offset 408h	44000700h
40Ch	40Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_1)—Offset 40Ch	000003C9
410h	413h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_2)—Offset 410h	44000700h
414h	417h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_2)—Offset 414h	00003C1A
418h	41Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_3)—Offset 418h	44000700h
41Ch	41Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_3)—Offset 41Ch	00003C1B
420h	423h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_4)—Offset 420h	44000700h
424h	427h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_4)—Offset 424h	00003C1C
428h	42Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_5)—Offset 428h	44000700h
42Ch	42Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_5)—Offset 42Ch	0000001D
430h	433h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_6)—Offset 430h	44000700h
434h	437h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_6)—Offset 434h	0000001Eh
438h	43Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_7)—Offset 438h	44000700h
43Ch	43Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_7)—Offset 43Ch	0000001Fh
440h	443h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_8)—Offset 440h	44000700h
444h	447h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_8)—Offset 444h	00000020h
448h	44Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_9)—Offset 448h	44000700h
44Ch	44Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_9)—Offset 44Ch	00001021h
450h	453h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_10)—Offset 450h	44000700h
454h	457h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_10)—Offset 454h	00001022h
458h	45Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_11)—Offset 458h	44000700h
45Ch	45Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_11)—Offset 45Ch	00003023
460h	463h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_12)—Offset 460h	44000300h
464h	467h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_12)—Offset 464h	00000024
468h	46Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_13)—Offset 468h	44000700h
46Ch	46Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_13)—Offset 46Ch	00000025
470h	473h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_14)—Offset 470h	44000300h
474h	477h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_14)—Offset 474h	00000026

Table 18-3. Summary of GPIO Community 0 Registers (Sheet 3 of 5)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
478h	47Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_15)—Offset 478h	44000700h
47Ch	47Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_15)—Offset 47Ch	00003027h
480h	483h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_16)—Offset 480h	44000700h
484h	487h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_16)—Offset 484h	00001028h
488h	48Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_17)—Offset 488h	44000300h
48Ch	48Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_17)—Offset 48Ch	00000029h
490h	493h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_18)—Offset 490h	44000300h
494h	497h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_18)—Offset 494h	0000002Ah
498h	49Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_19)—Offset 498h	44000300h
49Ch	49Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_19)—Offset 49Ch	0000002Bh
4A0h	4A3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_20)—Offset 4A0h	44000200h
4A4h	4A7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_20)—Offset 4A4h	0000002Ch
4A8h	4ABh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_21)—Offset 4A8h	44000200h
4ACh	4AFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_21)—Offset 4ACh	0000002Dh
4B0h	4B3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_22)—Offset 4B0h	44000200h
4B4h	4B7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_22)—Offset 4B4h	0000002Eh
4B8h	4BBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_23)—Offset 4B8h	44000200h
4BCh	4BFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_23)—Offset 4BCh	0000002Fh
4C0h	4C3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_0)—Offset 4C0h	44000600h
4C4h	4C7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_0)—Offset 4C4h	00003030h
4C8h	4CBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_1)—Offset 4C8h	44000600h
4CCh	4CFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_1)—Offset 4CCh	00003031h
4D0h	4D3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_2)—Offset 4D0h	44000300h
4D4h	4D7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_2)—Offset 4D4h	00000032h
4D8h	4DBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_3)—Offset 4D8h	44000300h
4DCh	4DFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_3)—Offset 4DCh	00000033h
4E0h	4E3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_4)—Offset 4E0h	44000300h
4E4h	4E7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_4)—Offset 4E4h	00000034h
4E8h	4EBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_5)—Offset 4E8h	44000300h
4ECh	4EFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_5)—Offset 4ECh	00000035h
4F0h	4F3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_6)—Offset 4F0h	44000300h
4F4h	4F7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_6)—Offset 4F4h	00000036h
4F8h	4FBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_7)—Offset 4F8h	44000300h
4FCh	4FFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_7)—Offset 4FCh	00000037h
500h	503h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_8)—Offset 500h	44000300h
504h	507h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_8)—Offset 504h	00000038h
508h	50Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_9)—Offset 508h	44000300h
50Ch	50Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_9)—Offset 50Ch	00000039h
510h	513h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_10)—Offset 510h	44000300h
514h	517h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_10)—Offset 514h	0000003Ah
518h	51Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_11)—Offset 518h	44000200h

Table 18-3. Summary of GPIO Community 0 Registers (Sheet 4 of 5)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
51Ch	51Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_11)—Offset 51Ch	0000003Bh
520h	523h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_12)—Offset 520h	44000700h
524h	527h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_12)—Offset 524h	0000003Ch
528h	52Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_13)—Offset 528h	44000700h
52Ch	52Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_13)—Offset 52Ch	0000003Dh
530h	533h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_14)—Offset 530h	44000200h
534h	537h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_14)—Offset 534h	0000003Eh
538h	53Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_15)—Offset 538h	44000300h
53Ch	53Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_15)—Offset 53Ch	0000003Fh
540h	543h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_16)—Offset 540h	44000300h
544h	547h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_16)—Offset 544h	00000040h
548h	54Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_17)—Offset 548h	44000300h
54Ch	54Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_17)—Offset 54Ch	00000041h
550h	553h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_18)—Offset 550h	44000200h
554h	557h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_18)—Offset 554h	00000042h
558h	55Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_19)—Offset 558h	44000300h
55Ch	55Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_19)—Offset 55Ch	00000043h
560h	563h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_20)—Offset 560h	44000300h
564h	567h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_20)—Offset 564h	00000044h
568h	56Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_21)—Offset 568h	44000300h
56Ch	56Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_21)—Offset 56Ch	00000045h
570h	573h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_22)—Offset 570h	44000200h
574h	577h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_22)—Offset 574h	00000046h
578h	57Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_23)—Offset 578h	44000200h
57Ch	57Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_23)—Offset 57Ch	00000047h
580h	583h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_0)—Offset 580h	44000300h
584h	587h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_0)—Offset 584h	00000055h
588h	58Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_1)—Offset 588h	44000300h
58Ch	58Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_1)—Offset 58Ch	00000056h
590h	593h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_2)—Offset 590h	44000300h
594h	597h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_2)—Offset 594h	00000057h
598h	59Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_3)—Offset 598h	44000300h
59Ch	59Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_3)—Offset 59Ch	00000058h
5A0h	5A3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_4)—Offset 5A0h	44000300h
5A4h	5A7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_4)—Offset 5A4h	00000059h
5A8h	5ABh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_5)—Offset 5A8h	44000300h
5ACh	5AFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_5)—Offset 5ACh	0000005Ah
5B0h	5B3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_6)—Offset 5B0h	44000300h
5B4h	5B7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_6)—Offset 5B4h	0000005Bh
5B8h	5BBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_7)—Offset 5B8h	44000300h
5BCh	5BFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_7)—Offset 5BCh	0000005Ch

Table 18-3. Summary of GPIO Community 0 Registers (Sheet 5 of 5)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
5C0h	5C3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_8)—Offset 5C0h	44000300h
5C4h	5C7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_8)—Offset 5C4h	0000005Dh
5C8h	5CBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_9)—Offset 5C8h	44000300h
5CCh	5CFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_9)—Offset 5CCh	0000005Eh
5D0h	5D3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_10)—Offset 5D0h	44000300h
5D4h	5D7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_10)—Offset 5D4h	0000005Fh
5D8h	5DBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_11)—Offset 5D8h	44000300h
5DCh	5DFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_11)—Offset 5DCh	00000060h
5E0h	5E3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_12)—Offset 5E0h	44000300h
5E4h	5E7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_12)—Offset 5E4h	00000061h
5E8h	5EBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_13)—Offset 5E8h	44000300h
5ECh	5EFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_13)—Offset 5ECh	00000062h
5F0h	5F3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_14)—Offset 5F0h	44000300h
5F4h	5F7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_14)—Offset 5F4h	00000063h
5F8h	5FBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_15)—Offset 5F8h	44000300h
5FCh	5FFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_15)—Offset 5FCh	00000064h
600h	603h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_16)—Offset 600h	44000300h
604h	607h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_16)—Offset 604h	00000065h
608h	60Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_17)—Offset 608h	44000300h
60Ch	60Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_17)—Offset 60Ch	00000066h
610h	613h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_18)—Offset 610h	44000300h
614h	617h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_18)—Offset 614h	00000067h
618h	61Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_19)—Offset 618h	44000700h
61Ch	61Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_19)—Offset 61Ch	00000068h
620h	623h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_20)—Offset 620h	44000700h
624h	627h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_20)—Offset 624h	00000069h
628h	62Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_21)—Offset 628h	44000700h
62Ch	62Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_21)—Offset 62Ch	0000006Ah
630h	633h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_22)—Offset 630h	44000300h
634h	637h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_22)—Offset 634h	0000006Bh
638h	63Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_23)—Offset 638h	44000300h
63Ch	63Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_23)—Offset 63Ch	0000006Ch

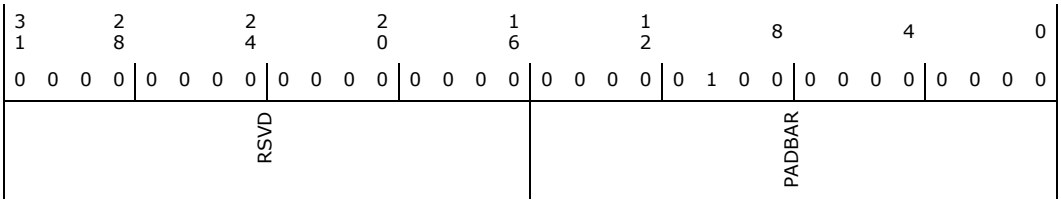
18.6.1 Pad Base Address (PADBAR)—Offset Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 400h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:0	400h RO	**Pad Base Address (PADBAR):** This field provides the starting byte-align address of Pad0 register sets. It is meant for software to discover from where the very first Pad register (i.e., Pad0 register) starts to compute the next Pad address offsets.

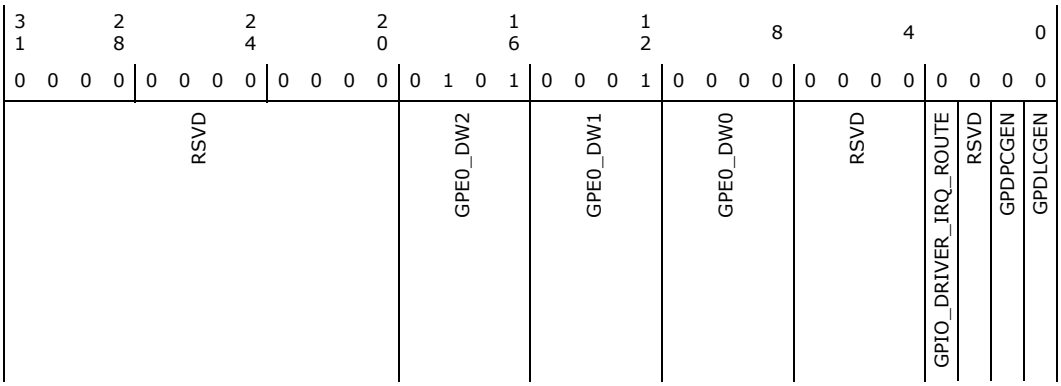
18.6.2 Miscellaneous Configuration (MISCCFG)—Offset 10h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00051000h



Bit Range	Default & Access	Field Name (ID): Description
31:20	0h RO	Reserved
19:16	5h RO	**GPIO Group to GPE_DW2 assignment encoding (GPE0_DW2):** This register assigns a specific GPIO Group to the ACPI GPE0[95:64]. 0h = GPP_A[23:0] mapped to GPE[87:64] and GPE[95:88] is not used 1h = GPP_B[23:0] mapped to GPE[87:64] and GPE[95:88] is not used 2h = GPP_C[23:0] mapped to GPE[87:64] and GPE[95:88] is not used] 3h = GPP_D[23:0] mapped to GPE[87:64] and GPE[95:88] is not used 4h = GPP_E[12:0] mapped to GPE[76:64] and GPE[95:77] is not used 5h = GPP_F[23:0] mapped to GPE[87:64] and GPE[95:88] is not used 6h = GPP_G[23:0] mapped to GPE[87:64] and GPE[95:88] is not used 7h = GPP_H[23:0] mapped to GPE[87:64] and GPE[95:88] is not used 8h = GPP_I[10:0] mapped to GPE[74:64] and GPE[95:75] is not used 9h = GPD[11:0] mapped to GPE[75:64] and GPE[95:76] is not used Ah= GPP_J[23:0] mapped to GPE[87:64] and GPE[95:88] not used Bh= GPP_K[11:0]mapped to GPE[75:64] and GPE[95:75] is not used
15:12	1h RO	**GPIO Group to GPE_DW1 assignment encoding (GPE0_DW1):** This register assigns a specific GPIO Group to the ACPI GPE0[63:32]. 0h = GPP_A[23:0] mapped to GPE[55:32] and GPE[63:56] is not used 1h = GPP_B[23:0] mapped to GPE[55:32] and GPE[63:56] is not used 2h = GPP_C[23:0] mapped to GPE[55:32] and GPE[63:56] is not used] 3h = GPP_D[23:0] mapped to GPE[55:32] and GPE[63:56] is not used 4h = GPP_E[12:0] mapped to GPE[44:32] and GPE[63:45] is not used 5h = GPP_F[23:0] mapped to GPE[55:32] and GPE[63:56] is not used 6h = GPP_G[23:0] mapped to GPE[55:32] and GPE[63:56] is not used 7h = GPP_H[23:0] mapped to GPE[55:32] and GPE[63:56] is not used 8h = GPP_I[10:0] mapped to GPE[42:32] and GPE[63:43] is not used 9h = GPD[11:0] mapped to GPE[43:32] and GPE[63:44] is not used Ah= GPP_J[23:0] mapped to GPE[55:32] and GPE[63:56] not used Bh= GPP_K[11:0]mapped to GPE[43:32] and GPE[63:44] is not used
11:8	0h RO	**GPIO Group to GPE_DW0 assignment encoding (GPE0_DW0):** This register assigns a specific GPIO Group to the ACPI GPE0[31:0]. 0h = GPP_A[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 1h = GPP_B[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 2h = GPP_C[23:0] mapped to GPE[23:0] and GPE[31:24] is not used] 3h = GPP_D[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 4h = GPP_E[12:0] mapped to GPE[12:0] and GPE[31:13] is not used 5h = GPP_F[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 6h = GPP_G[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 7h = GPP_H[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 8h = GPP_I[10:0] mapped to GPE[10:0] and GPE[31:11] is not used 9h = GPD[11:0] mapped to GPE[11:0] and GPE[31:12] is not used Ah= GPP_J[23:0] mapped to GPE[23:0] and GPE[31:24] not used Bh= GPP_K[11:0]mapped to GPE[11:0] and GPE[31:12] is not used
7:4	0h RO	Reserved
3	0h RW	**GPIO Driver IRQ Route (GPIO_DRIVER_IRQ_ROUTE):** Specifies the APIC IRQ globally for all pads within the current community (GPI_IS with corresponding GPI_IE enable). 0 = IRQ14 1 = IRQ15
2	0h RO	Reserved
1	0h RW	**GPIO Dynamic Partition Clock Gating Enable (GPDPCGEN):** Specifies whether the GPIO Community should take part in partition clock gating. 0 = Disable participation in dynamic partition clock gating. 1 = Enable participation in dynamic partition clock gating.
0	0h RW	**GPIO Dynamic Local Clock Gating Enable (GPDLCGEN):** Specifies whether the GPIO Community should perform local clock gating. 0 = Disable dynamic local clock gating. 1 = Enable dynamic local clock gating.

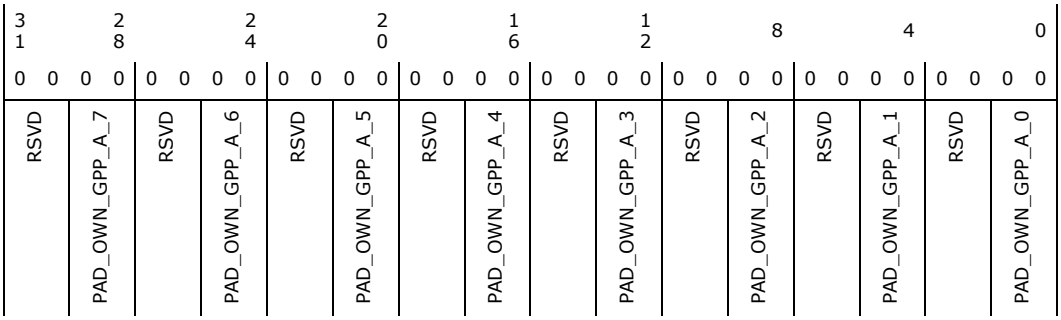
18.6.3 Pad Ownership (PAD_OWN_GPP_A_0)—Offset 20h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RO	**Pad Ownership (PAD_OWN_GPP_A_7):** Same description as bit 0, except that the bit field applies to GPP_A7.
27:26	0h RO	Reserved
25:24	0h RO	**Pad Ownership (PAD_OWN_GPP_A_6):** Same description as bit 0, except that the bit field applies to GPP_A6.
23:22	0h RO	Reserved
21:20	0h RO	**Pad Ownership (PAD_OWN_GPP_A_5):** Same description as bit 0, except that the bit field applies to GPP_A5.
19:18	0h RO	Reserved
17:16	0h RO	**Pad Ownership (PAD_OWN_GPP_A_4):** Same description as bit 0, except that the bit field applies to GPP_A4.
15:14	0h RO	Reserved
13:12	0h RO	**Pad Ownership (PAD_OWN_GPP_A_3):** Same description as bit 0, except that the bit field applies to GPP_A3.
11:10	0h RO	Reserved
9:8	0h RO	**Pad Ownership (PAD_OWN_GPP_A_2):** Same description as bit 0, except that the bit field applies to GPP_A2.
7:6	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
5:4	0h RO	**Pad Ownership (PAD_OWN_GPP_A_1):** Same description as bit 0, except that the bit field applies to GPP_A1.
3:2	0h RO	Reserved
1:0	0h RO	**Pad Ownership (PAD_OWN_GPP_A_0):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = Intel ME GPIO Mode. Intel ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad.

18.6.4 Pad Ownership (PAD_OWN_GPP_A_1)—Offset 24h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD		PAD_OWN_GPP_A_15		RSVD		PAD_OWN_GPP_A_14		RSVD		PAD_OWN_GPP_A_13		RSVD		PAD_OWN_GPP_A_12		RSVD		PAD_OWN_GPP_A_11		RSVD		PAD_OWN_GPP_A_10		RSVD		PAD_OWN_GPP_A_9		RSVD		PAD_OWN_GPP_A_8	

Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RO	**Pad Ownership (PAD_OWN_GPP_A_15):** Same description as bit 0, except that the bit field applies to GPP_A15.
27:26	0h RO	Reserved
25:24	0h RO	**Pad Ownership (PAD_OWN_GPP_A_14):** Same description as bit 0, except that the bit field applies to GPP_A14.
23:22	0h RO	Reserved
21:20	0h RO	**Pad Ownership (PAD_OWN_GPP_A_13):** Same description as bit 0, except that the bit field applies to GPP_A13.
19:18	0h RO	Reserved
17:16	0h RO	**Pad Ownership (PAD_OWN_GPP_A_12):** Same description as bit 0, except that the bit field applies to GPP_A12.
15:14	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
13:12	0h RO	**Pad Ownership (PAD_OWN_GPP_A_11):** Same description as bit 0, except that the bit field applies to GPP_A11.
11:10	0h RO	Reserved
9:8	0h RO	**Pad Ownership (PAD_OWN_GPP_A_10):** Same description as bit 0, except that the bit field applies to GPP_A10.
7:6	0h RO	Reserved
5:4	0h RO	**Pad Ownership (PAD_OWN_GPP_A_9):** Same description as bit 0, except that the bit field applies to GPP_A9.
3:2	0h RO	Reserved
1:0	0h RO	**Pad Ownership (PAD_OWN_GPP_A_8):** This applies to GPP_A8 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = Intel ME GPIO Mode. Intel ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad.

18.6.5 Pad Ownership (PAD_OWN_GPP_A_2)—Offset 28h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD		PAD_OWN_GPP_A_23		RSVD		PAD_OWN_GPP_A_22		RSVD		PAD_OWN_GPP_A_21		RSVD		PAD_OWN_GPP_A_20		RSVD		PAD_OWN_GPP_A_19		RSVD		PAD_OWN_GPP_A18		RSVD		PAD_OWN_GPP_A17		RSVD		PAD_OWN_GPP_A16	

Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RO	**Pad Ownership (PAD_OWN_GPP_A_23):** Same description as bit 0, except that the bit field applies to GPP_A23.
27:26	0h RO	Reserved
25:24	0h RO	**Pad Ownership (PAD_OWN_GPP_A_22):** Same description as bit 0, except that the bit field applies to GPP_A_22.

Bit Range	Default & Access	Field Name (ID): Description
23:22	0h RO	Reserved
21:20	0h RO	**Pad Ownership (PAD_OWN_GPP_A_21):** Same description as bit 0, except that the bit field applies to GPP_A21.
19:18	0h RO	Reserved
17:16	0h RO	**Pad Ownership (PAD_OWN_GPP_A_20):** Same description as bit 0, except that the bit field applies to GPP_A20.
15:14	0h RO	Reserved
13:12	0h RO	**Pad Ownership (PAD_OWN_GPP_A_19):** Same description as bit 0, except that the bit field applies to GPP_A19.
11:10	0h RO	Reserved
9:8	0h RO	**Pad Ownership (PAD_OWN_GPP_A_18):** Same description as bit 0, except that the bit field applies to GPP_A18.
7:6	0h RO	Reserved
5:4	0h RO	**Pad Ownership (PAD_OWN_GPP_A_17):** Same description as bit 0, except that the bit field applies to GPP_A17.
3:2	0h RO	Reserved
1:0	0h RO	**Pad Ownership (PAD_OWN_GPP_A_16):** This applies to GPP_A16 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = Intel ME GPIO Mode. Intel ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad.

18.6.6 Pad Ownership (PAD_OWN_GPP_B_0)—Offset 2Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD				PAD_OWN_GPP_B_7				RSVD				PAD_OWN_GPP_B_6				RSVD				PAD_OWN_GPP_B_5				RSVD				PAD_OWN_GPP_B_4			
RSVD				PAD_OWN_GPP_B_3				RSVD				PAD_OWN_GPP_B_2				RSVD				PAD_OWN_GPP_B_1				RSVD				PAD_OWN_GPP_B_0			

Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RO	**Pad Ownership (PAD_OWN_GPP_B_7):** Same description as bit 0, except that the bit field applies to GPP_B7.
27:26	0h RO	Reserved
25:24	0h RO	**Pad Ownership (PAD_OWN_GPP_B_6):** Same description as bit 0, except that the bit field applies to GPP_B6.
23:22	0h RO	Reserved
21:20	0h RO	**Pad Ownership (PAD_OWN_GPP_B_5):** Same description as bit 0, except that the bit field applies to GPP_B5.
19:18	0h RO	Reserved
17:16	0h RO	**Pad Ownership (PAD_OWN_GPP_B_4):** Same description as bit 0, except that the bit field applies to GPP_B4.
15:14	0h RO	Reserved
13:12	0h RO	**Pad Ownership (PAD_OWN_GPP_B_3):** Same description as bit 0, except that the bit field applies to GPP_B3.
11:10	0h RO	Reserved
9:8	0h RO	**Pad Ownership (PAD_OWN_GPP_B_2):** Same description as bit 0, except that the bit field applies to GPP_B2.
7:6	0h RO	Reserved
5:4	0h RO	**Pad Ownership (PAD_OWN_GPP_B_1):** Same description as bit 0, except that the bit field applies to GPP_B1.
3:2	0h RO	Reserved
1:0	0h RO	**Pad Ownership (PAD_OWN_GPP_B_0):** This applies to GPP_B0 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = Intel ME GPIO Mode. Intel ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad.

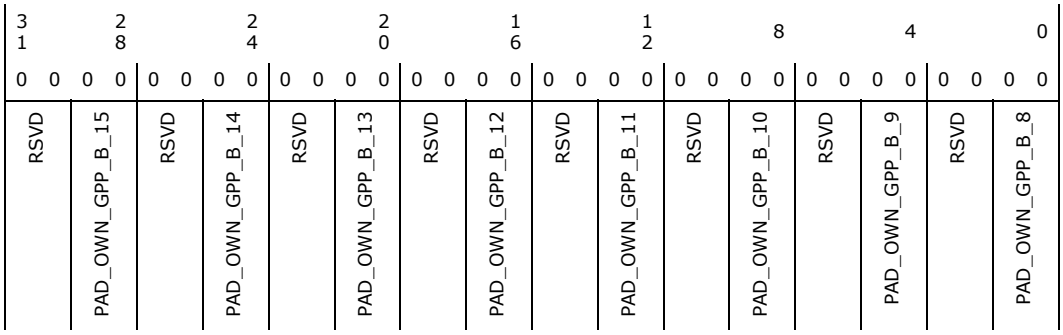
18.6.7 Pad Ownership (PAD_OWN_GPP_B_1)—Offset 30h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RO	**Pad Ownership (PAD_OWN_GPP_B_15):** Same description as bit 0, except that the bit field applies to GPP_B15.
27:26	0h RO	Reserved
25:24	0h RO	**Pad Ownership (PAD_OWN_GPP_B_14):** Same description as bit 0, except that the bit field applies to GPP_B14.
23:22	0h RO	Reserved
21:20	0h RO	**Pad Ownership (PAD_OWN_GPP_B_13):** Same description as bit 0, except that the bit field applies to GPP_B13.
19:18	0h RO	Reserved
17:16	0h RO	**Pad Ownership (PAD_OWN_GPP_B_12):** Same description as bit 0, except that the bit field applies to GPP_B12.
15:14	0h RO	Reserved
13:12	0h RO	**Pad Ownership (PAD_OWN_GPP_B_11):** Same description as bit 0, except that the bit field applies to GPP_B11.
11:10	0h RO	Reserved
9:8	0h RO	**Pad Ownership (PAD_OWN_GPP_B_10):** Same description as bit 0, except that the bit field applies to GPP_B10.
7:6	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
5:4	0h RO	**Pad Ownership (PAD_OWN_GPP_B_9):** Same description as bit 0, except that the bit field applies to GPP_B9.
3:2	0h RO	Reserved
1:0	0h RO	**Pad Ownership (PAD_OWN_GPP_B_8):** This applies to GPP_B8 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = Intel ME GPIO Mode. Intel ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad.

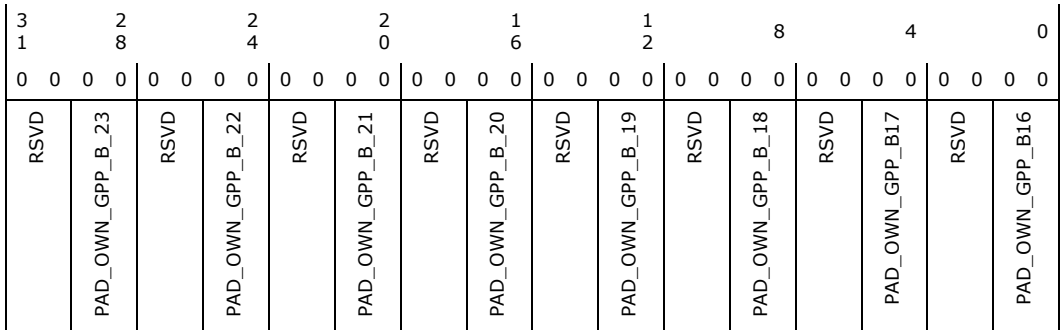
18.6.8 Pad Ownership (PAD_OWN_GPP_B_2)—Offset 34h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RO	**Pad Ownership (PAD_OWN_GPP_B_23):** Same description as bit 0, except that the bit field applies to GPP_B23.
27:26	0h RO	Reserved
25:24	0h RO	**Pad Ownership (PAD_OWN_GPP_B_22):** Same description as bit 0, except that the bit field applies to GPP_B22.
23:22	0h RO	Reserved
21:20	0h RO	**Pad Ownership (PAD_OWN_GPP_B_21):** Same description as bit 0, except that the bit field applies to GPP_B21.
19:18	0h RO	Reserved
17:16	0h RO	**Pad Ownership (PAD_OWN_GPP_B_20):** Same description as bit 0, except that the bit field applies to GPP_B20.

Bit Range	Default & Access	Field Name (ID): Description
15:14	0h RO	Reserved
13:12	0h RO	**Pad Ownership (PAD_OWN_GPP_B_19):** Same description as bit 0, except that the bit field applies to GPP_B19.
11:10	0h RO	Reserved
9:8	0h RO	**Pad Ownership (PAD_OWN_GPP_B_18):** Same description as bit 0, except that the bit field applies to GPP_B18.
7:6	0h RO	Reserved
5:4	0h RO	**Pad Ownership (PAD_OWN_GPP_B_17):** Same description as bit 0, except that the bit field applies to GPP_B17.
3:2	0h RO	Reserved
1:0	0h RO	**Pad Ownership (PAD_OWN_GPP_B_16):** This applies to GPP_B16 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = Intel ME GPIO Mode. Intel ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad.

18.6.9 Pad Ownership (PAD_OWN_GPP_F_0)—Offset 38h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD	PAD_OWN_GPP_F_7			RSVD	PAD_OWN_GPP_F_6			RSVD	PAD_OWN_GPP_F_5			RSVD	PAD_OWN_GPP_F_4			RSVD	PAD_OWN_GPP_F_3			RSVD	PAD_OWN_GPP_F_2			RSVD	PAD_OWN_GPP_F_1			RSVD	PAD_OWN_GPP_F_0		

Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RO	**Pad Ownership (PAD_OWN_GPP_F_7):** Same description as bit 0, except that the bit field applies to GPP_F7.
27:26	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
25:24	0h RO	**Pad Ownership (PAD_OWN_GPP_F_6):** Same description as bit 0, except that the bit field applies to GPP_F6.
23:22	0h RO	Reserved
21:20	0h RO	**Pad Ownership (PAD_OWN_GPP_F_5):** Same description as bit 0, except that the bit field applies to GPP_F5.
19:18	0h RO	Reserved
17:16	0h RO	**Pad Ownership (PAD_OWN_GPP_F_4):** Same description as bit 0, except that the bit field applies to GPP_F4.
15:14	0h RO	Reserved
13:12	0h RO	**Pad Ownership (PAD_OWN_GPP_F_3):** Same description as bit 0, except that the bit field applies to GPP_F3.
11:10	0h RO	Reserved
9:8	0h RO	**Pad Ownership (PAD_OWN_GPP_F_2):** Same description as bit 0, except that the bit field applies to GPP_F2.
7:6	0h RO	Reserved
5:4	0h RO	**Pad Ownership (PAD_OWN_GPP_F_1):** Same description as bit 0, except that the bit field applies to GPP_F1.
3:2	0h RO	Reserved
1:0	0h RO	**Pad Ownership (PAD_OWN_GPP_F_0):** This applies to GPP_F0 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = Intel ME GPIO Mode. Intel ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad.

18.6.10 Pad Ownership (PAD_OWN_GPP_F_1)—Offset 3Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD				RSVD				RSVD				RSVD				RSVD				RSVD				RSVD				RSVD			
PAD_OWN_GPP_F_15				PAD_OWN_GPP_F_14				PAD_OWN_GPP_F_13				PAD_OWN_GPP_F_12				PAD_OWN_GPP_F_11				PAD_OWN_GPP_F_10				PAD_OWN_GPP_F_9				PAD_OWN_GPP_F_8			

Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RO	**Pad Ownership (PAD_OWN_GPP_F_15):** Same description as bit 0, except that the bit field applies to GPP_F15.
27:26	0h RO	Reserved
25:24	0h RO	**Pad Ownership (PAD_OWN_GPP_F_14):** Same description as bit 0, except that the bit field applies to GPP_F14.
23:22	0h RO	Reserved
21:20	0h RO	**Pad Ownership (PAD_OWN_GPP_F_13):** Same description as bit 0, except that the bit field applies to GPP_F13.
19:18	0h RO	Reserved
17:16	0h RO	**Pad Ownership (PAD_OWN_GPP_F_12):** Same description as bit 0, except that the bit field applies to GPP_F12.
15:14	0h RO	Reserved
13:12	0h RO	**Pad Ownership (PAD_OWN_GPP_F_11):** Same description as bit 0, except that the bit field applies to GPP_F11.
11:10	0h RO	Reserved
9:8	0h RO	**Pad Ownership (PAD_OWN_GPP_F_10):** Same description as bit 0, except that the bit field applies to GPP_F10.
7:6	0h RO	Reserved
5:4	0h RO	**Pad Ownership (PAD_OWN_GPP_F_9):** Same description as bit 0, except that the bit field applies to GPP_F9.
3:2	0h RO	Reserved
1:0	0h RO	**Pad Ownership (PAD_OWN_GPP_F_8):** This applies to GPP_F8 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = Intel ME GPIO Mode. Intel ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad.

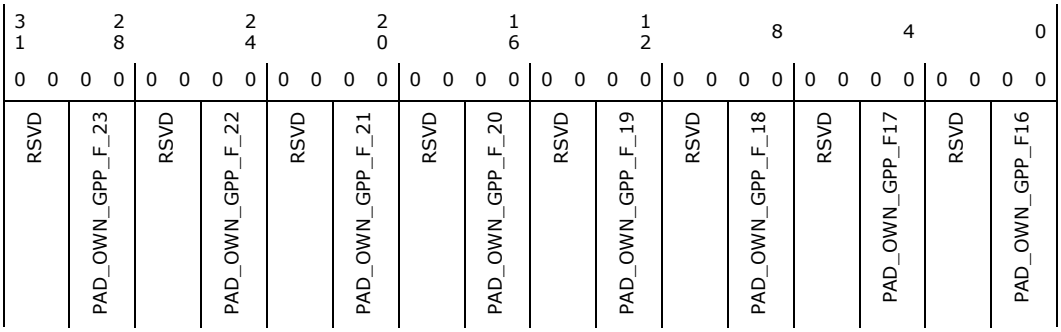
18.6.11 Pad Ownership (PAD_OWN_GPP_F_2)—Offset 40h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RO	**Pad Ownership (PAD_OWN_GPP_F_23):** Same description as bit 0, except that the bit field applies to GPP_F23.
27:26	0h RO	Reserved
25:24	0h RO	**Pad Ownership (PAD_OWN_GPP_F_22):** Same description as bit 0, except that the bit field applies to GPP_F22.
23:22	0h RO	Reserved
21:20	0h RO	**Pad Ownership (PAD_OWN_GPP_F_21):** Same description as bit 0, except that the bit field applies to GPP_F21.
19:18	0h RO	Reserved
17:16	0h RO	**Pad Ownership (PAD_OWN_GPP_F_20):** Same description as bit 0, except that the bit field applies to GPP_F20.
15:14	0h RO	Reserved
13:12	0h RO	**Pad Ownership (PAD_OWN_GPP_F_19):** Same description as bit 0, except that the bit field applies to GPP_F19.
11:10	0h RO	Reserved
9:8	0h RO	**Pad Ownership (PAD_OWN_GPP_F_18):** Same description as bit 0, except that the bit field applies to GPP_F18.
7:6	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
5:4	0h RO	**Pad Ownership (PAD_OWN_GPP_F_17):** Same description as bit 0, except that the bit field applies to GPP_F17.
3:2	0h RO	Reserved
1:0	0h RO	**Pad Ownership (PAD_OWN_GPP_F_16):** This applies to GPP_F16 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = Intel ME GPIO Mode. Intel ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad.

18.6.12 Pad Configuration Lock (PADCFGLOCK_GPP_A)—Offset 60h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD								PADCFGLOCK_GPP_A_23	PADCFGLOCK_GPP_A_22	PADCFGLOCK_GPP_A_21	PADCFGLOCK_GPP_A_20	PADCFGLOCK_GPP_A_19	PADCFGLOCK_GPP_A_18	PADCFGLOCK_GPP_A_17	PADCFGLOCK_GPP_A_16	PADCFGLOCK_GPP_A_15	PADCFGLOCK_GPP_A_14	PADCFGLOCK_GPP_A_13	PADCFGLOCK_GPP_A_12	PADCFGLOCK_GPP_A_11	PADCFGLOCK_GPP_A_10	PADCFGLOCK_GPP_A_9	PADCFGLOCK_GPP_A_8	PADCFGLOCK_GPP_A_7	PADCFGLOCK_GPP_A_6	PADCFGLOCK_GPP_A_5	PADCFGLOCK_GPP_A_4	PADCFGLOCK_GPP_A_3	PADCFGLOCK_GPP_A_2	PADCFGLOCK_GPP_A_1	PADCFGLOCK_GPP_A_0

Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_A_23):** Applied to GPP_A23. Same description as PADCFGLOCK_GPP_A_0.
22	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_A_22):** Applied to GPP_A22. Same description as PADCFGLOCK_GPP_A_0.
21	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_A_21):** Applied to GPP_A21. Same description as PADCFGLOCK_GPP_A_0.
20	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_A_20):** Applied to GPP_A20. Same description as PADCFGLOCK_GPP_A_0.
19	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_A_19):** Applied to GPP_A19. Same description as PADCFGLOCK_GPP_A_0.
18	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_A_18):** Applied to GPP_A18. Same description as PADCFGLOCK_GPP_A_0.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_A_17):** Applied to GPP_A17. Same description as PADCFGLOCK_GPP_A_0.
16	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_A_16):** Applied to GPP_A16. Same description as PADCFGLOCK_GPP_A_0.
15	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_A_15):** Applied to GPP_A15. Same description as PADCFGLOCK_GPP_A_0.
14	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_A_14):** Applied to GPP_A14. Same description as PADCFGLOCK_GPP_A_0.
13	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_A_13):** Applied to GPP_A13. Same description as PADCFGLOCK_GPP_A_0.
12	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_A_12):** Applied to GPP_A12. Same description as PADCFGLOCK_GPP_A_0.
11	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_A_11):** Applied to GPP_A11. Same description as PADCFGLOCK_GPP_A_0.
10	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_A_10):** Applied to GPP_A10. Same description as PADCFGLOCK_GPP_A_0.
9	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_A_9):** Applied to GPP_A9. Same description as PADCFGLOCK_GPP_A_0.
8	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_A_8):** Applied to GPP_A8. Same description as PADCFGLOCK_GPP_A_0.
7	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_A_7):** Applied to GPP_A7. Same description as PADCFGLOCK_GPP_A_0.
6	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_A_6):** Applied to GPP_A6. Same description as PADCFGLOCK_GPP_A_0.
5	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_A_5):** Applied to GPP_A5. Same description as PADCFGLOCK_GPP_A_0.
4	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_A_4):** Applied to GPP_A4. Same description as PADCFGLOCK_GPP_A_0.
3	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_A_3):** Applied to GPP_A3. Same description as PADCFGLOCK_GPP_A_0.
2	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_A_2):** Applied to GPP_A2. Same description as PADCFGLOCK_GPP_A_0.
1	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_A_1):** Applied to GPP_A1. Same description as PADCFGLOCK_GPP_A_0.
0	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_A_0):** Pad Configuration Lock locks specific register fields in the GPP specific registers from being configured. The registers affected become Read-Only and software writes to these registers have no effect. 0 = Unlock 1 = Lock the following register fields as read-only (RO): - Pad Configuration registers (exclude GPIOTXState) - GPI_NMI_EN Register (if implemented) - GPI_SMI_EN Register (if implemented) - GPI_GPE_EN Register (if implemented) When PadCfgLock is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

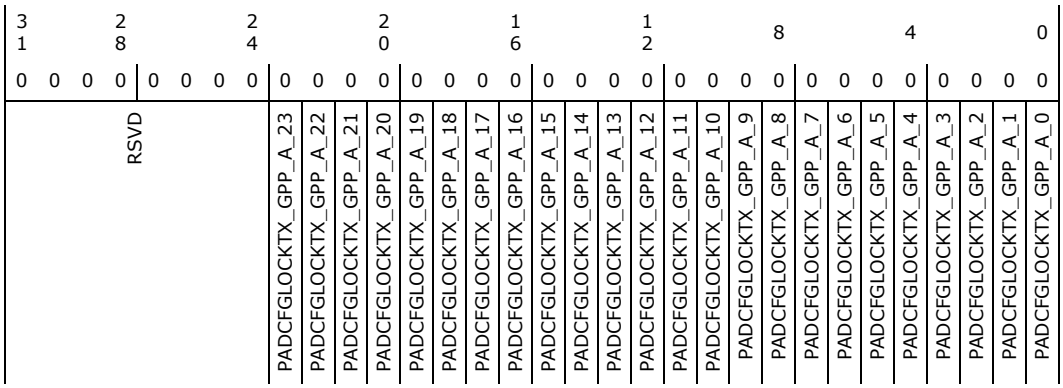
18.6.13 Pad Configuration Lock (PADCFGLOCKTX_GPP_A)—Offset 64h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_A_23):** Applied to GPP_A23. Same description as PADCFGLOCKTX_GPP_A_0.
22	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_A_22):** Applied to GPP_A22. Same description as PADCFGLOCKTX_GPP_A_0.
21	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_A_21):** Applied to GPP_A21. Same description as PADCFGLOCKTX_GPP_A_0.
20	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_A_20):** Applied to GPP_A20. Same description as PADCFGLOCKTX_GPP_A_0.
19	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_A_19):** Applied to GPP_A19. Same description as PADCFGLOCKTX_GPP_A_0.
18	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_A_18):** Applied to GPP_A18. Same description as PADCFGLOCKTX_GPP_A_0.
17	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_A_17):** Applied to GPP_A17. Same description as PADCFGLOCKTX_GPP_A_0.
16	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_A_16):** Applied to GPP_A16. Same description as PADCFGLOCKTX_GPP_A_0.
15	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_A_15):** Applied to GPP_A15. Same description as PADCFGLOCKTX_GPP_A_0.
14	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_A_14):** Applied to GPP_A14. Same description as PADCFGLOCKTX_GPP_A_0.
13	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_A_13):** Applied to GPP_A13. Same description as PADCFGLOCKTX_GPP_A_0.
12	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_A_12):** Applied to GPP_A12. Same description as PADCFGLOCKTX_GPP_A_0.
11	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_A_11):** Applied to GPP_A11. Same description as PADCFGLOCKTX_GPP_A_0.

Bit Range	Default & Access	Field Name (ID): Description
10	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_A_10):** Applied to GPP_A10. Same description as PADCFGLOCKTX_GPP_A_0.
9	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_A_9):** Applied to GPP_A9. Same description as PADCFGLOCKTX_GPP_A_0.
8	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_A_8):** Applied to GPP_A8. Same description as PADCFGLOCKTX_GPP_A_0.
7	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_A_7):** Applied to GPP_A7. Same description as PADCFGLOCKTX_GPP_A_0.
6	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_A_6):** Applied to GPP_A6. Same description as PADCFGLOCKTX_GPP_A_0.
5	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_A_5):** Applied to GPP_A5. Same description as PADCFGLOCKTX_GPP_A_0.
4	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_A_4):** Applied to GPP_A4. Same description as PADCFGLOCKTX_GPP_A_0.
3	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_A_3):** Applied to GPP_A3. Same description as PADCFGLOCKTX_GPP_A_0.
2	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_A_2):** Applied to GPP_A2. Same description as PADCFGLOCKTX_GPP_A_0.
1	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_A_1):** Applied to GPP_A1. Same description as PADCFGLOCKTX_GPP_A_0.
0	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_A_0):** PadCfgLockTx locks the GPIOTxState bit from being configured. The GPIOTxState register becomes Read-Only and software writes to the register have no effect. 0 = Unlock 1 = Locks the Pad Configuration GPIOTXState field as read-only (RO) When PadCfgLockTx is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

18.6.14 Pad Configuration Lock (PADCFGLOCK_GPP_B)—Offset 68h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD								PADCFGLOCK_GPP_B_23	PADCFGLOCK_GPP_B_22	PADCFGLOCK_GPP_B_21	PADCFGLOCK_GPP_B_20	PADCFGLOCK_GPP_B_19	PADCFGLOCK_GPP_B_18	PADCFGLOCK_GPP_B_17	PADCFGLOCK_GPP_B_16	PADCFGLOCK_GPP_B_15	PADCFGLOCK_GPP_B_14	PADCFGLOCK_GPP_B_13	PADCFGLOCK_GPP_B_12	PADCFGLOCK_GPP_B_11	PADCFGLOCK_GPP_B_10	PADCFGLOCK_GPP_B_9	PADCFGLOCK_GPP_B_8	PADCFGLOCK_GPP_B_7	PADCFGLOCK_GPP_B_6	PADCFGLOCK_GPP_B_5	PADCFGLOCK_GPP_B_4	PADCFGLOCK_GPP_B_3	PADCFGLOCK_GPP_B_2	PADCFGLOCK_GPP_B_1	PADCFGLOCK_GPP_B_0

Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_B_23):** Applied to GPP_B23. Same description as PADCFGLOCK_GPP_B_0.
22	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_B_22):** Applied to GPP_B22. Same description as PADCFGLOCK_GPP_BGPP_B_0.
21	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_B_21):** Applied to GPP_B21. Same description as PADCFGLOCK_GPP_B_0.
20	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_B_20):** Applied to GPP_B20. Same description as PADCFGLOCK_GPP_B_0.
19	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_B_19):** Applied to GPP_B19. Same description as PADCFGLOCK_GPP_B_0.
18	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_B_18):** Applied to GPP_B18. Same description as PADCFGLOCK_GPP_B_0.
17	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_B_17):** Applied to GPP_B17. Same description as PADCFGLOCK_GPP_B_0.
16	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_B_16):** Applied to GPP_B16. Same description as PADCFGLOCK_GPP_B_0.
15	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_B_15):** Applied to GPP_B15. Same description as PADCFGLOCK_GPP_B_0.
14	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_B_14):** Applied to GPP_B14. Same description as PADCFGLOCK_GPP_B_0.
13	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_B_13):** Applied to GPP_B13. Same description as PADCFGLOCK_GPP_B_0.
12	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_B_12):** Applied to GPP_B12. Same description as PADCFGLOCK_GPP_B_0.
11	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_B_11):** Applied to GPP_B11. Same description as PADCFGLOCK_GPP_B_0.
10	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_B_10):** Applied to GPP_B10. Same description as PADCFGLOCK_GPP_B_0.
9	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_B_9):** Applied to GPP_B9. Same description as PADCFGLOCK_GPP_B_0.
8	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_B_8):** Applied to GPP_B8. Same description as PADCFGLOCK_GPP_B_0.
7	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_B_7):** Applied to GPP_B7. Same description as PADCFGLOCK_GPP_B_0.
6	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_B_6):** Applied to GPP_B6. Same description as PADCFGLOCK_GPP_B_0.
5	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_B_5):** Applied to GPP_B5. Same description as PADCFGLOCK_GPP_B_0.
4	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_B_4):** Applied to GPP_B4. Same description as PADCFGLOCK_GPP_B_0.
3	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_B_3):** Applied to GPP_B3. Same description as PADCFGLOCK_GPP_B_0.
2	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_B_2):** Applied to GPP_B2. Same description as PADCFGLOCK_GPP_B_0.
1	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_B_1):** Applied to GPP_B1. Same description as PADCFGLOCK_GPP_B_0.

Bit Range	Default & Access	Field Name (ID): Description
0	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_B_0):** Pad Configuration Lock locks specific register fields in the GPP specific registers from being configured. The registers affected become Read-Only and software writes to these registers have no effect. 0 = Unlock 1 = Lock the following register fields as read-only (RO): - Pad Configuration registers (exclude GPIOTXState) - GPI_NMI_EN Register (if implemented) - GPI_SMI_EN Register (if implemented) - GPI_GPE_EN Register (if implemented) When PadCfgLock is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

18.6.15 Pad Configuration Lock (PADCFGLOCKTX_GPP_B)—Offset 6Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD								PADCFGLOCKTX_GPP_BGPP_B_23	PADCFGLOCKTX_GPP_B_22	PADCFGLOCKTX_GPP_B_21	PADCFGLOCKTX_GPP_B_20	PADCFGLOCKTX_GPP_B_19	PADCFGLOCKTX_GPP_B_18	PADCFGLOCKTX_GPP_B_17	PADCFGLOCKTX_GPP_B_16	PADCFGLOCKTX_GPP_B_15	PADCFGLOCKTX_GPP_B_14	PADCFGLOCKTX_GPP_B_13	PADCFGLOCKTX_GPP_B_12	PADCFGLOCKTX_GPP_B_11	PADCFGLOCKTX_GPP_B_10	PADCFGLOCKTX_GPP_B_9	PADCFGLOCKTX_GPP_B_8	PADCFGLOCKTX_GPP_B_7	PADCFGLOCKTX_GPP_B_6	PADCFGLOCKTX_GPP_B_5	PADCFGLOCKTX_GPP_B_4	PADCFGLOCKTX_GPP_B_3	PADCFGLOCKTX_GPP_B_2	PADCFGLOCKTX_GPP_B_1	PADCFGLOCKTX_GPP_B_0

Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_B_23):** Applied to GPP_B23. Same description as PADCFGLOCKTX_GPP_B_0.
22	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_B_22):** Applied to GPP_B22. Same description as PADCFGLOCKTX_GPP_B_0.
21	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_B_21):** Applied to GPP_B21. Same description as PADCFGLOCKTX_GPP_B_0.
20	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_B_20):** Applied to GPP_B20. Same description as PADCFGLOCKTX_GPP_B_0.
19	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_B_19):** Applied to GPP_B19. Same description as PADCFGLOCKTX_GPP_B_0.
18	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_B_18):** Applied to GPP_B18. Same description as PADCFGLOCKTX_GPP_B_0.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_B_17):** Applied to GPP_B17. Same description as PADCFGLOCKTX_GPP_B_0.
16	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_B_16):** Applied to GPP_B16. Same description as PADCFGLOCKTX_GPP_B_0.
15	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_B_15):** Applied to GPP_B15. Same description as PADCFGLOCKTX_GPP_B_0.
14	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_B_14):** Applied to GPP_B14. Same description as PADCFGLOCKTX_GPP_B_0.
13	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_B_13):** Applied to GPP_B13. Same description as PADCFGLOCKTX_GPP_B_0.
12	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_B_12):** Applied to GPP_B12. Same description as PADCFGLOCKTX_GPP_B_0.
11	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_B_11):** Applied to GPP_B11. Same description as PADCFGLOCKTX_GPP_B_0.
10	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_B_10):** Applied to GPP_B10. Same description as PADCFGLOCKTX_GPP_B_0.
9	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_B_9):** Applied to GPP_B9. Same description as PADCFGLOCKTX_GPP_B_0.
8	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_B_8):** Applied to GPP_B8. Same description as PADCFGLOCKTX_GPP_B_0.
7	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_B_7):** Applied to GPP_B7. Same description as PADCFGLOCKTX_GPP_B_0.
6	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_B_6):** Applied to GPP_B6. Same description as PADCFGLOCKTX_GPP_B_0.
5	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_B_5):** Applied to GPP_B5. Same description as PADCFGLOCKTX_GPP_B_0.
4	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_B_4):** Applied to GPP_B4. Same description as PADCFGLOCKTX_GPP_B_0.
3	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_B_3):** Applied to GPP_B3. Same description as PADCFGLOCKTX_GPP_B_0.
2	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_B_2):** Applied to GPP_B2. Same description as PADCFGLOCKTX_GPP_B_0.
1	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_B_1):** Applied to GPP_B1. Same description as PADCFGLOCKTX_GPP_B_0.
0	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_B_0):** PadCfgLockTx locks the GPIOTxState bit from being configured. The GPIOTxState register becomes Read-Only and software writes to the register have no effect. 0 = Unlock 1 = Locks the Pad Configuration GPIOTXState field as read-only (RO) When PadCfgLockTx is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

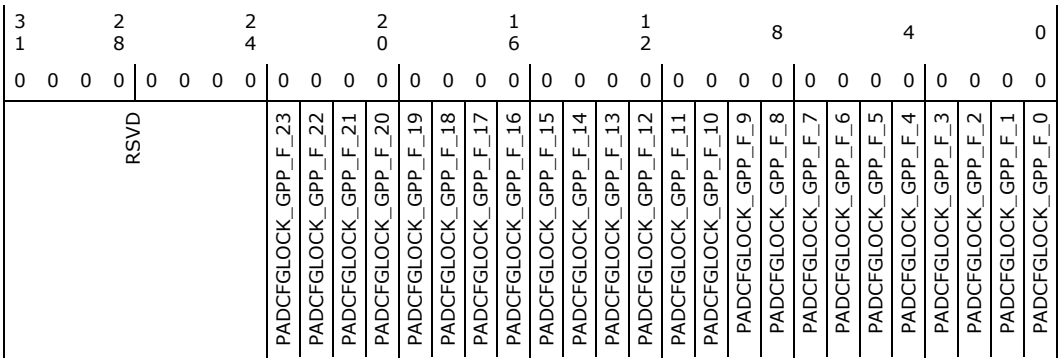
18.6.16 Pad Configuration Lock (PADCFGLOCK_GPP_F)—Offset 70h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_F_23):** Applied to GPP_F23. Same description as PADCFGLOCK_GPP_F_0.
22	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_F_22):** Applied to GPP_F22. Same description as PADCFGLOCK_GPP_FGPP_F_0.
21	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_F_21):** Applied to GPP_F21. Same description as PADCFGLOCK_GPP_F_0.
20	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_F_20):** Applied to GPP_F20. Same description as PADCFGLOCK_GPP_F_0.
19	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_F_19):** Applied to GPP_F19. Same description as PADCFGLOCK_GPP_F_0.
18	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_F_18):** Applied to GPP_F18. Same description as PADCFGLOCK_GPP_F_0.
17	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_F_17):** Applied to GPP_F17. Same description as PADCFGLOCK_GPP_F_0.
16	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_F_16):** Applied to GPP_F16. Same description as PADCFGLOCK_GPP_F_0.
15	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_F_15):** Applied to GPP_F15. Same description as PADCFGLOCK_GPP_FGPP_F_0.
14	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_F_14):** Applied to GPP_F14. Same description as PADCFGLOCK_GPP_F_0.
13	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_F_13):** Applied to GPP_F13. Same description as PADCFGLOCK_GPP_F_0.
12	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_F_12):** Applied to GPP_F12. Same description as PADCFGLOCK_GPP_F_0.
11	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_F_11):** Applied to GPP_F11. Same description as PADCFGLOCK_GPP_F_0.

Bit Range	Default & Access	Field Name (ID): Description
10	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_F_10):** Applied to GPP_F10. Same description as PADCFGLOCK_GPP_F_0.
9	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_F_9):** Applied to GPP_F9. Same description as PADCFGLOCK_GPP_F_0.
8	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_F_8):** Applied to GPP_F8. Same description as PADCFGLOCK_GPP_F_0.
7	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_F_7):** Applied to GPP_F7. Same description as PADCFGLOCK_GPP_F_0.
6	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_F_6):** Applied to GPP_F6. Same description as PADCFGLOCK_GPP_F_0.
5	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_F_5):** Applied to GPP_F5. Same description as PADCFGLOCK_GPP_F_0.
4	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_F_4):** Applied to GPP_F4. Same description as PADCFGLOCK_GPP_F_0.
3	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_F_3):** Applied to GPP_F3. Same description as PADCFGLOCK_GPP_F_0.
2	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_F_2):** Applied to GPP_F2. Same description as PADCFGLOCK_GPP_F_0.
1	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_F_1):** Applied to GPP_F1. Same description as PADCFGLOCK_GPP_F_0.
0	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_F_0):** Pad Configuration Lock locks specific register fields in the GPP specific registers from being configured. The registers affected become Read-Only and software writes to these registers have no effect. 0 = Unlock 1 = Lock the following register fields as read-only (RO): - Pad Configuration registers (exclude GPIOTXState) - GPI_NMI_EN Register (if implemented) - GPI_SMI_EN Register (if implemented) - GPI_GPE_EN Register (if implemented) When PadCfgLock is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

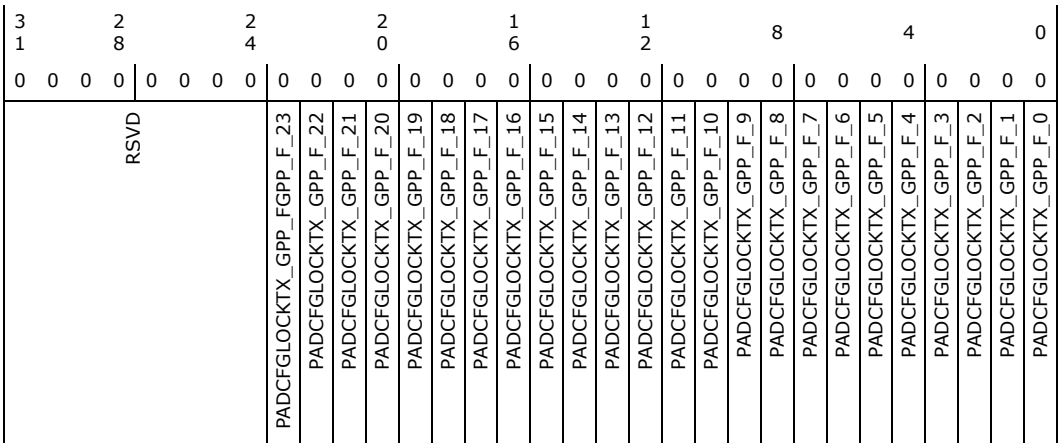
18.6.17 Pad Configuration Lock (PADCFGLOCKTX_GPP_F)—Offset 74h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_F_23):** Applied to GPP_F23. Same description as PADCFGLOCKTX_GPP_F_0.
22	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_F_22):** Applied to GPP_F22. Same description as PADCFGLOCKTX_GPP_F_0.
21	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_F_21):** Applied to GPP_F21. Same description as PADCFGLOCKTX_GPP_F_0.
20	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_F_20):** Applied to GPP_F20. Same description as PADCFGLOCKTX_GPP_F_0.
19	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_F_19):** Applied to GPP_F19. Same description as PADCFGLOCKTX_GPP_F_0.
18	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_F_18):** Applied to GPP_F18. Same description as PADCFGLOCKTX_GPP_F_0.
17	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_F_17):** Applied to GPP_F17. Same description as PADCFGLOCKTX_GPP_F_0.
16	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_F_16):** Applied to GPP_F16. Same description as PADCFGLOCKTX_GPP_F_0.
15	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_F_15):** Applied to GPP_F15. Same description as PADCFGLOCKTX_GPP_F_0.
14	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_F_14):** Applied to GPP_F14. Same description as PADCFGLOCKTX_GPP_F_0.
13	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_F_13):** Applied to GPP_F13. Same description as PADCFGLOCKTX_GPP_F_0.
12	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_F_12):** Applied to GPP_F12. Same description as PADCFGLOCKTX_GPP_F_0.

Bit Range	Default & Access	Field Name (ID): Description
11	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_F_11):** Applied to GPP_F11. Same description as PADCFGLOCKTX_GPP_F_0.
10	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_F_10):** Applied to GPP_F10. Same description as PADCFGLOCKTX_GPP_F_0.
9	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_F_9):** Applied to GPP_F9. Same description as PADCFGLOCKTX_GPP_F_0.
8	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_F_8):** Applied to GPP_F8. Same description as PADCFGLOCKTX_GPP_F_0.
7	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_F_7):** Applied to GPP_F7. Same description as PADCFGLOCKTX_GPP_F_0.
6	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_F_6):** Applied to GPP_F6. Same description as PADCFGLOCKTX_GPP_F_0.
5	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_F_5):** Applied to GPP_F5. Same description as PADCFGLOCKTX_GPP_F_0.
4	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_F_4):** Applied to GPP_F4. Same description as PADCFGLOCKTX_GPP_F_0.
3	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_F_3):** Applied to GPP_F3. Same description as PADCFGLOCKTX_GPP_F_0.
2	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_F_2):** Applied to GPP_F2. Same description as PADCFGLOCKTX_GPP_F_0.
1	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_F_1):** Applied to GPP_F1. Same description as PADCFGLOCKTX_GPP_F_0.
0	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_F_0):** PadCfgLockTx locks the GPIOTxState bit from being configured. The GPIOTxState register becomes Read-Only and software writes to the register have no effect. 0 = Unlock 1 = Locks the Pad Configuration GPIOTXState field as read-only (RO) When PadCfgLockTx is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

18.6.18 Host Software Pad Ownership (HOSTSW_OWN_GPP_A)—Offset 80h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	28	24	20	16	12	8	4	0
0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0
RSVD		HOSTSW_OWN_GPP_A_23	HOSTSW_OWN_GPP_A_22	HOSTSW_OWN_GPP_A_21	HOSTSW_OWN_GPP_A_20	HOSTSW_OWN_GPP_A_19	HOSTSW_OWN_GPP_A_18	HOSTSW_OWN_GPP_A_17	HOSTSW_OWN_GPP_A_16	HOSTSW_OWN_GPP_A_15	HOSTSW_OWN_GPP_A_14	HOSTSW_OWN_GPP_A_13	HOSTSW_OWN_GPP_A_12	HOSTSW_OWN_GPP_A_11	HOSTSW_OWN_GPP_A_10	HOSTSW_OWN_GPP_A_9	HOSTSW_OWN_GPP_A_8	HOSTSW_OWN_GPP_A_7	HOSTSW_OWN_GPP_A_6	HOSTSW_OWN_GPP_A_5	HOSTSW_OWN_GPP_A_4	HOSTSW_OWN_GPP_A_3	HOSTSW_OWN_GPP_A_2	HOSTSW_OWN_GPP_A_1	HOSTSW_OWN_GPP_A_0

Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_A_23):** Applied to GPP_A23. Same description as bit 0.
22	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_A_22):** Applied to GPP_A22. Same description as bit 0.
21	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_A_21):** Applied to GPP_A21. Same description as bit 0.
20	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_A_20):** Applied to GPP_A20. Same description as bit 0.
19	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_A_19):** Applied to GPP_A19. Same description as bit 0.
18	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_A_18):** Applied to GPP_A18. Same description as bit 0.
17	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_A_17):** Applied to GPP_A17. Same description as bit 0.
16	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_A_16):** Applied to GPP_A16. Same description as bit 0.
15	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_A_15):** Applied to GPP_A15. Same description as bit 0.
14	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_A_14):** Applied to GPP_A14. Same description as bit 0.
13	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_A_13):** Applied to GPP_A13. Same description as bit 0.
12	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_A_12):** Applied to GPP_A12. Same description as bit 0.
11	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_A_11):** Applied to GPP_A11. Same description as bit 0.
10	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_A_10):** Applied to GPP_A10. Same description as bit 0.
9	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_A_9):** Applied to GPP_A9. Same description as bit 0.
8	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_A_8):** Applied to GPP_A8. Same description as bit 0.
7	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_A_7):** Applied to GPP_A7. Same description as bit 0.
6	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_A_6):** Applied to GPP_A6. Same description as bit 0.
5	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_A_5):** Applied to GPP_A5. Same description as bit 0.
4	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_A_4):** Applied to GPP_A4. Same description as bit 0.
3	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_A_3):** Applied to GPP_A3. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
2	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_A_2):** Applied to GPP_A2. Same description as bit 0.
1	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_A_1):** Applied to GPP_A1. Same description as bit 0.
0	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_A_0):** This register determines the appropriate host status bit update when a pad is under host ownership. 0 = ACPI Mode. GPIO input event updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. GPI_STS update is masked. 1 = GPIO Driver Mode. GPIO input event updates are limited to GPI_STS. GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS updates are masked.

18.6.19 Host Software Pad Ownership (HOSTSW_OWN_GPP_B)—Offset 84h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD								HOSTSW_OWN_GPP_B_23	HOSTSW_OWN_GPP_B_22	HOSTSW_OWN_GPP_B_21	HOSTSW_OWN_GPP_B_20	HOSTSW_OWN_GPP_B_19	HOSTSW_OWN_GPP_B_18	HOSTSW_OWN_GPP_B_17	HOSTSW_OWN_GPP_B_16	HOSTSW_OWN_GPP_B_15	HOSTSW_OWN_GPP_B_14	HOSTSW_OWN_GPP_B_13	HOSTSW_OWN_GPP_B_12	HOSTSW_OWN_GPP_B_11	HOSTSW_OWN_GPP_B_10	HOSTSW_OWN_GPP_B_9	HOSTSW_OWN_GPP_B_8	HOSTSW_OWN_GPP_B_7	HOSTSW_OWN_GPP_B_6	HOSTSW_OWN_GPP_B_5	HOSTSW_OWN_GPP_B_4	HOSTSW_OWN_GPP_B_3	HOSTSW_OWN_GPP_B_2	HOSTSW_OWN_GPP_B_1	HOSTSW_OWN_GPP_B_0

Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_B_23):** Applied to GPP_B23. Same description as bit 0.
22	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_B_22):** Applied to GPP_B22. Same description as bit 0.
21	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_B_21):** Applied to GPP_B21. Same description as bit 0.
20	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_B_20):** Applied to GPP_B20. Same description as bit 0.
19	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_B_19):** Applied to GPP_B19. Same description as bit 0.
18	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_B_18):** Applied to GPP_B18. Same description as bit 0.
17	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_B_17):** Applied to GPP_B17. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
16	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_B_16):** Applied to GPP_B16. Same description as bit 0.
15	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_B_15):** Applied to GPP_B15. Same description as bit 0.
14	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_B_14):** Applied to GPP_B14. Same description as bit 0.
13	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_B_13):** Applied to GPP_B13. Same description as bit 0.
12	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_B_12):** Applied to GPP_B12. Same description as bit 0.
11	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_B_11):** Applied to GPP_B11. Same description as bit 0.
10	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_B_10):** Applied to GPP_B10. Same description as bit 0.
9	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_B_9):** Applied to GPP_B9. Same description as bit 0.
8	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_B_8):** Applied to GPP_B8. Same description as bit 0.
7	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_B_7):** Applied to GPP_B7. Same description as bit 0.
6	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_B_6):** Applied to GPP_B6. Same description as bit 0.
5	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_B_5):** Applied to GPP_B5. Same description as bit 0.
4	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_B_4):** Applied to GPP_B4. Same description as bit 0.
3	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_B_3):** Applied to GPP_B3. Same description as bit 0.
2	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_B_2):** Applied to GPP_B2. Same description as bit 0.
1	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_B_1):** Applied to GPP_B1. Same description as bit 0.
0	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_B_0):** This register determines the appropriate host status bit update when a pad is under host ownership. 0 = ACPI Mode. GPIO input event updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. GPI_STS update is masked. 1 = GPIO Driver Mode. GPIO input event updates are limited to GPI_STS. GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS updates are masked.

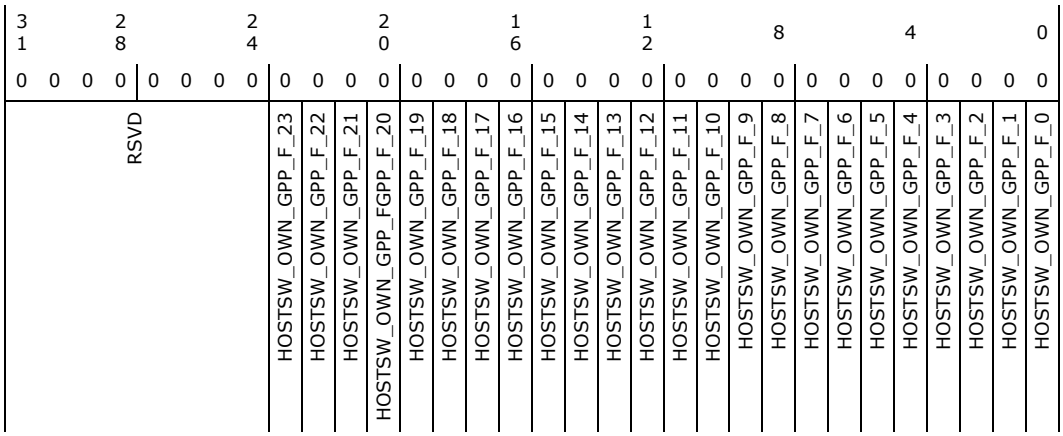
18.6.20 Host Software Pad Ownership (HOSTSW_OWN_GPP_F)—Offset 88h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_F_23):** Applied to GPP_F23. Same description as bit 0.
22	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_F_22):** Applied to GPP_F22. Same description as bit 0.
21	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_F_21):** Applied to GPP_F21. Same description as bit 0.
20	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_F_20):** Applied to GPP_F20. Same description as bit 0.
19	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_F_19):** Applied to GPP_F19. Same description as bit 0.
18	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_F_18):** Applied to GPP_F18. Same description as bit 0.
17	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_F_17):** Applied to GPP_F17. Same description as bit 0.
16	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_F_16):** Applied to GPP_F16. Same description as bit 0.
15	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_F_15):** Applied to GPP_F15. Same description as bit 0.
14	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_F_14):** Applied to GPP_F14. Same description as bit 0.
13	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_F_13):** Applied to GPP_F13. Same description as bit 0.
12	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_F_12):** Applied to GPP_F12. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
11	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_F_11):** Applied to GPP_F11. Same description as bit 0.
10	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_F_10):** Applied to GPP_F10. Same description as bit 0.
9	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_F_9):** Applied to GPP_F9. Same description as bit 0.
8	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_F_8):** Applied to GPP_F8. Same description as bit 0.
7	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_F_7):** Applied to GPP_F7. Same description as bit 0.
6	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_F_6):** Applied to GPP_F6. Same description as bit 0.
5	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_F_5):** Applied to GPP_F5. Same description as bit 0.
4	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_F_4):** Applied to GPP_F4. Same description as bit 0.
3	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_F_3):** Applied to GPP_F3. Same description as bit 0.
2	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_F_2):** Applied to GPP_F2. Same description as bit 0.
1	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_F_1):** Applied to GPP_F1. Same description as bit 0.
0	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_F_0):** This register determines the appropriate host status bit update when a pad is under host ownership. 0 = ACPI Mode. GPIO input event updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. GPI_STS update is masked. 1 = GPIO Driver Mode. GPIO input event updates are limited to GPI_STS. GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS updates are masked.

18.6.21 GPI Interrupt Status (GPI_IS_GPP_A)—Offset 100h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD								GPI_INT_STS_GPP_A_23	GPI_INT_STS_GPP_A_22	GPI_INT_STS_GPP_A_21	GPI_INT_STS_GPP_A_20	GPI_INT_STS_GPP_A_19	GPI_INT_STS_GPP_A_18	GPI_INT_STS_GPP_A_17	GPI_INT_STS_GPP_A_16	GPI_INT_STS_GPP_A_15	GPI_INT_STS_GPP_A_14	GPI_INT_STS_GPP_A_13	GPI_INT_STS_GPP_A_12	GPI_INT_STS_GPP_A_11	GPI_INT_STS_GPP_A_10	GPI_INT_STS_GPP_A_9	GPI_INT_STS_GPP_A_8	GPI_INT_STS_GPP_A_7	GPI_INT_STS_GPP_A_6	GPI_INT_STS_GPP_A_5	GPI_INT_STS_GPP_A_4	GPI_INT_STS_GPP_A_3	GPI_INT_STS_GPP_A_2	GPI_INT_STS_GPP_A_1	GPI_INT_STS_GPP_A_0

Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_A_23):** Applied to GPP_A23. Same description as bit 0.
22	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_A_22):** Applied to GPP_A22. Same description as bit 0.
21	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_A_21):** Applied to GPP_A21. Same description as bit 0.
20	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_A_20):** Applied to GPP_A20. Same description as bit 0.
19	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_A_19):** Applied to GPP_A19. Same description as bit 0.
18	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_A_18):** Applied to GPP_A18. Same description as bit 0.
17	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_A_17):** Applied to GPP_A17. Same description as bit 0.
16	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_A_16):** Applied to GPP_A16. Same description as bit 0.
15	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_A_15):** Applied to GPP_A15. Same description as bit 0.
14	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_A_14):** Applied to GPP_A14. Same description as bit 0.
13	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_A_13):** Applied to GPP_A13. Same description as bit 0.
12	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_A_12):** Applied to GPP_A12. Same description as bit 0.
11	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_A_11):** Applied to GPP_A11. Same description as bit 0.
10	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_A_10):** Applied to GPP_A10. Same description as bit 0.
9	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_A_9):** Applied to GPP_A9. Same description as bit 0.
8	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_A_8):** Applied to GPP_A8. Same description as bit 0.
7	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_A_7):** Applied to GPP_A7. Same description as bit 0.
6	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_A_6):** Applied to GPP_A6. Same description as bit 0.
5	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_A_5):** Applied to GPP_A5. Same description as bit 0.
4	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_A_4):** Applied to GPP_A4. Same description as bit 0.
3	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_A_3):** Applied to GPP_A3. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
2	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_A_2):** Applied to GPP_A2. Same description as bit 0.
1	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_A_1):** Applied to GPP_A1. Same description as bit 0.
0	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_A_0):** GPI Interrupt Status (GPI_INT_STS) This bit is set to '1' by hardware when either an edge or a level event is detected (See RxEdCfg RxInv) on pad and all the following conditions are true: - The corresponding pad is used in GPIO input mode - HOSTSW_OWN = 1 (i.e. Host GPIO Driver Mode). Writing a value of 1 will clear the bit while writing a value of 0 has no effect. 0 = No interrupt 1 = Interrupt asserts The state of GPI_INT_EN[x] does not prevent the setting of GPI_INT_STS[x].

18.6.22 GPI Interrupt Status (GPI_IS_GPP_B)—Offset 104h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD								GPI_INT_STS_GPP_B_23	GPI_INT_STS_GPP_B_22	GPI_INT_STS_GPP_B_21	GPI_INT_STS_GPP_B_20	GPI_INT_STS_GPP_B_19	GPI_INT_STS_GPP_B_18	GPI_INT_STS_GPP_B_17	GPI_INT_STS_GPP_B_16	GPI_INT_STS_GPP_B_15	GPI_INT_STS_GPP_B_14	GPI_INT_STS_GPP_B_13	GPI_INT_STS_GPP_B_12	GPI_INT_STS_GPP_B_11	GPI_INT_STS_GPP_B_10	GPI_INT_STS_GPP_B_9	GPI_INT_STS_GPP_B_8	GPI_INT_STS_GPP_B_7	GPI_INT_STS_GPP_B_6	GPI_INT_STS_GPP_B_5	GPI_INT_STS_GPP_B_4	GPI_INT_STS_GPP_B_3	GPI_INT_STS_GPP_B_2	GPI_INT_STS_GPP_B_1	GPI_INT_STS_GPP_B_0

Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_B_23):** Applied to GPP_B23. Same description as bit 0.
22	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_B_22):** Applied to GPP_B22. Same description as bit 0.
21	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_B_21):** Applied to GPP_B21. Same description as bit 0.
20	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_B_20):** Applied to GPP_B20. Same description as bit 0.
19	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_B_19):** Applied to GPP_B19. Same description as bit 0.
18	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_B_18):** Applied to GPP_B18. Same description as bit 0.
17	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_B_17):** Applied to GPP_B17. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
16	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_B_16):** Applied to GPP_B16. Same description as bit 0.
15	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_B_15):** Applied to GPP_B15. Same description as bit 0.
14	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_B_14):** Applied to GPP_B14. Same description as bit 0.
13	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_B_13):** Applied to GPP_B13. Same description as bit 0.
12	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_B_12):** Applied to GPP_B12. Same description as bit 0.
11	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_B_11):** Applied to GPP_B11. Same description as bit 0.
10	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_B_10):** Applied to GPP_B10. Same description as bit 0.
9	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_B_9):** Applied to GPP_B9. Same description as bit 0.
8	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_B_8):** Applied to GPP_B8. Same description as bit 0.
7	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_B_7):** Applied to GPP_B7. Same description as bit 0.
6	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_B_6):** Applied to GPP_B6. Same description as bit 0.
5	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_B_5):** Applied to GPP_B5. Same description as bit 0.
4	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_B_4):** Applied to GPP_B4. Same description as bit 0.
3	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_B_3):** Applied to GPP_B3. Same description as bit 0.
2	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_B_2):** Applied to GPP_B2. Same description as bit 0.
1	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_B_1):** Applied to GPP_B1. Same description as bit 0.
0	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_B_0):** GPI Interrupt Status (GPI_INT_STS) This bit is set to '1' by hardware when either an edge or a level event is detected (See RxEdCfg RxInv) on pad and all the following conditions are true: - The corresponding pad is used in GPIO input mode - HOSTSW_OWN = 1 (i.e. Host GPIO Driver Mode). Writing a value of 1 will clear the bit while writing a value of 0 has no effect. 0 = No interrupt 1 = Interrupt asserts The state of GPI_INT_EN[x] does not prevent the setting of GPI_INT_STS[x].

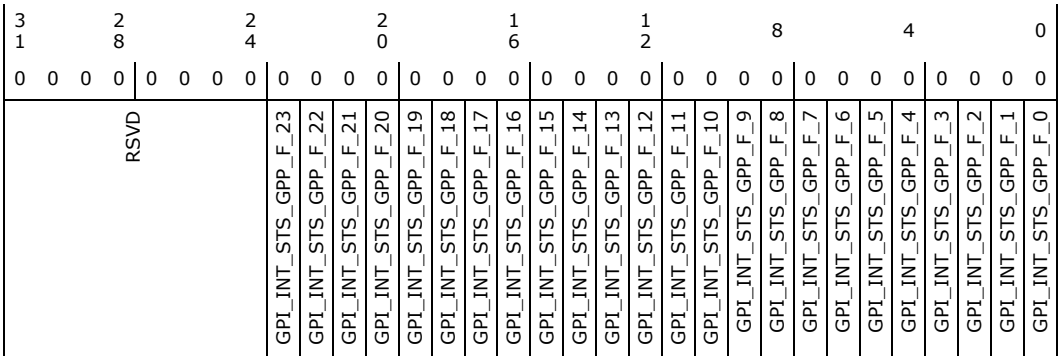
18.6.23 GPI Interrupt Status (GPI_IS_GPP_F)—Offset 108h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_F_23):** Applied to GPP_F23. Same description as bit 0.
22	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_F_22):** Applied to GPP_F22. Same description as bit 0.
21	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_F_21):** Applied to GPP_F21. Same description as bit 0.
20	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_F_20):** Applied to GPP_F20. Same description as bit 0.
19	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_F_19):** Applied to GPP_F19. Same description as bit 0.
18	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_F_18):** Applied to GPP_F18. Same description as bit 0.
17	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_F_17):** Applied to GPP_F17. Same description as bit 0.
16	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_F_16):** Applied to GPP_F16. Same description as bit 0.
15	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_F_15):** Applied to GPP_F15. Same description as bit 0.
14	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_F_14):** Applied to GPP_F14. Same description as bit 0.
13	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_F_13):** Applied to GPP_F13. Same description as bit 0.
12	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_F_12):** Applied to GPP_F12. Same description as bit 0.
11	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_F_11):** Applied to GPP_F11. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
10	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_F_10):** Applied to GPP_F10. Same description as bit 0.
9	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_F_9):** Applied to GPP_F9. Same description as bit 0.
8	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_F_8):** Applied to GPP_F8. Same description as bit 0.
7	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_F_7):** Applied to GPP_F7. Same description as bit 0.
6	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_F_6):** Applied to GPP_F6. Same description as bit 0.
5	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_F_5):** Applied to GPP_F5. Same description as bit 0.
4	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_F_4):** Applied to GPP_F4. Same description as bit 0.
3	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_F_3):** Applied to GPP_F3. Same description as bit 0.
2	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_F_2):** Applied to GPP_F2. Same description as bit 0.
1	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_F_1):** Applied to GPP_F1. Same description as bit 0.
0	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_F_0):** GPI Interrupt Status (GPI_INT_STS) This bit is set to '1' by hardware when either an edge or a level event is detected (See RxEdCfg RxInv) on pad and all the following conditions are true: - The corresponding pad is used in GPIO input mode - HOSTSW_OWN = 1 (i.e. Host GPIO Driver Mode). Writing a value of 1 will clear the bit while writing a value of 0 has no effect. 0 = No interrupt 1 = Interrupt asserts The state of GPI_INT_EN[x] does not prevent the setting of GPI_INT_STS[x].

18.6.24 GPI Interrupt Enable (GPI_IE_GPP_A)—Offset 110h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD								GPI_INT_EN_GPP_A_23	GPI_INT_EN_GPP_A_22	GPI_INT_EN_GPP_A_21	GPI_INT_EN_GPP_A_20	GPI_INT_EN_GPP_A_19	GPI_INT_EN_GPP_A_18	GPI_INT_EN_GPP_A_17	GPI_INT_EN_GPP_A_16	GPI_INT_EN_GPP_A_15	GPI_INT_EN_GPP_A_14	GPI_INT_EN_GPP_A_13	GPI_INT_EN_GPP_A_12	GPI_INT_EN_GPP_A_11	GPI_INT_EN_GPP_A_10	GPI_INT_EN_GPP_A_9	GPI_INT_EN_GPP_A_8	GPI_INT_EN_GPP_A_7	GPI_INT_EN_GPP_A_6	GPI_INT_EN_GPP_A_5	GPI_INT_EN_GPP_A_4	GPI_INT_EN_GPP_A_3	GPI_INT_EN_GPP_A_2	GPI_INT_EN_GPP_A_1	GPI_INT_EN_GPP_A_0

Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_A_23):** Applied to GPP_A23. Same description as bit 0.
22	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_A_22):** Applied to GPP_A22. Same description as bit 0.
21	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_A_21):** Applied to GPP_A21. Same description as bit 0.
20	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_A_20):** Applied to GPP_A20. Same description as bit 0.
19	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_A_19):** Applied to GPP_A19. Same description as bit 0.
18	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_A_18):** Applied to GPP_A18. Same description as bit 0.
17	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_A_17):** Applied to GPP_A17. Same description as bit 0.
16	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_A_16):** Applied to GPP_A16. Same description as bit 0.
15	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_A_15):** Applied to GPP_A15. Same description as bit 0.
14	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_A_14):** Applied to GPP_A14. Same description as bit 0.
13	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_A_13):** Applied to GPP_A13. Same description as bit 0.
12	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_A_12):** Applied to GPP_A12. Same description as bit 0.
11	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_A_11):** Applied to GPP_A11. Same description as bit 0.
10	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_A_10):** Applied to GPP_A10. Same description as bit 0.
9	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_A_9):** Applied to GPP_A9. Same description as bit 0.
8	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_A_8):** Applied to GPP_A8. Same description as bit 0.
7	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_A_7):** Applied to GPP_A7. Same description as bit 0.
6	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_A_6):** Applied to GPP_A6. Same description as bit 0.
5	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_A_5):** Applied to GPP_A5. Same description as bit 0.
4	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_A_4):** Applied to GPP_A3. Same description as bit 0.
3	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_A_3):** Applied to GPP_A3. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
2	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_A_2):** Applied to GPP_A2. Same description as bit 0.
1	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_A_1):** Applied to GPP_A1. Same description as bit 0.
0	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_A_0):** This bit is used to enable/ disable the generation of APIC interrupt when the corresponding GPI_INT_STS bit is set. 0 = disable interrupt generation 1 = enable interrupt generation

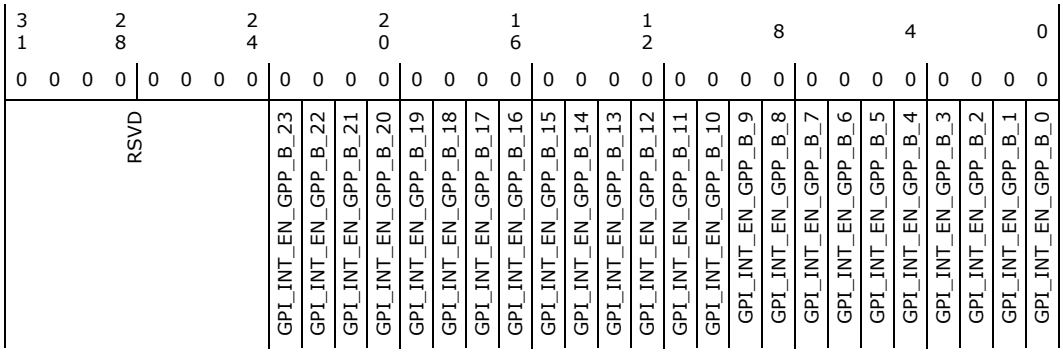
18.6.25 GPI Interrupt Enable (GPI_IE_GPP_B)—Offset 114h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_B_23):** Applied to GPP_B23. Same description as bit 0.
22	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_B_22):** Applied to GPP_B22. Same description as bit 0.
21	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_B_21):** Applied to GPP_B21. Same description as bit 0.
20	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_B_20):** Applied to GPP_B20. Same description as bit 0.
19	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_B_19):** Applied to GPP_B19. Same description as bit 0.
18	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_B_18):** Applied to GPP_B18. Same description as bit 0.
17	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_B_17):** Applied to GPP_B17. Same description as bit 0.
16	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_B_16):** Applied to GPP_B16. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
15	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_B_15):** Applied to GPP_B15. Same description as bit 0.
14	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_B_14):** Applied to GPP_B14. Same description as bit 0.
13	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_B_13):** Applied to GPP_B13. Same description as bit 0.
12	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_B_12):** Applied to GPP_B12. Same description as bit 0.
11	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_B_11):** Applied to GPP_B11. Same description as bit 0.
10	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_B_10):** Applied to GPP_B10. Same description as bit 0.
9	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_B_9):** Applied to GPP_B9. Same description as bit 0.
8	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_B_8):** Applied to GPP_B8. Same description as bit 0.
7	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_B_7):** Applied to GPP_B7. Same description as bit 0.
6	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_B_6):** Applied to GPP_B6. Same description as bit 0.
5	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_B_5):** Applied to GPP_B5. Same description as bit 0.
4	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_B_4):** Applied to GPP_B3. Same description as bit 0.
3	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_B_3):** Applied to GPP_B3. Same description as bit 0.
2	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_B_2):** Applied to GPP_B2. Same description as bit 0.
1	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_B_1):** Applied to GPP_B1. Same description as bit 0.
0	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_B_0):** This bit is used to enable/ disable the generation of APIC interrupt when the corresponding GPI_INT_STS bit is set. 0 = disable interrupt generation 1 = enable interrupt generation

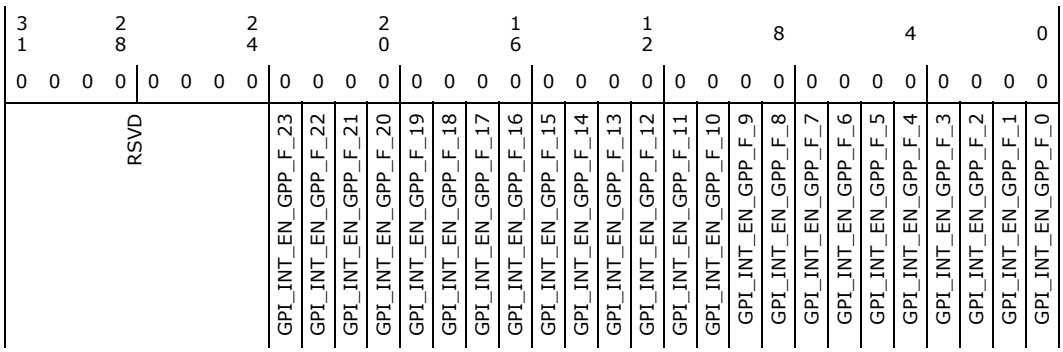
18.6.26 GPI Interrupt Enable (GPI_IE_GPP_F)—Offset 118h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_F_23):** Applied to GPP_F23. Same description as bit 0.
22	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_F_22):** Applied to GPP_F22. Same description as bit 0.
21	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_F_21):** Applied to GPP_F21. Same description as bit 0.
20	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_F_20):** Applied to GPP_F20. Same description as bit 0.
19	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_F_19):** Applied to GPP_F19. Same description as bit 0.
18	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_F_18):** Applied to GPP_F18. Same description as bit 0.
17	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_F_17):** Applied to GPP_F17. Same description as bit 0.
16	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_F_16):** Applied to GPP_F16. Same description as bit 0.
15	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_F_15):** Applied to GPP_F15. Same description as bit 0.
14	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_F_14):** Applied to GPP_F14. Same description as bit 0.
13	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_F_13):** Applied to GPP_F13. Same description as bit 0.
12	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_F_12):** Applied to GPP_F12. Same description as bit 0.
11	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_F_11):** Applied to GPP_F11. Same description as bit 0.
10	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_F_10):** Applied to GPP_F10. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
9	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_F_9):** Applied to GPP_F9. Same description as bit 0.
8	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_F_8):** Applied to GPP_F8. Same description as bit 0.
7	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_F_7):** Applied to GPP_F7. Same description as bit 0.
6	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_F_6):** Applied to GPP_F6. Same description as bit 0.
5	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_F_5):** Applied to GPP_F5. Same description as bit 0.
4	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_F_4):** Applied to GPP_F3. Same description as bit 0.
3	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_F_3):** Applied to GPP_F3. Same description as bit 0.
2	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_F_2):** Applied to GPP_F2. Same description as bit 0.
1	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_F_1):** Applied to GPP_F1. Same description as bit 0.
0	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_F_0):** This bit is used to enable/ disable the generation of APIC interrupt when the corresponding GPI_INT_STS bit is set. 0 = disable interrupt generation 1 = enable interrupt generation

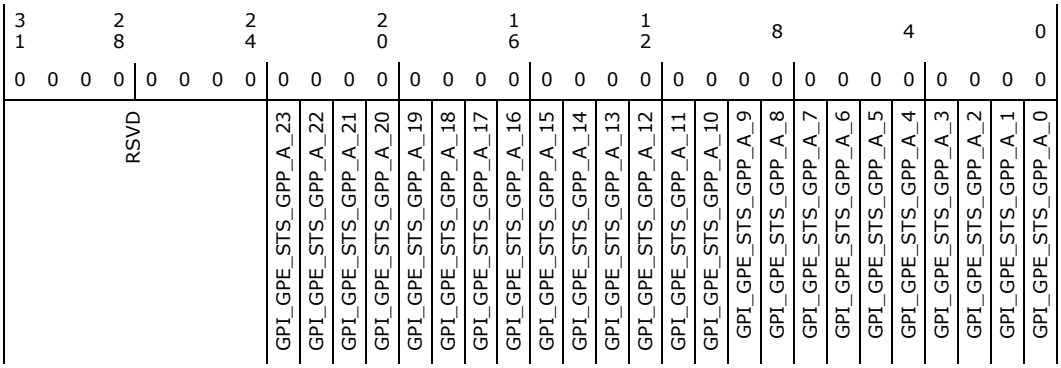
18.6.27 GPI General Purpose Events Status (GPI_GPE_STS_GPP_A)—Offset 120h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_A_23):** Applied to GPP_A23. Same description as bit 0.
22	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_A_22):** Applied to GPP_A22. Same description as bit 0.
21	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_A_21):** Applied to GPP_A21. Same description as bit 0.
20	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_A_20):** Applied to GPP_A20. Same description as bit 0.
19	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_A_19):** Applied to GPP_A19. Same description as bit 0.
18	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_A_18):** Applied to GPP_A18. Same description as bit 0.
17	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_A_17):** Applied to GPP_A17. Same description as bit 0.
16	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_A_16):** Applied to GPP_A16. Same description as bit 0.
15	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_A_15):** Applied to GPP_A15. Same description as bit 0.
14	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_A_14):** Applied to GPP_A14. Same description as bit 0.
13	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_A_13):** Applied to GPP_A13. Same description as bit 0.
12	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_A_12):** Applied to GPP_A12. Same description as bit 0.
11	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_A_11):** Applied to GPP_A11. Same description as bit 0.
10	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_A_10):** Applied to GPP_A10. Same description as bit 0.
9	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_A_9):** Applied to GPP_A9. Same description as bit 0.
8	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_A_8):** Applied to GPP_A8. Same description as bit 0.
7	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_A_7):** Applied to GPP_A7. Same description as bit 0.
6	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_A_6):** Applied to GPP_A6. Same description as bit 0.
5	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_A_5):** Applied to GPP_A5. Same description as bit 0.
4	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_A_4):** Applied to GPP_A4. Same description as bit 0.
3	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_A_3):** Applied to GPP_A3. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
2	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_A_2):** Applied to GPP_A2. Same description as bit 0.
1	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_A_1):** Applied to GPP_A1. Same description as bit 0.
0	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_A_0):** These bits are set any time the corresponding GPIO pad is set up as an input, under host ownership and the corresponding GPIO signal is high (or low if the corresponding RXINV bit is set). If the corresponding enable bit is set in the GPI_GPE_EN register, then when the GPI_GPE_STS bit is set: - If the system is in an S3-S5 state, the event will also wake the system. - If the system is in an S0 state (or upon waking back to an S0 state), an SCI will be caused, depending on the GPIRoutSCI bit for the corresponding pad. These bits are sticky bits and are cleared by writing a 1 back to this bit position. The state of GPI_GPE_EN does not prevent the setting of GPI_GPE_STS.

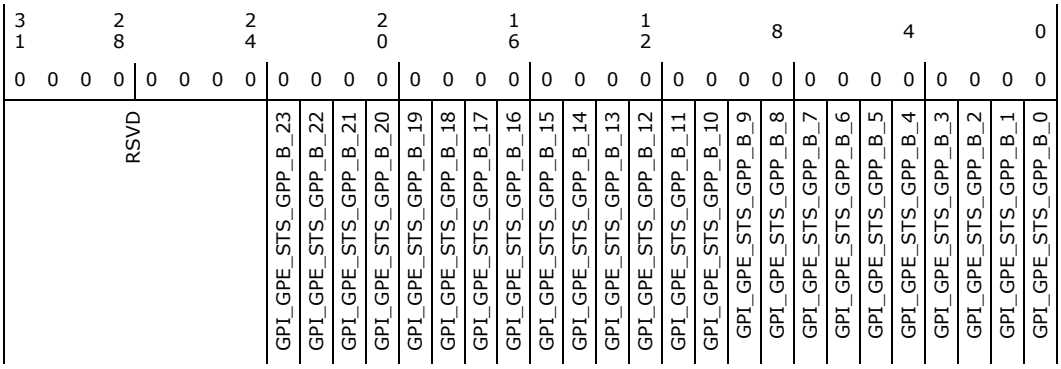
18.6.28 GPI General Purpose Events Status (GPI_GPE_STS_GPP_B)—Offset 124h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_B_23):** Applied to GPP_B23. Same description as bit 0.
22	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_B_22):** Applied to GPP_B22. Same description as bit 0.
21	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_B_21):** Applied to GPP_B21. Same description as bit 0.
20	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_B_20):** Applied to GPP_B20. Same description as bit 0.
19	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_B_19):** Applied to GPP_B19. Same description as bit 0.
18	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_B_18):** Applied to GPP_B18. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_B_17):** Applied to GPP_B17. Same description as bit 0.
16	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_B_16):** Applied to GPP_B16. Same description as bit 0.
15	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_B_15):** Applied to GPP_B15. Same description as bit 0.
14	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_B_14):** Applied to GPP_B14. Same description as bit 0.
13	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_B_13):** Applied to GPP_B13. Same description as bit 0.
12	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_B_12):** Applied to GPP_B12. Same description as bit 0.
11	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_B_11):** Applied to GPP_B11. Same description as bit 0.
10	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_B_10):** Applied to GPP_B10. Same description as bit 0.
9	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_B_9):** Applied to GPP_B9. Same description as bit 0.
8	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_B_8):** Applied to GPP_B8. Same description as bit 0.
7	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_B_7):** Applied to GPP_B7. Same description as bit 0.
6	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_B_6):** Applied to GPP_B6. Same description as bit 0.
5	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_B_5):** Applied to GPP_B5. Same description as bit 0.
4	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_B_4):** Applied to GPP_B4. Same description as bit 0.
3	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_B_3):** Applied to GPP_B3. Same description as bit 0.
2	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_B_2):** Applied to GPP_B2. Same description as bit 0.
1	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_B_1):** Applied to GPP_B1. Same description as bit 0.
0	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_B_0):** These bits are set any time the corresponding GPIO pad is set up as an input, under host ownership and the corresponding GPIO signal is high (or low if the corresponding RXINV bit is set). If the corresponding enable bit is set in the GPI_GPE_EN register, then when the GPI_GPE_STS bit is set: - If the system is in an S3-S5 state, the event will also wake the system. - If the system is in an S0 state (or upon waking back to an S0 state), an SCI will be caused, depending on the GPIRoutSCI bit for the corresponding pad. These bits are sticky bits and are cleared by writing a 1 back to this bit position. The state of GPI_GPE_EN does not prevent the setting of GPI_GPE_STS.

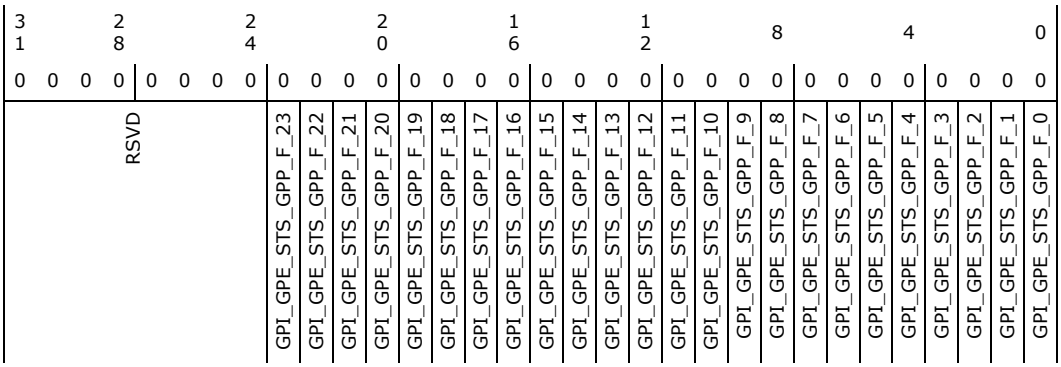
18.6.29 GPI General Purpose Events Status (GPI_GPE_STS_GPP_F)—Offset 128h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_F_23):** Applied to GPP_F23. Same description as bit 0.
22	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_F_22):** Applied to GPP_F22. Same description as bit 0.
21	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_F_21):** Applied to GPP_F21. Same description as bit 0.
20	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_F_20):** Applied to GPP_F20. Same description as bit 0.
19	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_F_19):** Applied to GPP_F19. Same description as bit 0.
18	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_F_18):** Applied to GPP_F18. Same description as bit 0.
17	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_F_17):** Applied to GPP_F17. Same description as bit 0.
16	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_F_16):** Applied to GPP_F16. Same description as bit 0.
15	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_F_15):** Applied to GPP_F15. Same description as bit 0.
14	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_F_14):** Applied to GPP_F14. Same description as bit 0.
13	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_F_13):** Applied to GPP_F13. Same description as bit 0.
12	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_F_12):** Applied to GPP_F12. Same description as bit 0.
11	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_F_11):** Applied to GPP_F11. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
10	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_F_10):** Applied to GPP_F10. Same description as bit 0.
9	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_F_9):** Applied to GPP_F9. Same description as bit 0.
8	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_F_8):** Applied to GPP_F8. Same description as bit 0.
7	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_F_7):** Applied to GPP_F7. Same description as bit 0.
6	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_F_6):** Applied to GPP_F6. Same description as bit 0.
5	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_F_5):** Applied to GPP_F5. Same description as bit 0.
4	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_F_4):** Applied to GPP_F4. Same description as bit 0.
3	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_F_3):** Applied to GPP_F3. Same description as bit 0.
2	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_F_2):** Applied to GPP_F2. Same description as bit 0.
1	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_F_1):** Applied to GPP_F1. Same description as bit 0.
0	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_F_0):** These bits are set any time the corresponding GPIO pad is set up as an input, under host ownership and the corresponding GPIO signal is high (or low if the corresponding RXINV bit is set). If the corresponding enable bit is set in the GPI_GPE_EN register, then when the GPI_GPE_STS bit is set: - If the system is in an S3-S5 state, the event will also wake the system. - If the system is in an S0 state (or upon waking back to an S0 state), an SCI will be caused, depending on the GPIRoutSCI bit for the corresponding pad. These bits are sticky bits and are cleared by writing a 1 back to this bit position. The state of GPI_GPE_EN does not prevent the setting of GPI_GPE_STS.

18.6.30 GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A)—Offset 130h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD								GPI_GPE_EN_GPP_A_23	GPI_GPE_EN_GPP_A_22	GPI_GPE_EN_GPP_A_21	GPI_GPE_EN_GPP_A_20	GPI_GPE_EN_GPP_A_19	GPI_GPE_EN_GPP_A_18	GPI_GPE_EN_GPP_A_17	GPI_GPE_EN_GPP_A_16	GPI_GPE_EN_GPP_A_15	GPI_GPE_EN_GPP_A_14	GPI_GPE_EN_GPP_A_13	GPI_GPE_EN_GPP_A_12	GPI_GPE_EN_GPP_A_11	GPI_GPE_EN_GPP_A_10	GPI_GPE_EN_GPP_A_9	GPI_GPE_EN_GPP_A_8	GPI_GPE_EN_GPP_A_7	GPI_GPE_EN_GPP_A_6	GPI_GPE_EN_GPP_A_5	GPI_GPE_EN_GPP_A_4	GPI_GPE_EN_GPP_A_3	GPI_GPE_EN_GPP_A_2	GPI_GPE_EN_GPP_A_1	GPI_GPE_EN_GPP_A_0

Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A_23):** Applied to GPP_A23. Same description as bit 0.
22	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A_22):** Applied to GPP_A22. Same description as bit 0.
21	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A_21):** Applied to GPP_A21. Same description as bit 0.
20	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A_20):** Applied to GPP_A20. Same description as bit 0.
19	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A_19):** Applied to GPP_A19. Same description as bit 0.
18	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A_18):** Applied to GPP_A18. Same description as bit 0.
17	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A_17):** Applied to GPP_A17. Same description as bit 0.
16	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A_16):** Applied to GPP_A16. Same description as bit 0.
15	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A_15):** Applied to GPP_A15. Same description as bit 0.
14	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A_14):** Applied to GPP_A14. Same description as bit 0.
13	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A_13):** Applied to GPP_A13. Same description as bit 0.
12	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A_12):** Applied to GPP_A12. Same description as bit 0.
11	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A_11):** Applied to GPP_A11. Same description as bit 0.
10	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A_10):** Applied to GPP_A10. Same description as bit 0.
9	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A_9):** Applied to GPP_A9. Same description as bit 0.
8	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A_8):** Applied to GPP_A8. Same description as bit 0.
7	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A_7):** Applied to GPP_A7. Same description as bit 0.
6	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A_6):** Applied to GPP_A6. Same description as bit 0.
5	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A_5):** Applied to GPP_A5. Same description as bit 0.
4	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A_4):** Applied to GPP_A4. Same description as bit 0.
3	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A_3):** Applied to GPP_A3. Same description as bit 0.
2	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A_2):** Applied to GPP_A2. Same description as bit 0.
1	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A_1):** Applied to GPP_A1. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
0	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_A_0):** This bit is used to enable/disable the generation of GPE to cause SCI and/or wake when the corresponding GPI_GPE_STS bit is set. 0 = Disable GPE generation 1 = Enable GPE generation ***Note:*** The pad must also be routed for GPE functionality in order for GPE to be generated, i.e., the corresponding GPIRoutSCI must be set to 1.

18.6.31 GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B)—Offset 134h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD								GPI_GPE_EN_GPP_B_23	GPI_GPE_EN_GPP_B_22	GPI_GPE_EN_GPP_B_21	GPI_GPE_EN_GPP_B_20	GPI_GPE_EN_GPP_B_19	GPI_GPE_EN_GPP_B_18	GPI_GPE_EN_GPP_B_17	GPI_GPE_EN_GPP_B_16	GPI_GPE_EN_GPP_B_15	GPI_GPE_EN_GPP_B_14	GPI_GPE_EN_GPP_B_13	GPI_GPE_EN_GPP_B_12	GPI_GPE_EN_GPP_B_11	GPI_GPE_EN_GPP_B_10	GPI_GPE_EN_GPP_B_9	GPI_GPE_EN_GPP_B_8	GPI_GPE_EN_GPP_B_7	GPI_GPE_EN_GPP_B_6	GPI_GPE_EN_GPP_B_5	GPI_GPE_EN_GPP_B_4	GPI_GPE_EN_GPP_B_3	GPI_GPE_EN_GPP_B_2	GPI_GPE_EN_GPP_B_1	GPI_GPE_EN_GPP_B_0

Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B_23):** Applied to GPP_B23. Same description as bit 0.
22	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B_22):** Applied to GPP_B22. Same description as bit 0.
21	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B_21):** Applied to GPP_B21. Same description as bit 0.
20	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B_20):** Applied to GPP_B20. Same description as bit 0.
19	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B_19):** Applied to GPP_B19. Same description as bit 0.
18	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B_18):** Applied to GPP_B18. Same description as bit 0.
17	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B_17):** Applied to GPP_B17. Same description as bit 0.
16	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B_16):** Applied to GPP_B16. Same description as bit 0.
15	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B_15):** Applied to GPP_B15. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
14	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B_14):** Applied to GPP_B14. Same description as bit 0.
13	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B_13):** Applied to GPP_B13. Same description as bit 0.
12	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B_12):** Applied to GPP_B12. Same description as bit 0.
11	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B_11):** Applied to GPP_B11. Same description as bit 0.
10	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B_10):** Applied to GPP_B10. Same description as bit 0.
9	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B_9):** Applied to GPP_B9. Same description as bit 0.
8	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B_8):** Applied to GPP_B8. Same description as bit 0.
7	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B_7):** Applied to GPP_B7. Same description as bit 0.
6	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B_6):** Applied to GPP_B6. Same description as bit 0.
5	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B_5):** Applied to GPP_B5. Same description as bit 0.
4	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B_4):** Applied to GPP_B4. Same description as bit 0.
3	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B_3):** Applied to GPP_B3. Same description as bit 0.
2	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B_2):** Applied to GPP_B2. Same description as bit 0.
1	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B_1):** Applied to GPP_B1. Same description as bit 0.
0	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_B_0):** This bit is used to enable/disable the generation of GPE to cause SCI and/or wake when the corresponding GPI_GPE_STS bit is set. 0 = Disable GPE generation 1 = Enable GPE generation ***Note:*** The pad must also be routed for GPE functionality in order for GPE to be generated, i.e., the corresponding GPIRoutSCI must be set to 1.

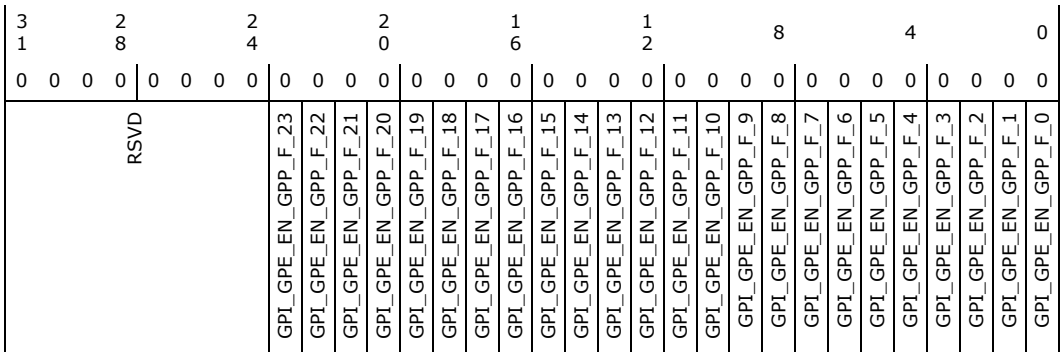
18.6.32 GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F)—Offset 138h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F_23):** Applied to GPP_F23. Same description as bit 0.
22	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F_22):** Applied to GPP_F22. Same description as bit 0.
21	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F_21):** Applied to GPP_F21. Same description as bit 0.
20	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F_20):** Applied to GPP_F20. Same description as bit 0.
19	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F_19):** Applied to GPP_F19. Same description as bit 0.
18	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F_18):** Applied to GPP_F18. Same description as bit 0.
17	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F_17):** Applied to GPP_F17. Same description as bit 0.
16	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F_16):** Applied to GPP_F16. Same description as bit 0.
15	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F_15):** Applied to GPP_F15. Same description as bit 0.
14	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F_14):** Applied to GPP_F14. Same description as bit 0.
13	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F_13):** Applied to GPP_F13. Same description as bit 0.
12	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F_12):** Applied to GPP_F12. Same description as bit 0.
11	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F_11):** Applied to GPP_F11. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
10	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F_10):** Applied to GPP_F10. Same description as bit 0.
9	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F_9):** Applied to GPP_F9. Same description as bit 0.
8	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F_8):** Applied to GPP_F8. Same description as bit 0.
7	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F_7):** Applied to GPP_F7. Same description as bit 0.
6	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F_6):** Applied to GPP_F6. Same description as bit 0.
5	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F_5):** Applied to GPP_F5. Same description as bit 0.
4	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F_4):** Applied to GPP_F4. Same description as bit 0.
3	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F_3):** Applied to GPP_F3. Same description as bit 0.
2	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F_2):** Applied to GPP_F2. Same description as bit 0.
1	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F_1):** Applied to GPP_F1. Same description as bit 0.
0	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_F_0):** This bit is used to enable/disable the generation of GPE to cause SCI and/or wake when the corresponding GPI_GPE_STS bit is set. 0 = Disable GPE generation 1 = Enable GPE generation ***Note:*** The pad must also be routed for GPE functionality in order for GPE to be generated i.e., the corresponding GPIRoutSCI must be set to 1.

18.6.33 SMI Status (GPI_SMI_STS_GPP_B)—Offset 144h

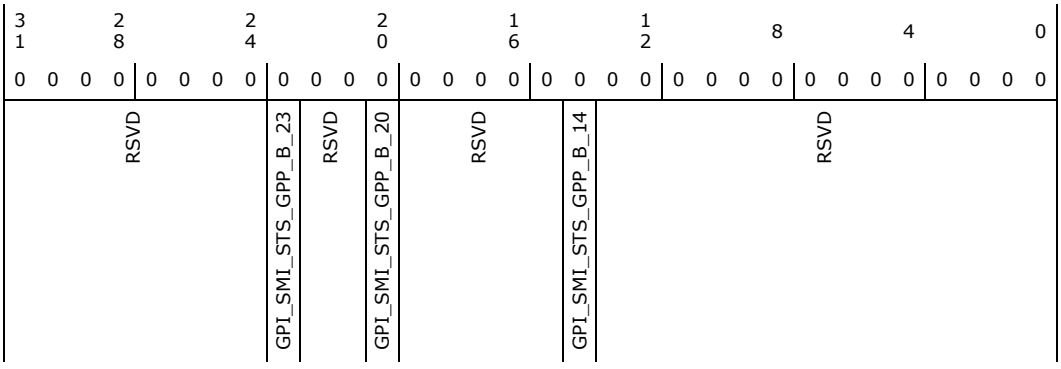
Register bits in this register are implemented for GPP_B signals that have SMI capability only. Other bits are reserved and RO.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW1C	**GPI SMI Status (GPI_SMI_STS_GPP_B_23):** Same description as bit 14.
22:21	0h RO	Reserved
20	0h RW1C	**GPI SMI Status (GPI_SMI_STS_GPP_B_20):** Same description as bit 14.
19:15	0h RO	Reserved
14	0h RW1C	**GPI SMI Status (GPI_SMI_STS_GPP_B_14):** This bit is set to 1 by hardware when a level event (See RxEdCfg,RxInv) is detected, and all the following conditions are true: - The corresponding pad is used in GPIO input mode. - The corresponding PAD_OWN[2:0] is '000' (i.e., ACPI GPIO Mode). If the following conditions are true, then an SMI will be generated if the GPI_SMI_STS bit is set: 1. The corresponding bit in the GPI_SMI_EN register is set. 2. The corresponding pad's GPIROUTSMI is set. Writing a value of 1 will clear the bit while writing a value of 0 has no effect. 0 = There is no SMI event. 1 = There is an SMI event. The state of GPI_SMI_EN does not prevent the setting of GPI_SMI_STS. Defaults for these bits are dependent on the state of the GPI pads.
13:0	0h RO	Reserved

18.6.34 SMI Enable (GPI_SMI_EN_GPP_B)—Offset 154h

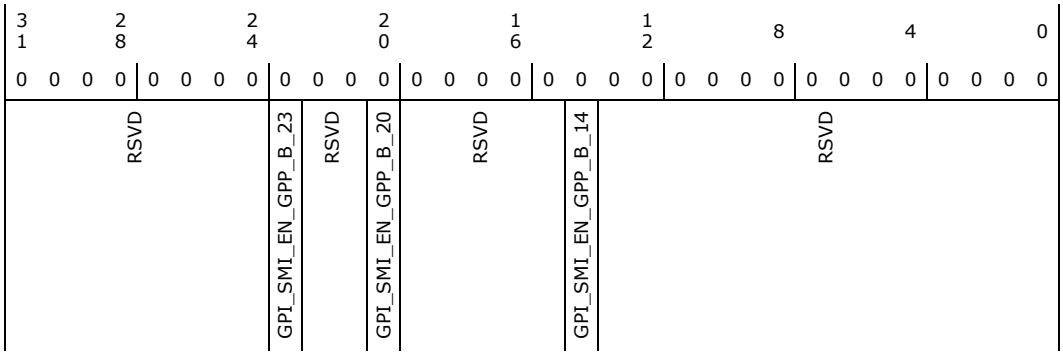
Register bits in this register are implemented for GPP_B signals that have SMI capability only. Other bits are reserved and RO.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**GPI SMI Enable (GPI_SMI_EN_GPP_B_23):** Same description as bit 14.
22:21	0h RO	Reserved
20	0h RW	**GPI SMI Enable (GPI_SMI_EN_GPP_B_20):** Same description as bit 14.
19:15	0h RO	Reserved
14	0h RW	**GPI SMI Enable (GPI_SMI_EN_GPP_B_14):** This bit is used to enable/disable the generation of SMI when the corresponding GPI_SMI_STS bit is set. The pad must also be routed for SMI functionality in order for SMI to be generated, i.e., the corresponding GPIROUTSMI must be set to 1. 0 = Disable SMI generation 1 = Enable SMI generation ***Note:*** Each of the bit is lock-able dependent on the associated Pad Configuration register PadCfgLock setting. For example, if the PadCfgLock of pad0 is 1, bit 0 of this bit is locked down to read-only.
13:0	0h RO	Reserved

18.6.35 NMI Status (GPI_NMI_STS_GPP_B)—Offset 164h

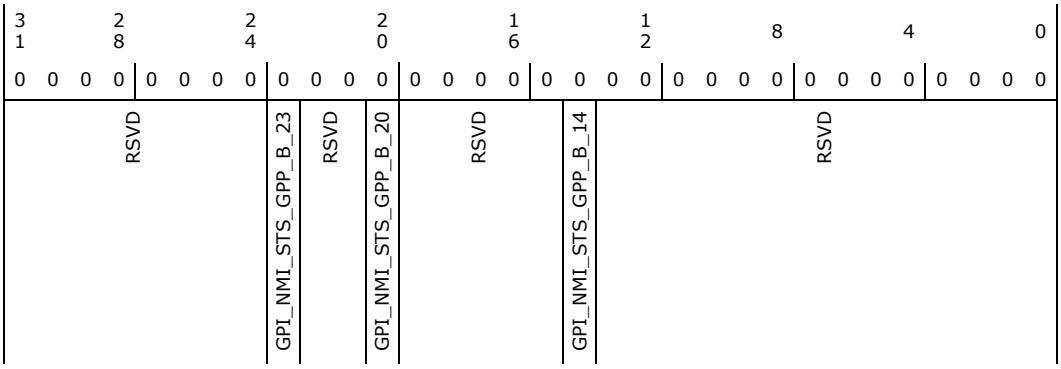
Register bits in this register are implemented for GPP_B signals that have NMI capability only. Other bits are reserved and RO.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW1C	**GPI NMI Status (GPI_NMI_STS_GPP_B_23):** Same description as bit 14.
22:21	0h RO	Reserved
20	0h RW1C	**GPI NMI Status (GPI_NMI_STS_GPP_B_20):** Same description as bit 14.
19:15	0h RO	Reserved
14	0h RW1C	**GPI NMI Status (GPI_NMI_STS_GPP_B_14):** This bit is set to 1 by hardware when an edge event is detected (See RxEdCfg, RxInv) on pad and all the following conditions are true: - The corresponding pad is used in GPIO input mode (PMode). - The corresponding GPIONMIRout is set to 1, i.e., programmed to route as NMI. - The corresponding GPIOOwn[2:0] is '000' (i.e. ACPI GPIO Mode). - The corresponding GPI_NMI_EN is set. Writing a value of 1 will clear the bit while writing a value of 0 has no effect. 0 = There is no NMI event. 1 = There is an NMI event.
13:0	0h RO	Reserved

18.6.36 NMI Enable (GPI_NMI_EN_GPP_B)—Offset 174h

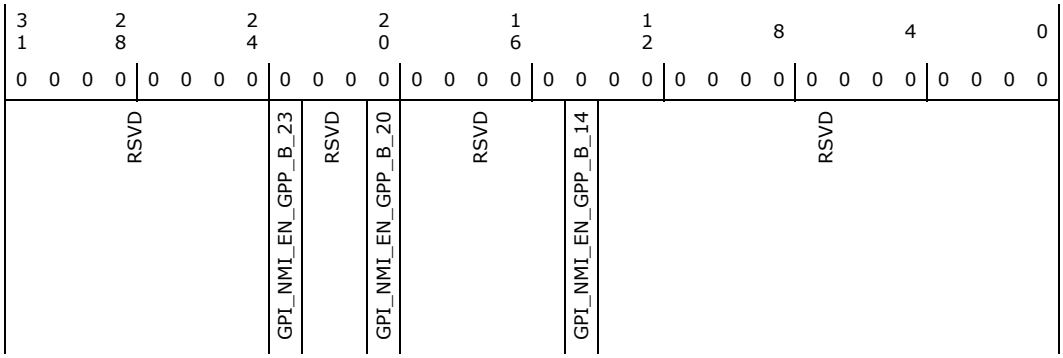
Register bits in this register are implemented for GPP_B signals that have NMI capability only. Other bits are reserved and RO.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**GPI NMI Enable (GPI_NMI_EN_GPP_B_23):** Same description as bit 14.
22:21	0h RO	Reserved
20	0h RW	**GPI NMI Enable (GPI_NMI_EN_GPP_B_20):** Same description as bit 14.
19:15	0h RO	Reserved
14	0h RW	**GPI NMI Enable (GPI_NMI_EN_GPP_B_14):** This bit is used to enable/disable the generation of NMI when the corresponding GPI_NMI_STS bit is set and its GPIROUTNMI is set. 0 = Disable NMI generation 1 = Enable NMI generation Each bit is lock-able dependent on the associated Pad Configuration register PadCfgLock setting. For example, if the PadCfgLock of pad0 is 1, bit 0 of this bit is locked down to read-only.
13:0	0h RO	Reserved

18.6.37 Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_0)—Offset 400h

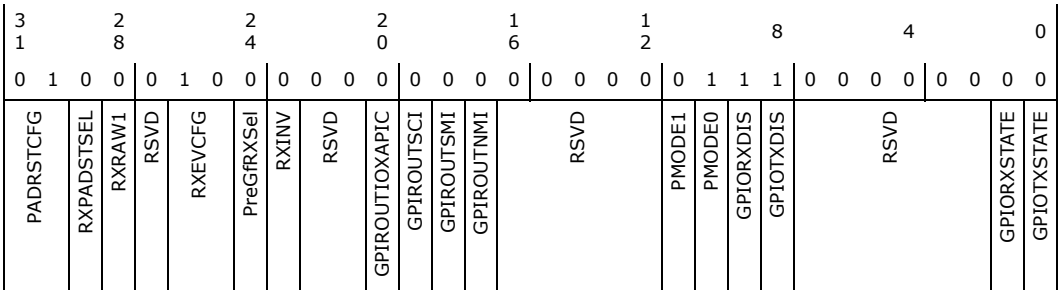
This register applies to GPP_A0.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).

Bit Range	Default & Access	Field Name (ID): Description
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0s in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.6.38 Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_0)—Offset 404h

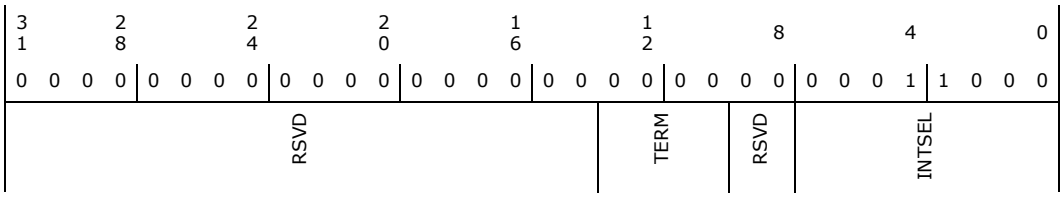
This register applies to GPP_A0.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 18h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	18h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.39 Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_1)—Offset 408h

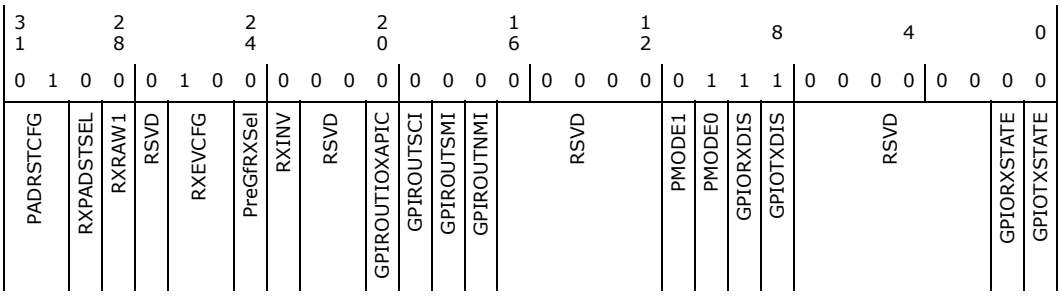
This register applies to GPP_A1.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ. 1 = Routing can cause peripheral IRQ. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.6.40 Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_1)—Offset 40Ch

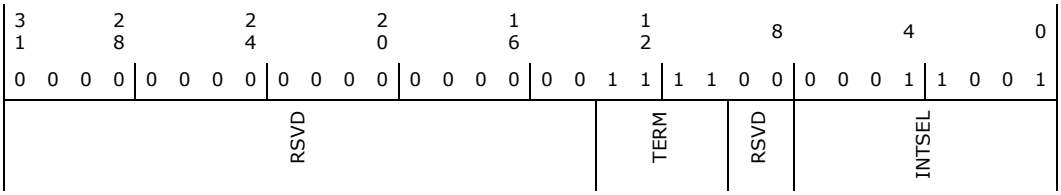
This register applies to GPP_A1.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003C19h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	Fh RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu 1111: Native controller selected in Pad mode All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	19h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.41 Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_2)—Offset 410h

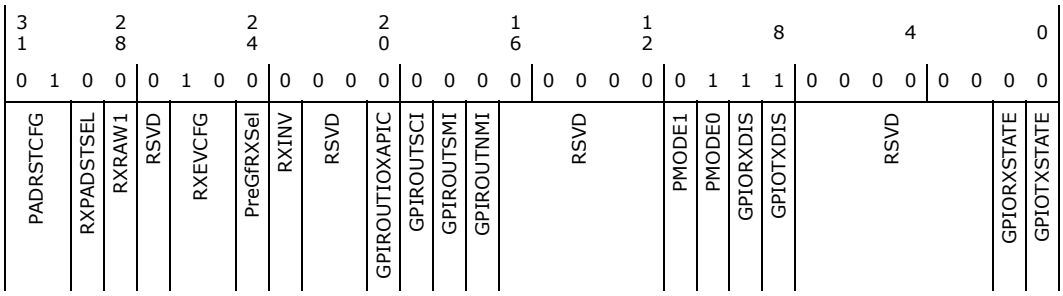
This register applies to GPP_A2.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.6.42 Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_2)—Offset 414h

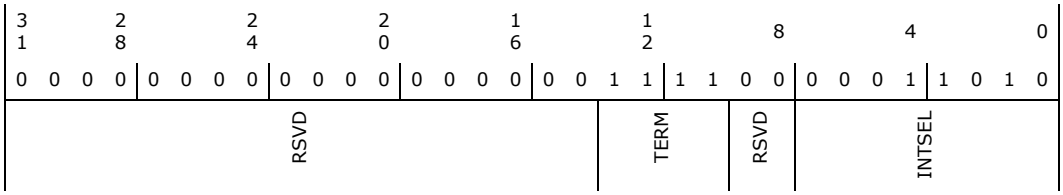
This register applies to GPP_A2.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003C1Ah



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	Fh RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu 1111: Native controller selected by Pad Mode All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Ah RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.43 Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_3)—Offset 418h

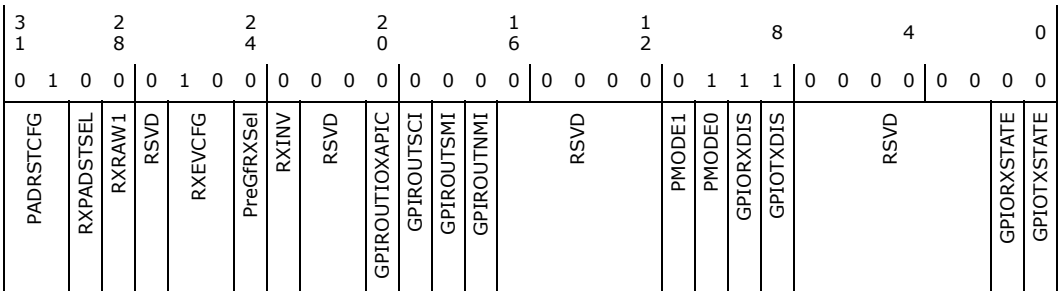
This register applies to GPP_A3.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ. 1 = Routing can cause peripheral IRQ. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.6.44 Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_3)—Offset 41Ch

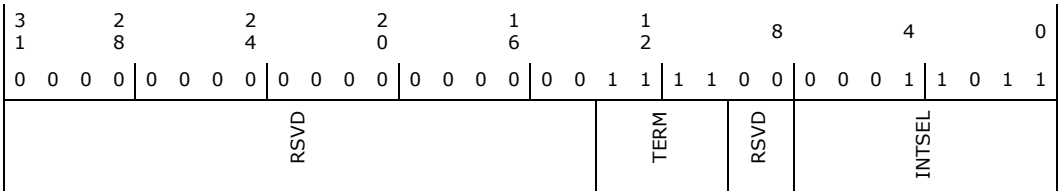
This register applies to GPP_A3.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003C1Bh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	Fh RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu 1111:Native controller selected by Pad Mode All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Bh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.45 Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_4)—Offset 420h

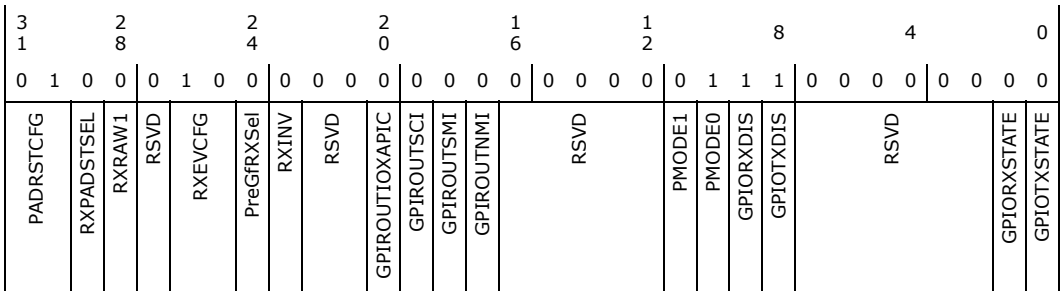
This register applies to GPP_A4.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.6.46 Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_4)—Offset 424h

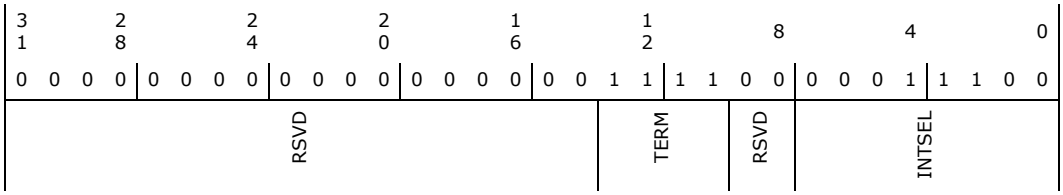
This register applies to GPP_A4

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003C1Ch



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	Fh RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu 1111: Native mode controller selected by Pad Mode All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Ch RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.6.47 Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_5)—Offset 428h

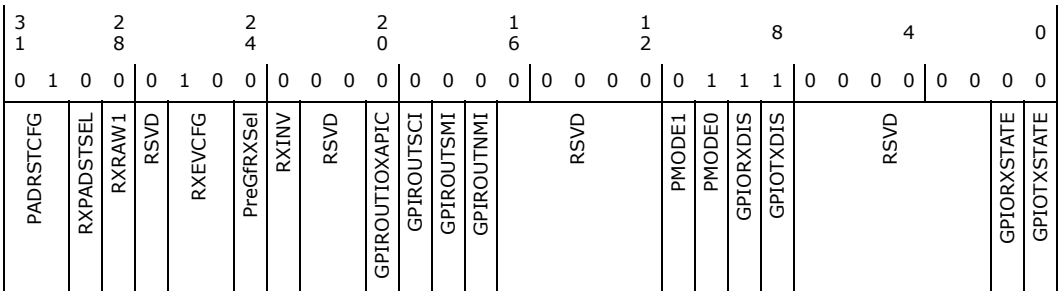
This register applies to GPP_A5.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.48 Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_5)—Offset 42Ch

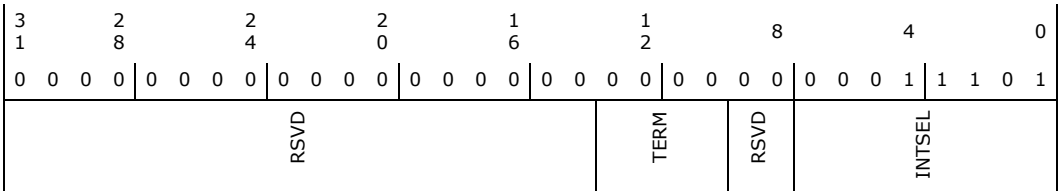
This register applies to GPP_A5.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Dh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Dh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.49 Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_6)—Offset 430h

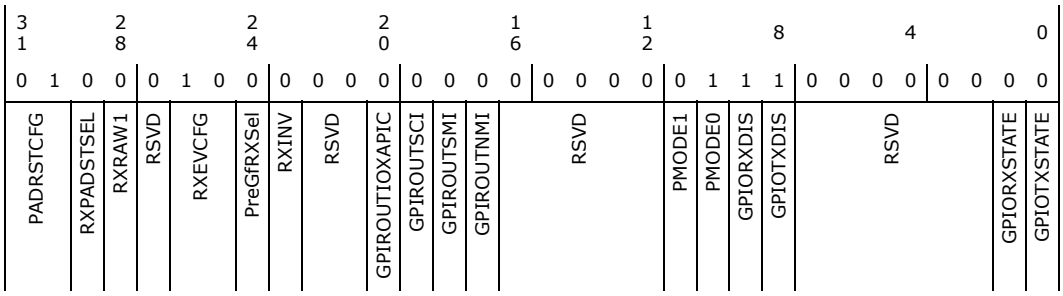
This register applies to GPP_A6.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.50 Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_6)—Offset 434h

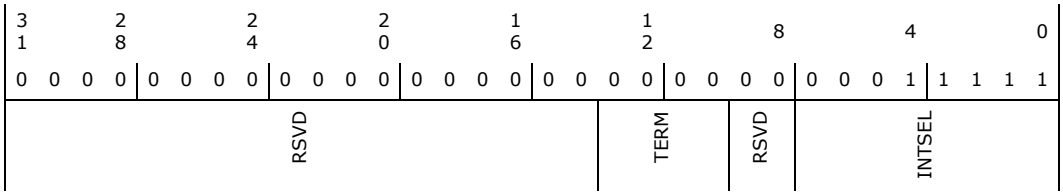
This register applies to GPP_A6

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Eh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Eh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.51 Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_7)—Offset 438h

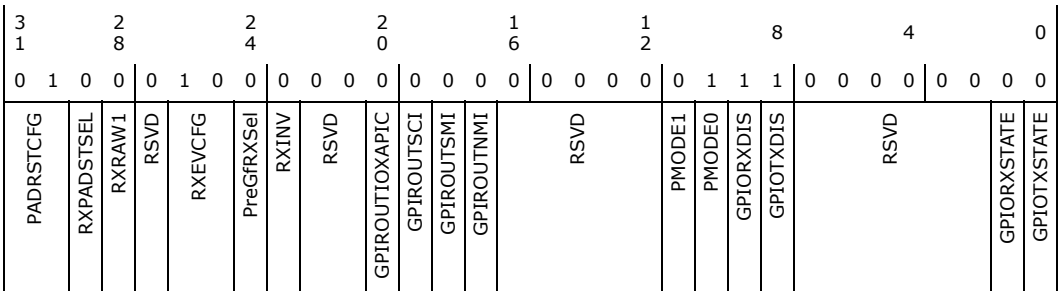
This register applies to GPP_A7.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.52 Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_7)—Offset 43Ch

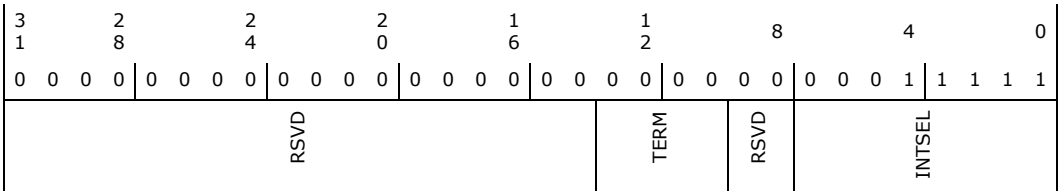
This register applies to GPP_A7.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Fh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Fh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.53 Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_8)—Offset 440h

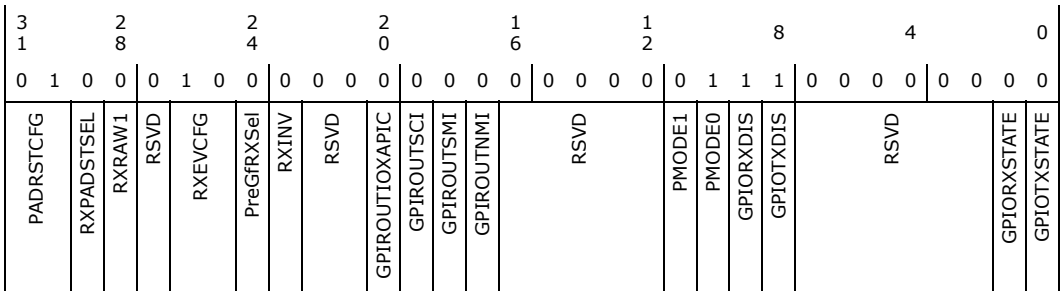
This register applies to GPP_A8.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ. 1 = Routing can cause peripheral IRQ. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.6.54 Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_8)—Offset 444h

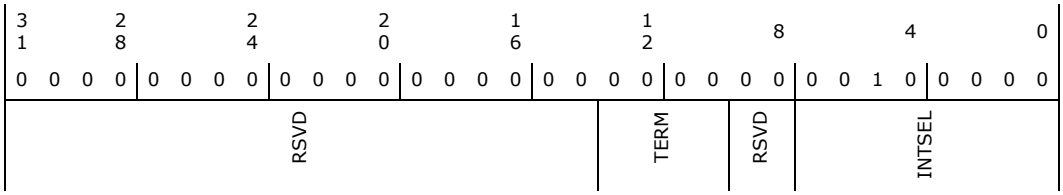
This register applies to GPP_A8.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 20h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	20h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.55 Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_9)—Offset 448h

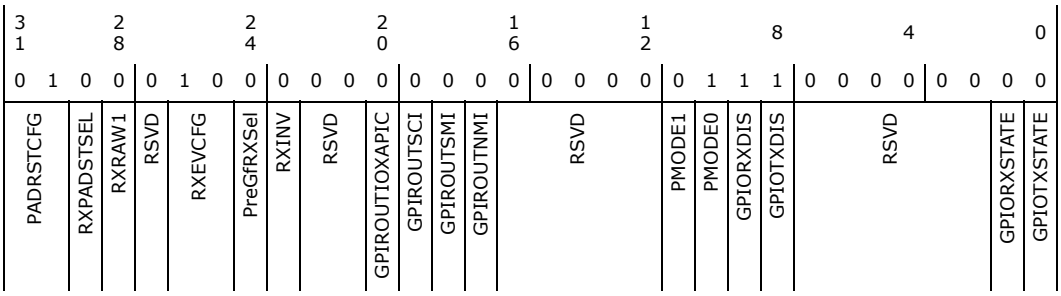
This register applies to GPP_A9.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ. 1 = Routing can cause peripheral IRQ. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.56 Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_9)—Offset 44Ch

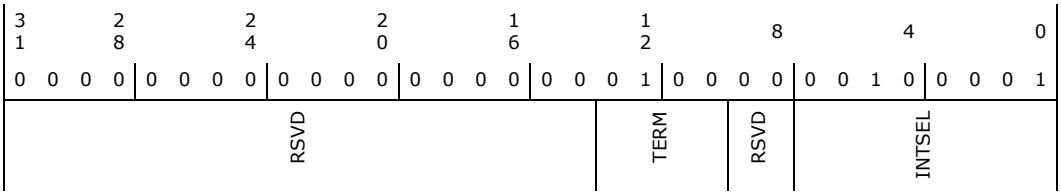
This register applies to GPP_A9.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00001021h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0100b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	21h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.57 Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_10)—Offset 450h

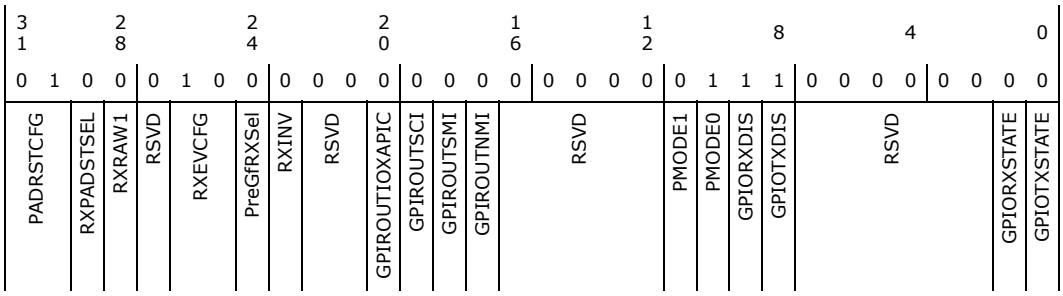
This register applies to GPP_A10.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ. 1 = Routing can cause peripheral IRQ. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.58 Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_10)—Offset 454h

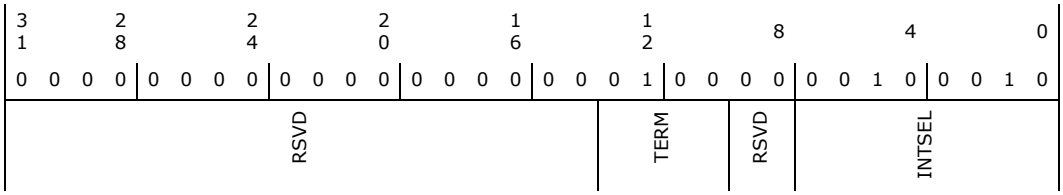
This register applies to GPP_A10.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00001022h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0100b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	22h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.59 Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_11)—Offset 458h

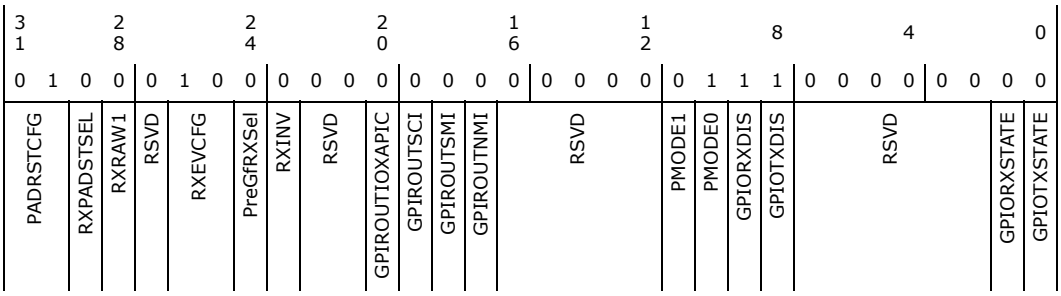
This register applies to GPP_A11.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ. 1 = Routing can cause peripheral IRQ. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.6.60 Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_11)—Offset 45Ch

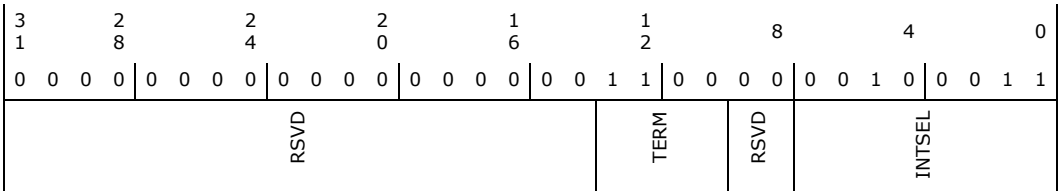
This register applies to GPP_A11.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003023h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	1100b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	23h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.61 Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_12)—Offset 460h

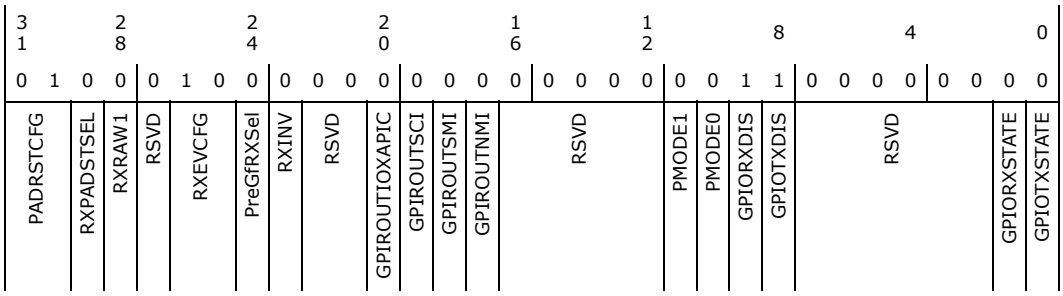
This register applies to GPP_A12.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e., Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ. 1 = Routing can cause peripheral IRQ. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.6.62 Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_12)—Offset 464h

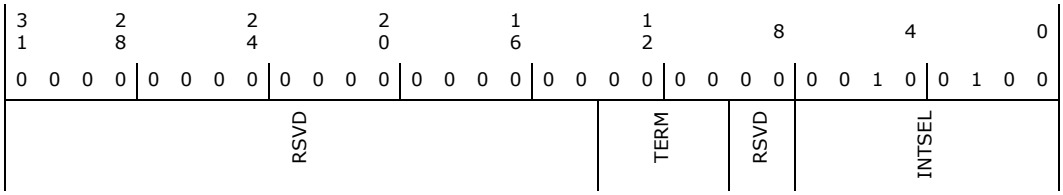
This register applies to GPP_A12.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 24h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	24h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.63 Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_13)—Offset 468h

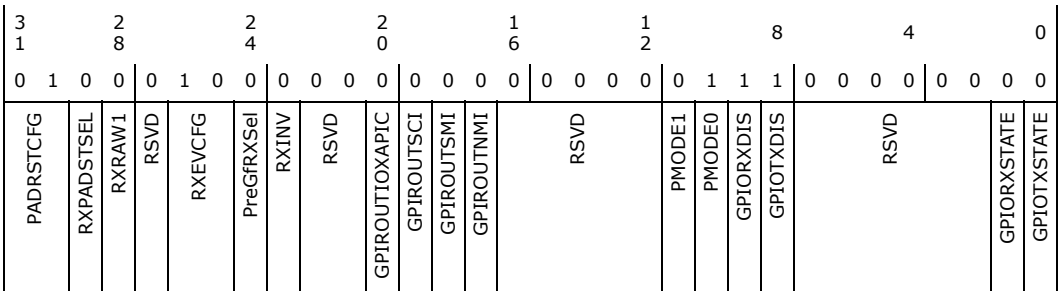
This register applies to GPP_A13.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ. 1 = Routing can cause peripheral IRQ. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.64 Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_13)—Offset 46Ch

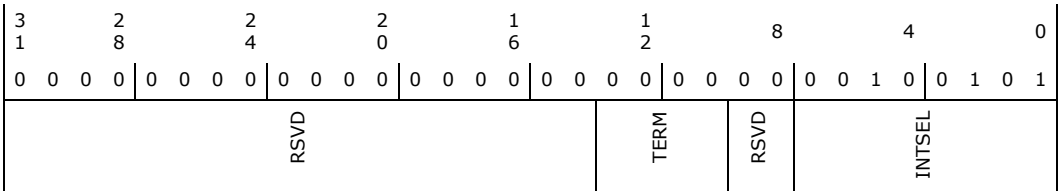
This register applies to GPP_A13.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 25h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	25h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.65 Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_14)—Offset 470h

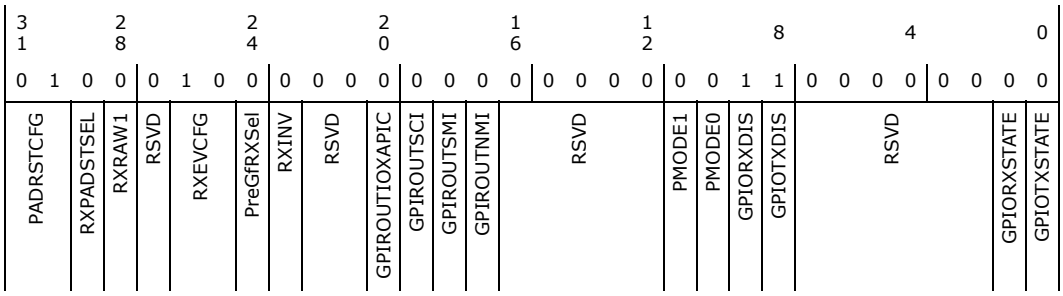
This register applies to GPP_A14.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e., Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ. 1 = Routing can cause peripheral IRQ. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.6.66 Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_14)—Offset 474h

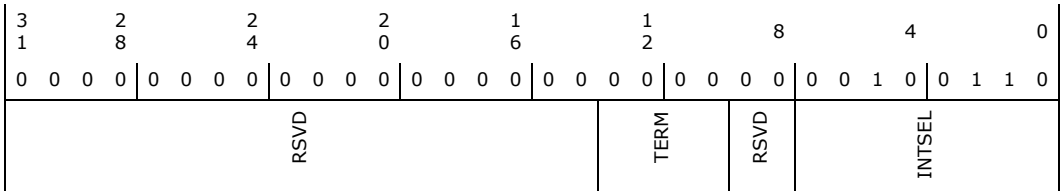
This register applies to GPP_A14.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 26h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	26h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.67 Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_15)—Offset 478h

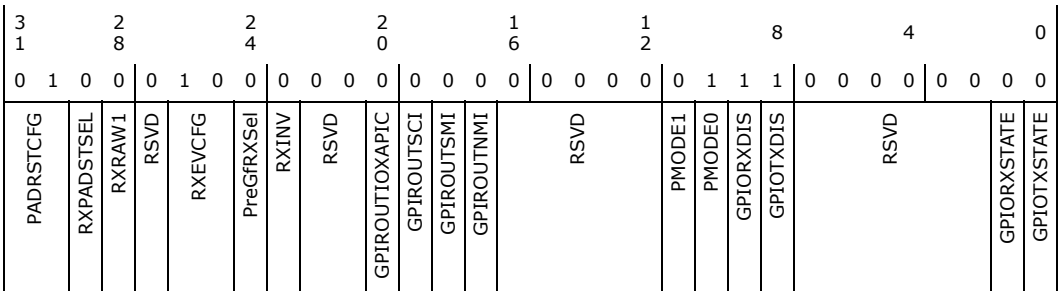
This register applies to GPP_A15.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.6.68 Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_15)—Offset 47Ch

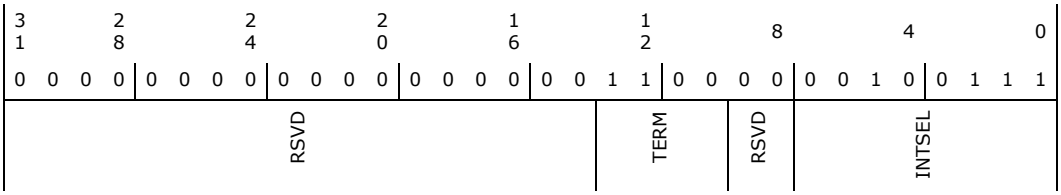
This register applies to GPP_A15.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003027h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	1100b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	27h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.69 Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_16)—Offset 480h

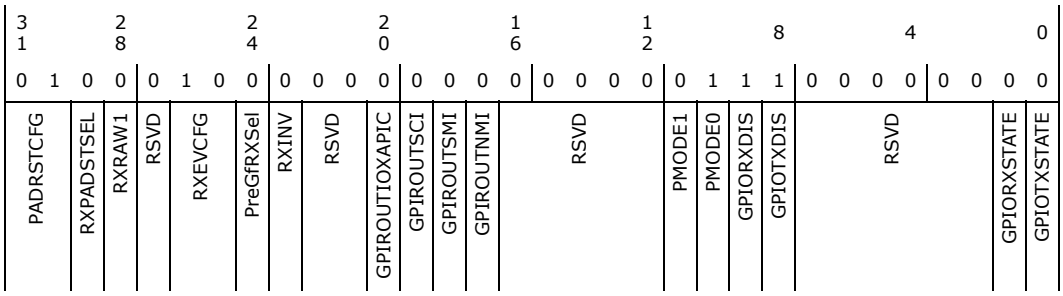
This register applies to GPP_A16.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e., Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ. 1 = Routing can cause peripheral IRQ. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.70 Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_16)—Offset 484h

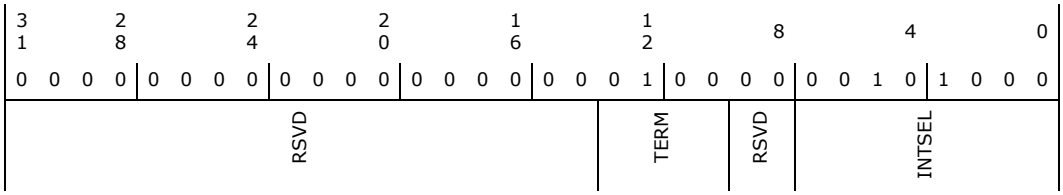
This register applies to GPP_A16.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00001028h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0100b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	28h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.71 Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_17)—Offset 488h

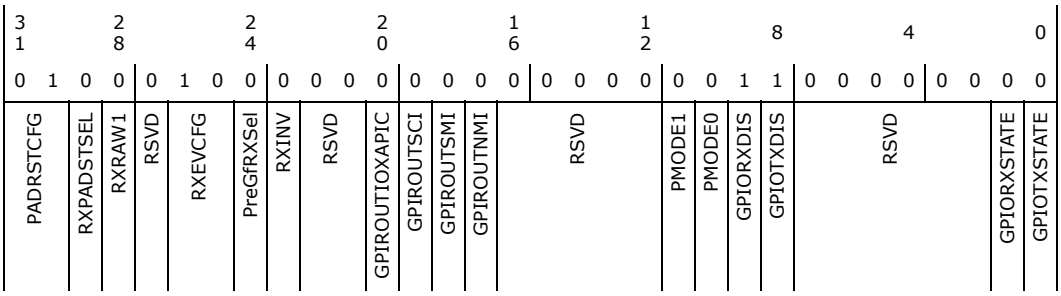
This register applies to GPP_A17.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.6.72 Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_17)—Offset 48Ch

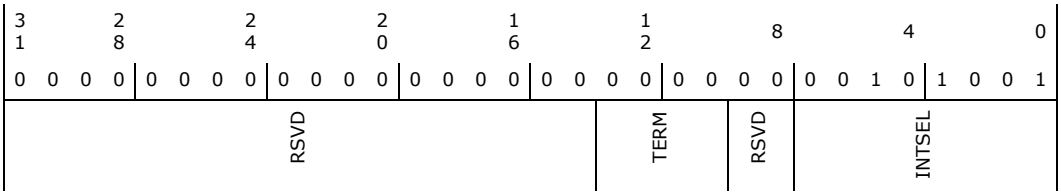
This register applies to GPP_A17.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 29h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	29h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.73 Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_18)—Offset 490h

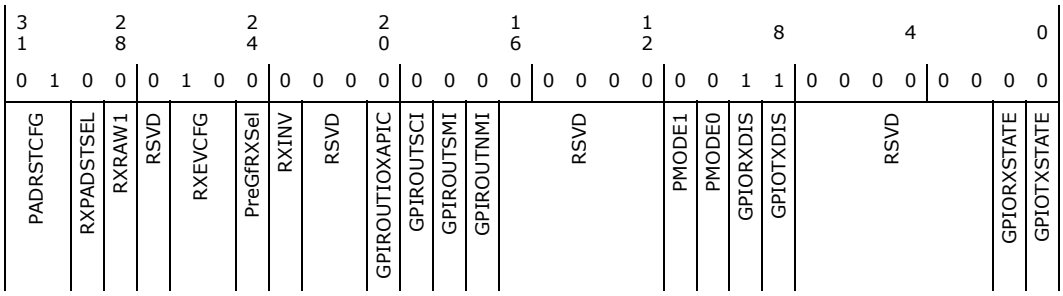
This register applies to GPP_A18.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.74 Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_18)—Offset 494h

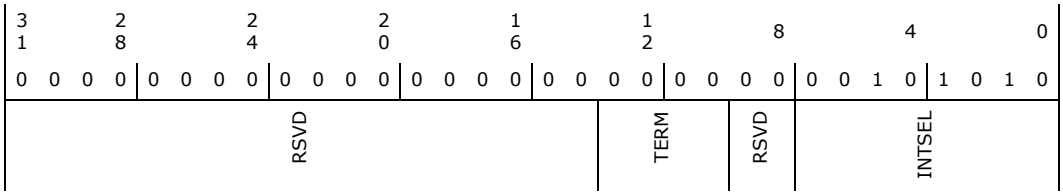
This register applies to GPP_A18.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 2Ah



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	2Ah RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.75 Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_19)—Offset 498h

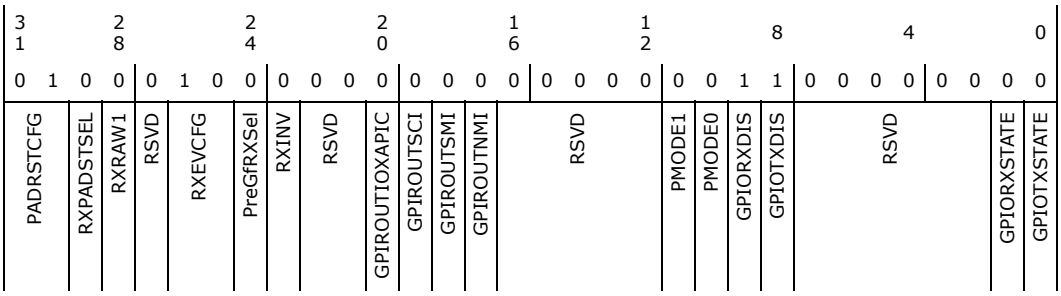
This register applies to GPP_A19.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.76 Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_19)—Offset 49Ch

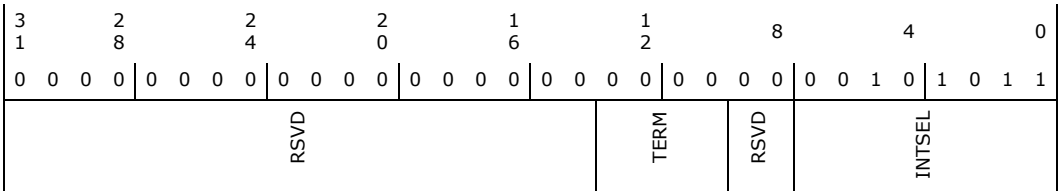
This register applies to GPP_A19.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 2Bh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	2Bh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.77 Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_20)—Offset 4A0h

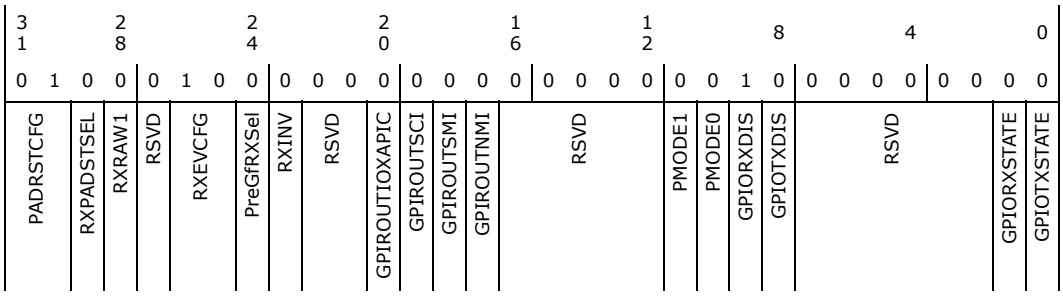
This register applies to GPP_A20.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000200h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	00h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.78 Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_20)—Offset 4A4h

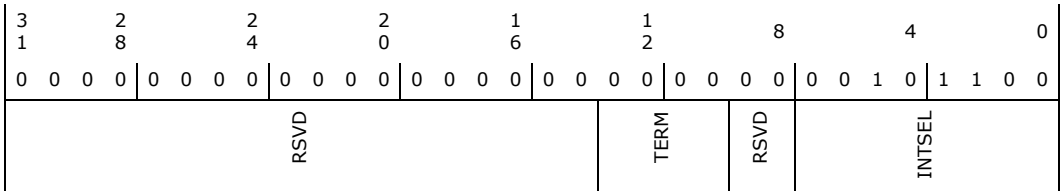
This register applies to GPP_A20.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 2Ch



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	2Ch RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.79 Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_21)—Offset 4A8h

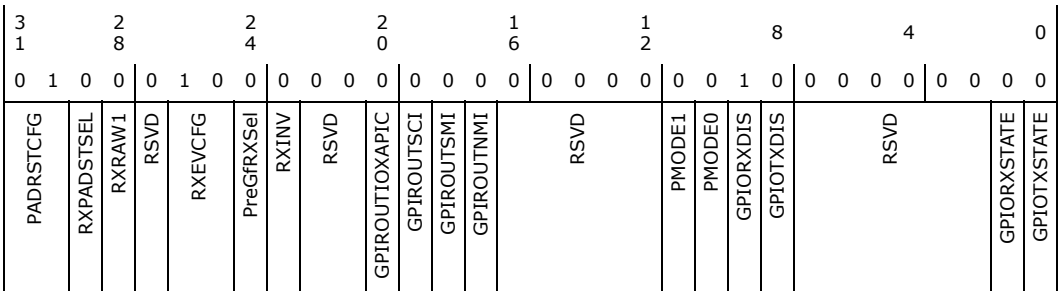
This register applies to GPP_A21.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000200h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.80 Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_21)—Offset 4ACh

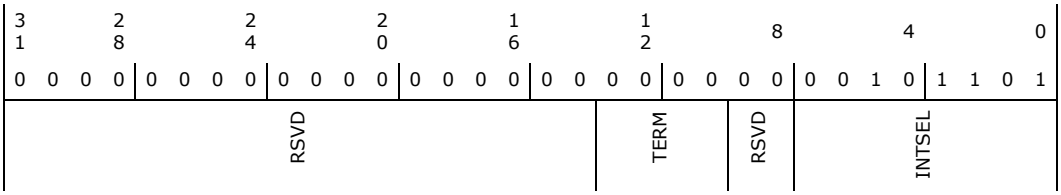
This register applies to GPP_A21.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 2Dh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	2Dh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.81 Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_22)—Offset 4B0h

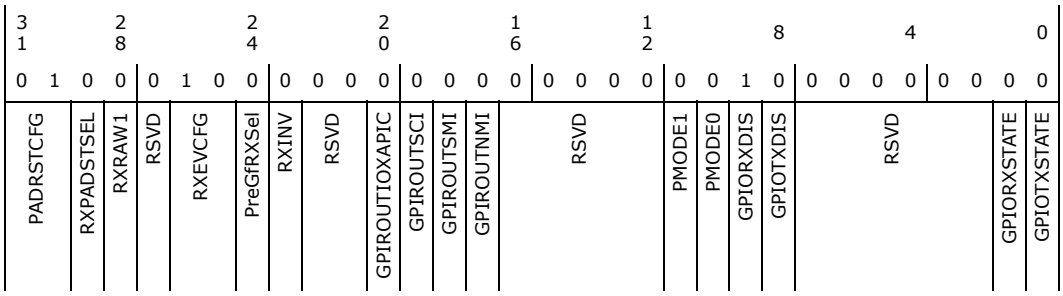
This register applies to GPP_A22.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000200h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ. 1 = Routing can cause peripheral IRQ. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.6.82 Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_22)—Offset 4B4h

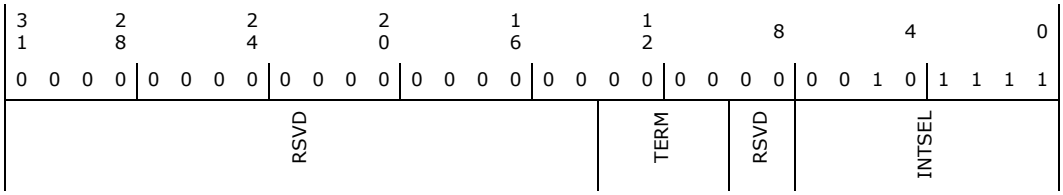
This register applies to GPP_A22.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 2Eh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	2Eh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.83 Pad Configuration DW0 (PAD_CFG_DW0_GPP_A_23)—Offset 4B8h

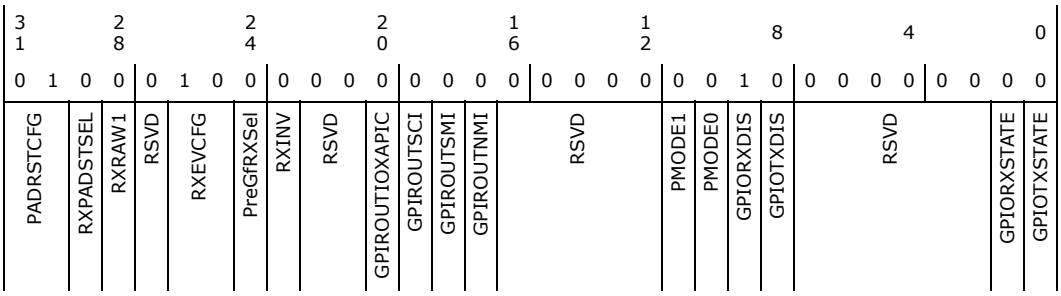
This register applies to GPP_A23.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000200h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.6.84 Pad Configuration DW1 (PAD_CFG_DW1_GPP_A_23)—Offset 4BCh

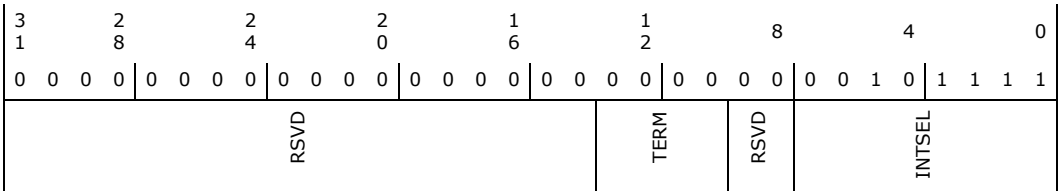
This register applies to GPP_A23.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 2Fh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	2Fh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.85 Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_0)—Offset 4C0h

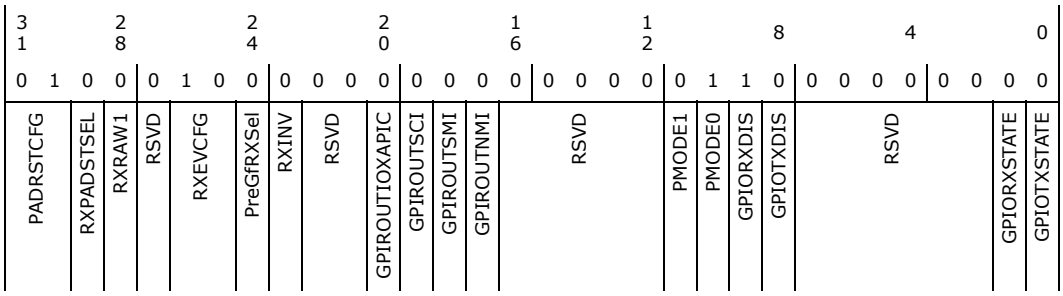
This register applies to GPP_B0.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000600h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1b RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.6.86 Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_0)—Offset 4C4h

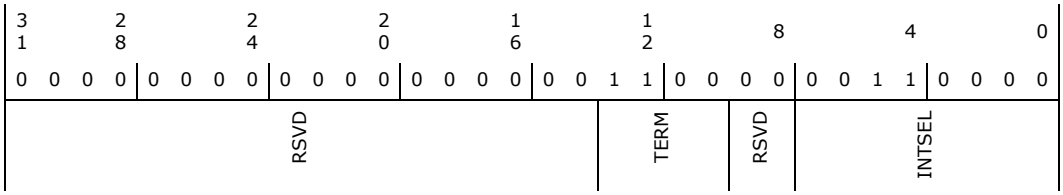
This register applies to GPP_B0.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003030h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	1100b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	30h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.6.87 Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_1)—Offset 4C8h

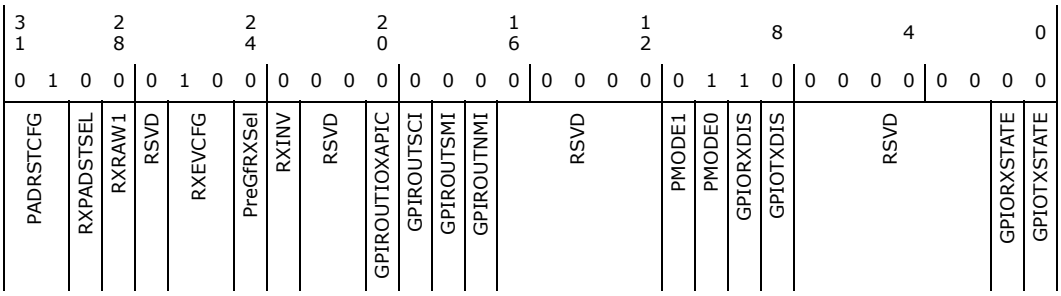
This register applies to GPP_B1.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000600h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.88 Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_1)—Offset 4CCh

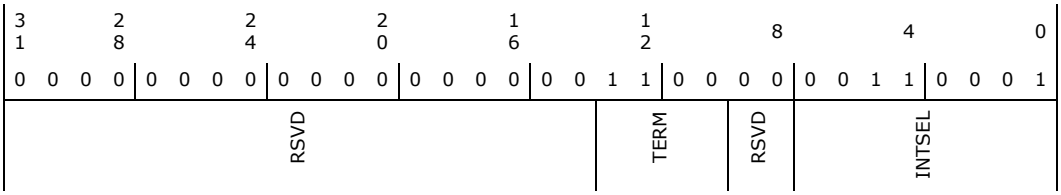
This register applies to GPP_B1.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003031h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	1100b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	31h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.89 Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_2)—Offset 4D0h

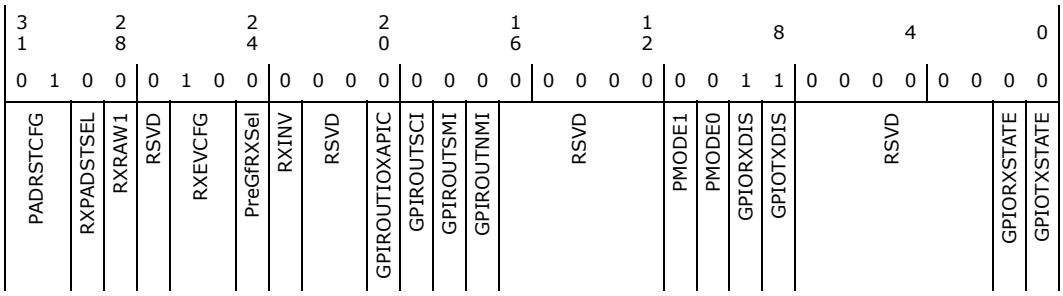
This register applies to GPP_B2.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.6.90 Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_2)—Offset 4D4h

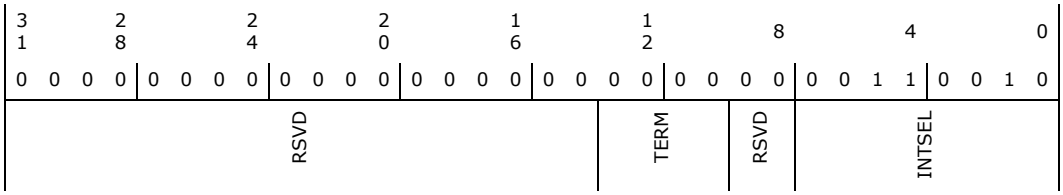
This register applies to GPP_B2.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 32h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	32h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.91 Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_3)—Offset 4D8h

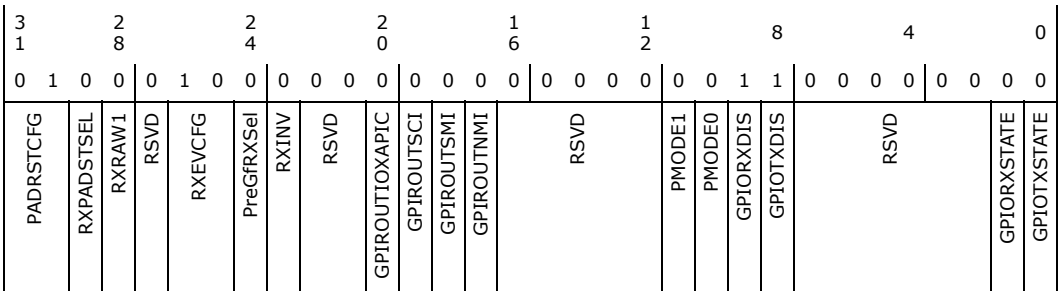
This register applies to GPP_B3.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ. 1 = Routing can cause peripheral IRQ. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.92 Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_3)—Offset 4DCh

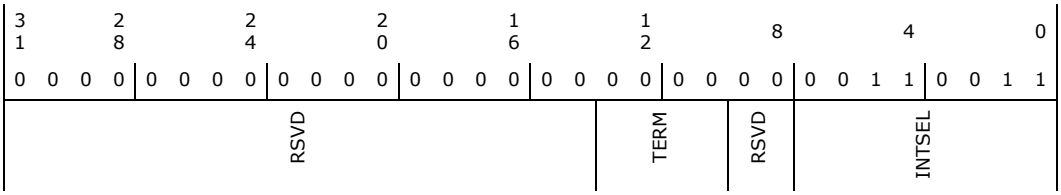
This register applies to GPP_B3.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 33h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	33h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.93 Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_4)—Offset 4E0h

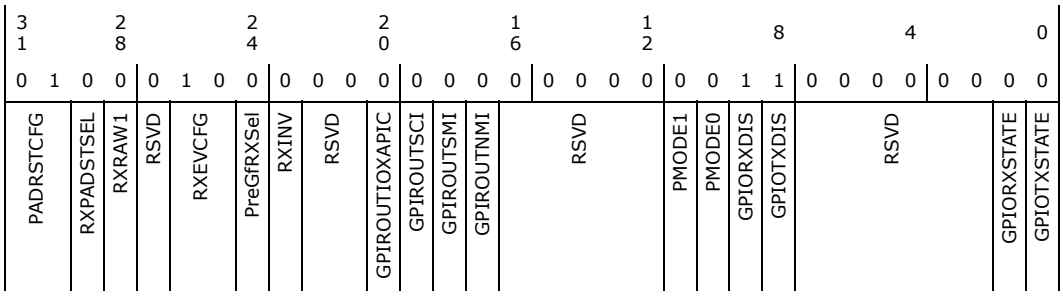
This register applies to GPP_B4.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.94 Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_4)—Offset 4E4h

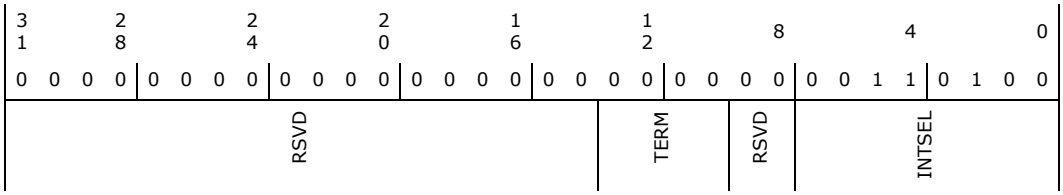
This register applies to GPP_B4.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 34h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	34h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.95 Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_5)—Offset 4E8h

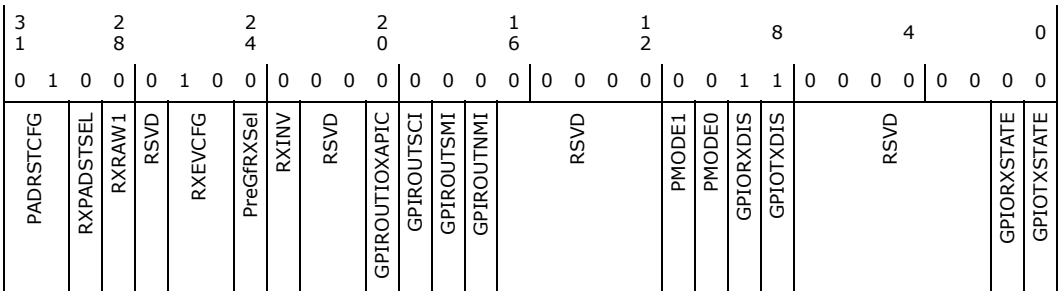
This register applies to GPP_B5.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.96 Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_5)—Offset 4ECh

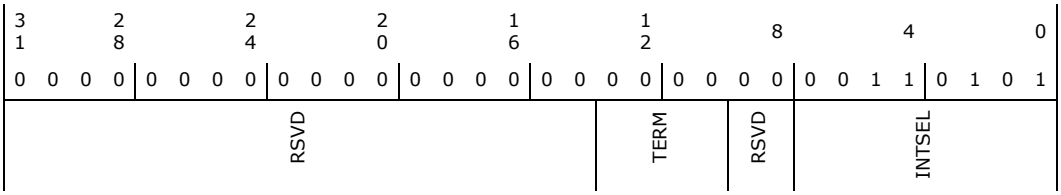
This register applies to GPP_B5.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 35h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	35h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.97 Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_6)—Offset 4F0h

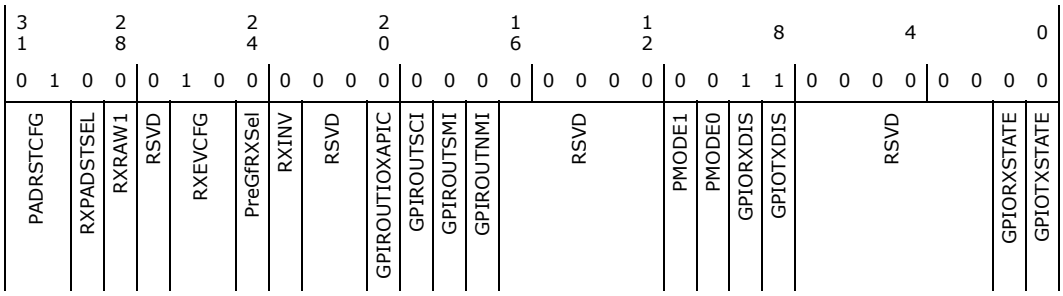
This register applies to GPP_B6.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.98 Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_6)—Offset 4F4h

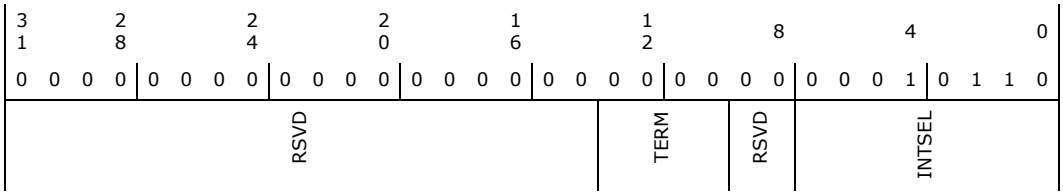
This register applies to GPP_B6.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 36h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	36h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.99 Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_7)—Offset 4F8h

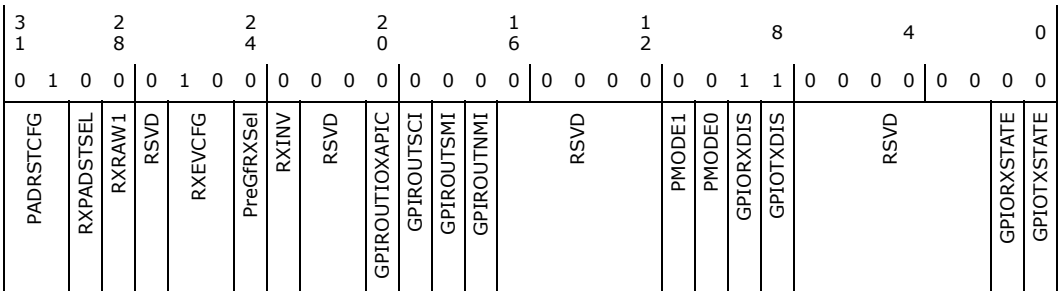
This register applies to GPP_B7.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.100 Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_7)—Offset 4FCh

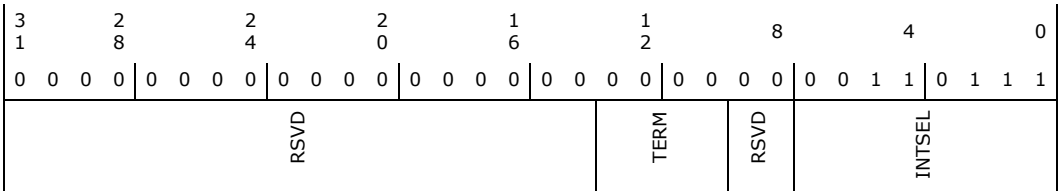
This register applies to GPP_B7.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 37h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	37h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.101 Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_8)—Offset 500h

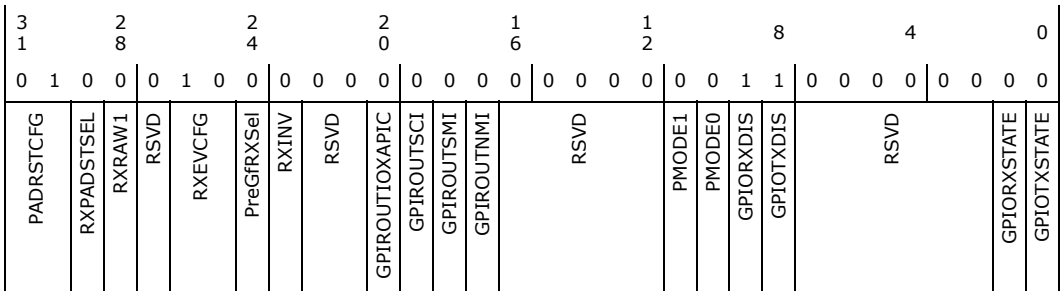
This register applies to GPP_B8.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.102 Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_8)—Offset 504h

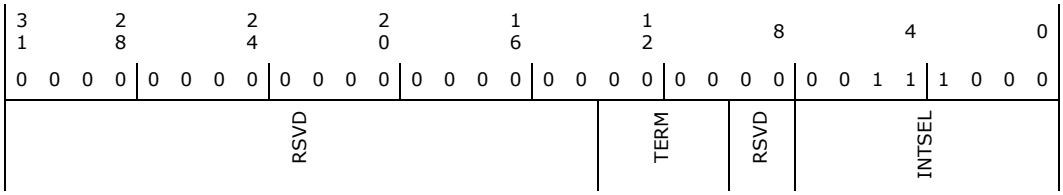
This register applies to GPP_B8.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 38h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	38h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.103 Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_9)—Offset 508h

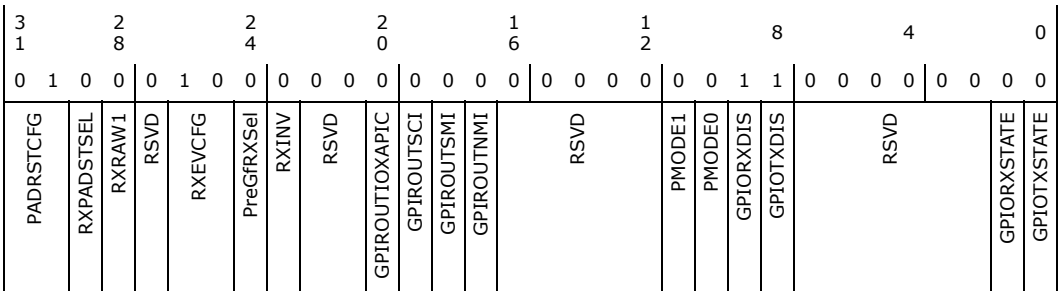
This register applies to GPP_B9.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.104 Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_9)—Offset 50Ch

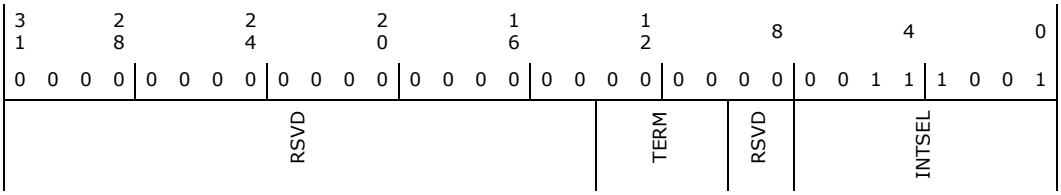
This register applies to GPP_B9.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 39h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	39h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.105 Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_10)—Offset 510h

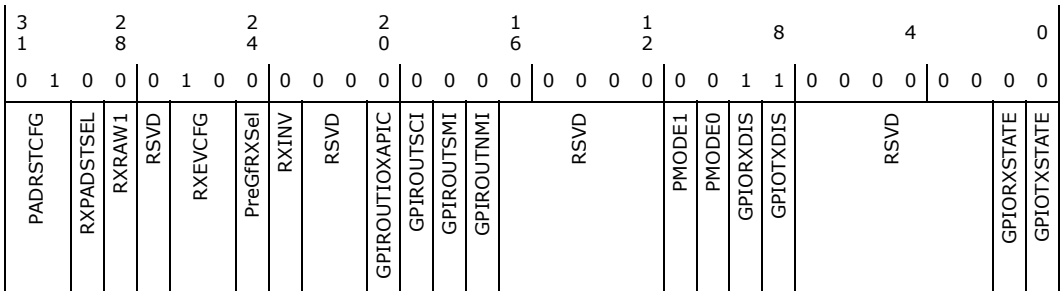
This register applies to GPP_B10.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.106 Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_10)—Offset 514h

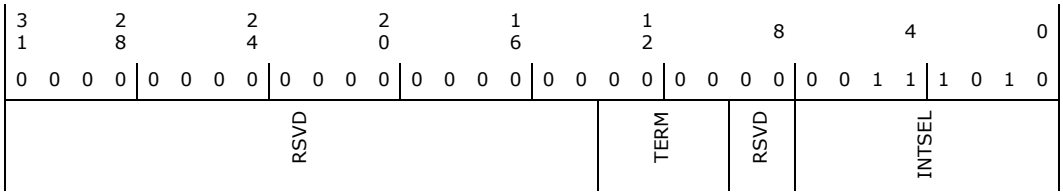
This register applies to GPP_B10.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 3Ah



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	3Ah RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.107 Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_11)—Offset 518h

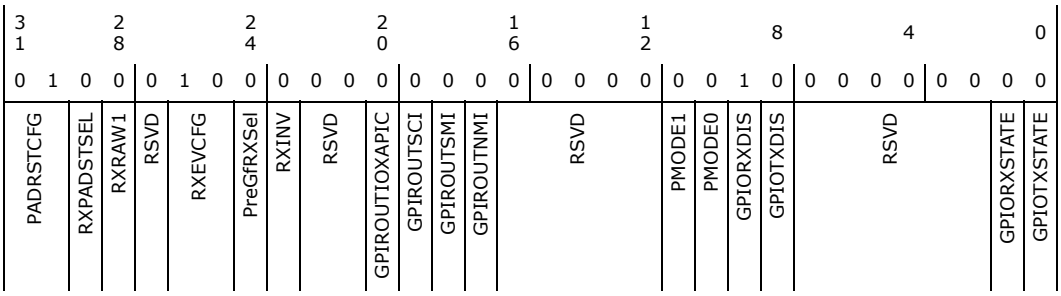
This register applies to GPP_B11.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000200h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.108 Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_11)—Offset 51Ch

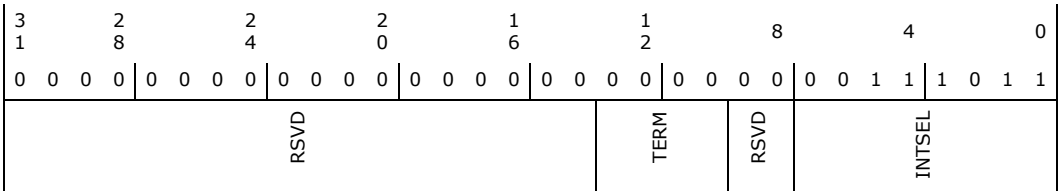
This register applies to GPP_B11.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 3Bh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	3Bh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.109 Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_12)—Offset 520h

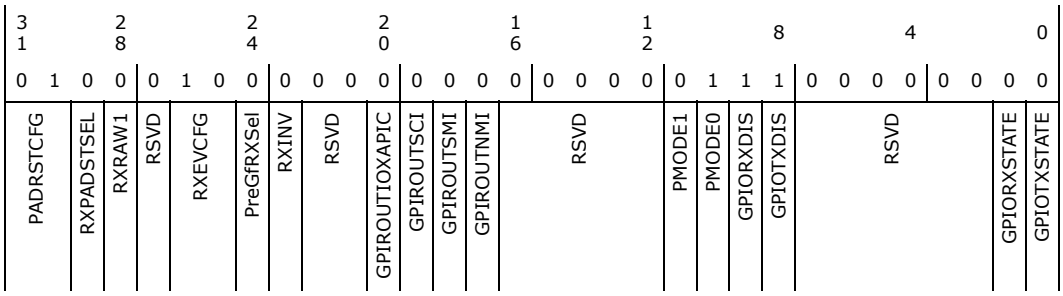
This register applies to GPP_B12.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.110 Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_12)—Offset 524h

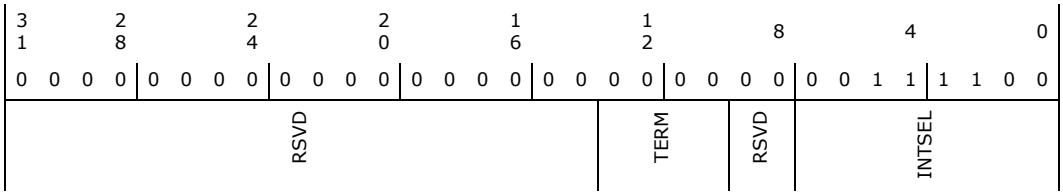
This register applies to GPP_B12.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 3Ch



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	3Ch RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.111 Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_13)—Offset 528h

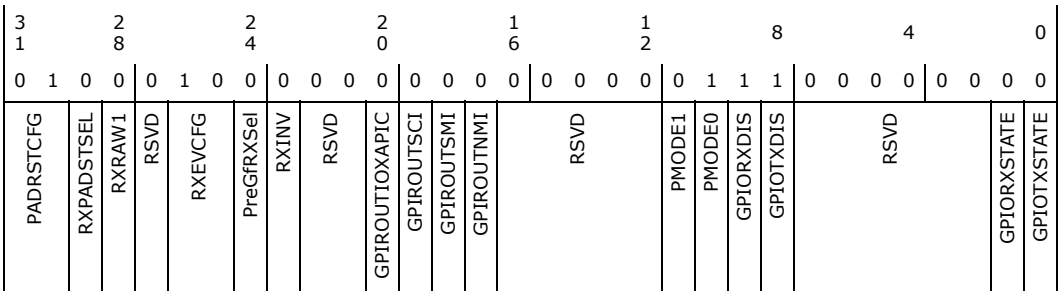
This register applies to GPP_B13.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.112 Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_13)—Offset 52Ch

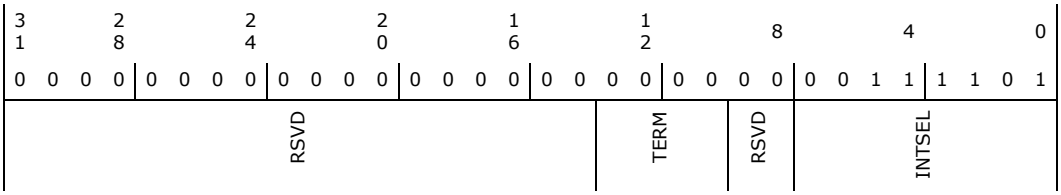
This register applies to GPP_B13.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 3Dh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	3Dh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.113 Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_14)—Offset 530h

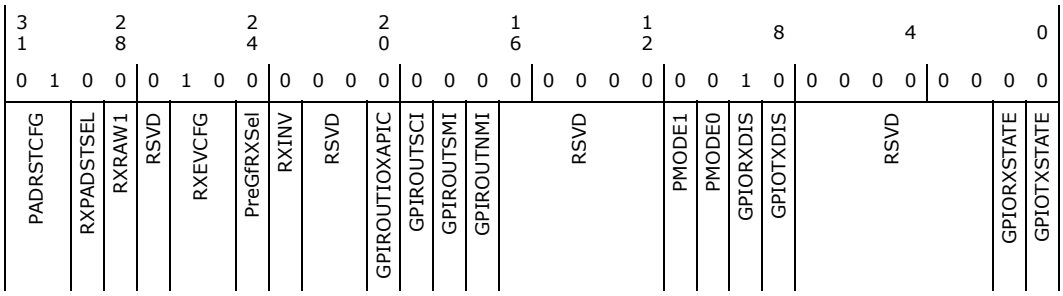
This register applies to GPP_B14.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000200h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.114 Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_14)—Offset 534h

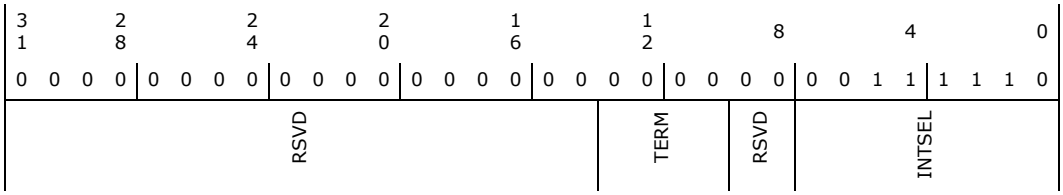
This register applies to GPP_B14.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 3Eh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	3Eh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.115 Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_15)—Offset 538h

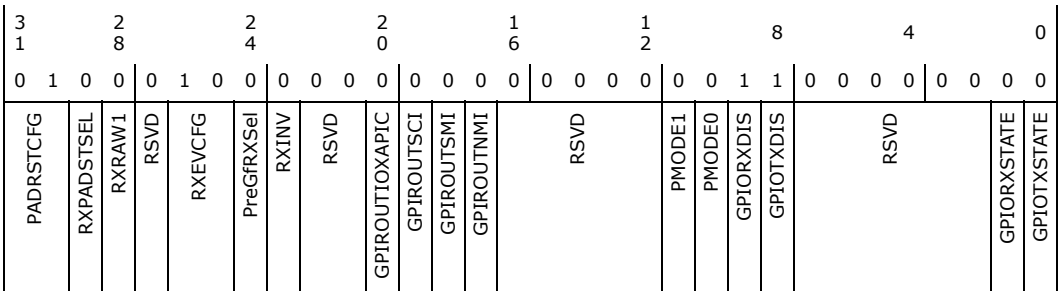
This register applies to GPP_B15.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.116 Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_15)—Offset 53Ch

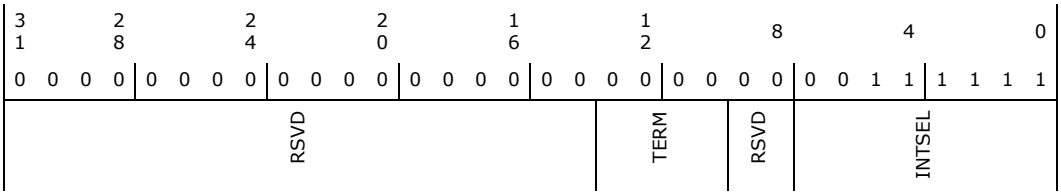
This register applies to GPP_B15.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 3Fh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	3Fh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.117 Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_16)—Offset 540h

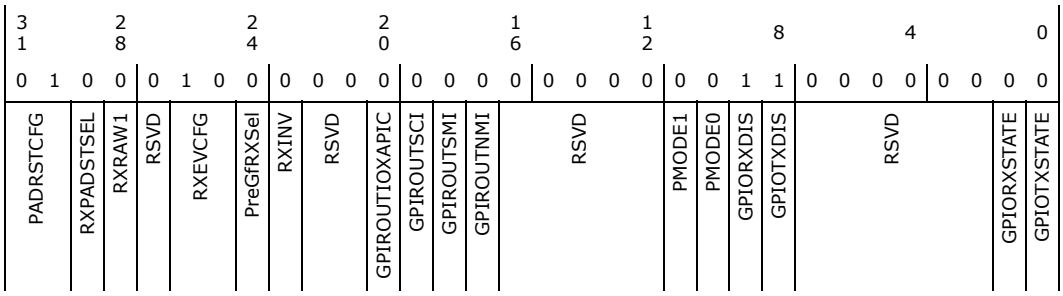
This register applies to GPP_B16.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.118 Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_16)—Offset 544h

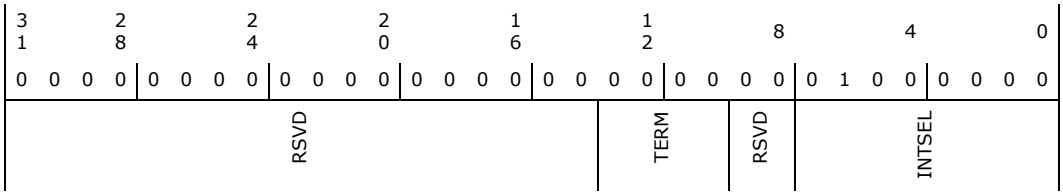
This register applies to GPP_B16.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 40h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	40h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.119 Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_17)—Offset 548h

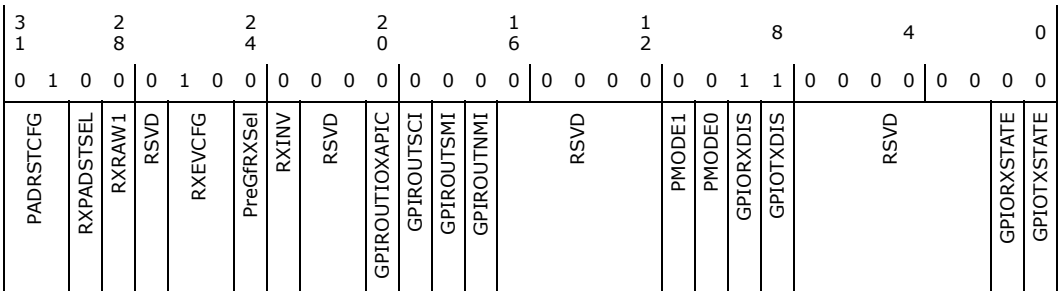
This register applies to GPP_B17.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.120 Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_17)—Offset 54Ch

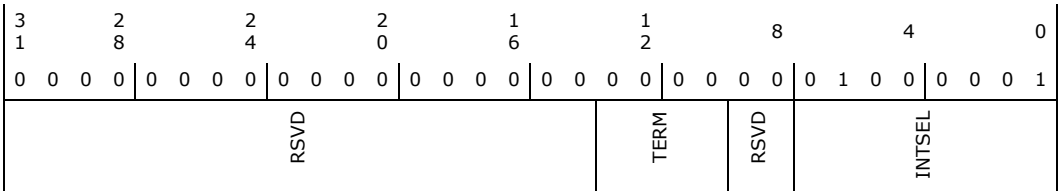
This register applies to GPP_B17.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 41h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	41h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.121 Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_18)—Offset 550h

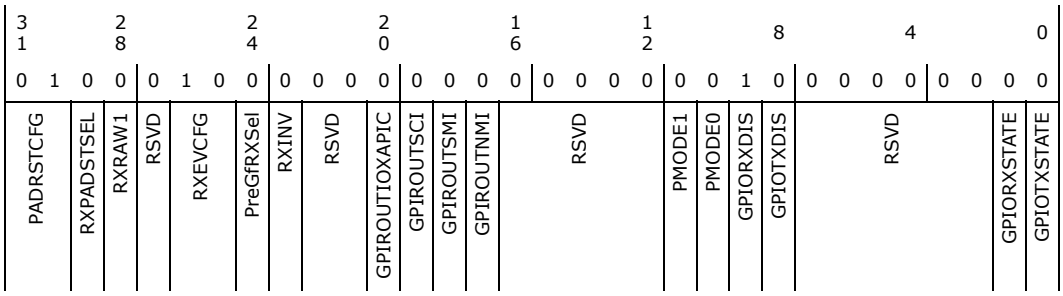
This register applies to GPP_B18.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000200h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.122 Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_18)—Offset 554h

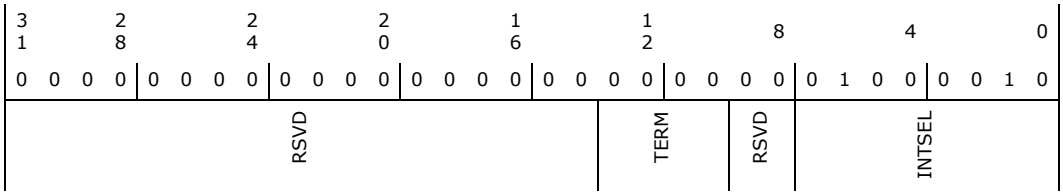
This register applies to GPP_B18.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 42h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	42h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.123 Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_19)—Offset 558h

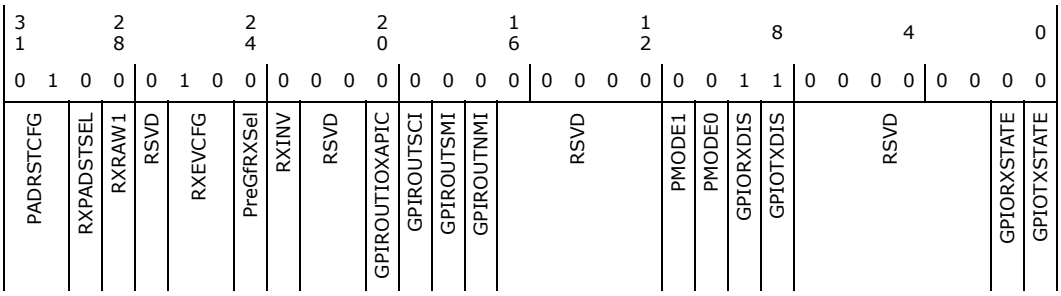
This register applies to GPP_B19.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.124 Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_19)—Offset 55Ch

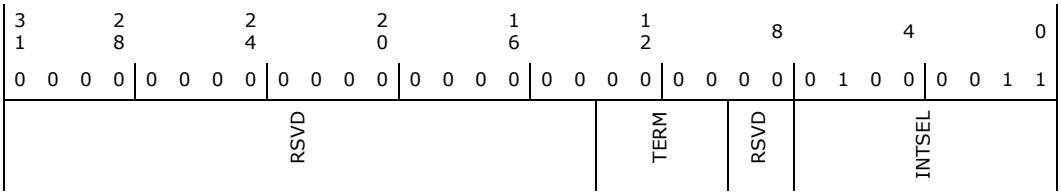
This register applies to GPP_B19.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 43h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	43h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.125 Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_20)—Offset 560h

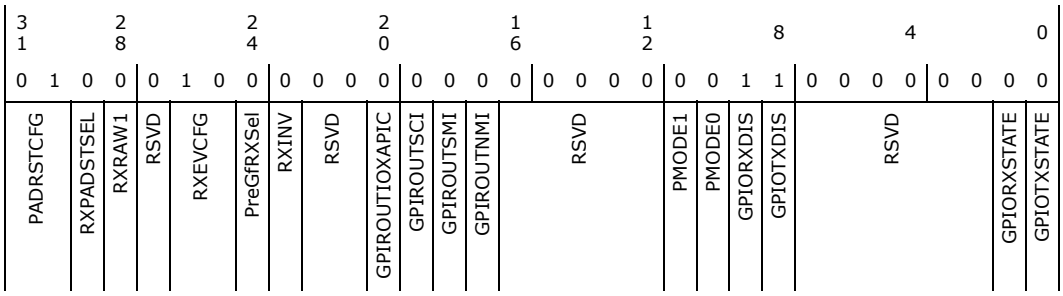
This register applies to GPP_B20.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.126 Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_20)—Offset 564h

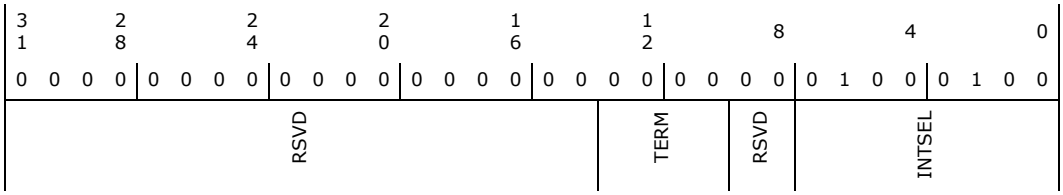
This register applies to GPP_B20.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	44h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.127 Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_21)—Offset 568h

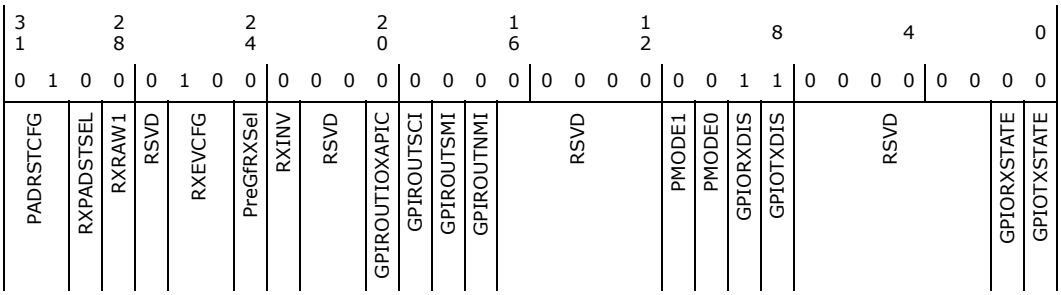
This register applies to GPP_B21.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.128 Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_21)—Offset 56Ch

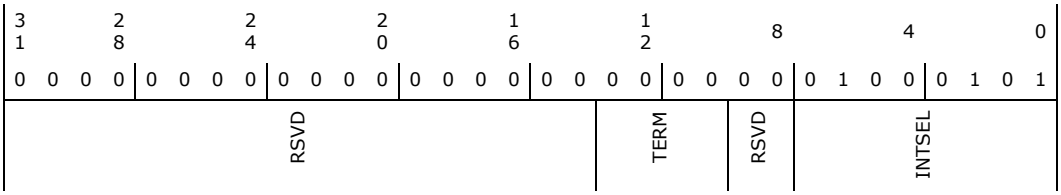
This register applies to GPP_B21.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 45h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	45h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.129 Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_22)—Offset 570h

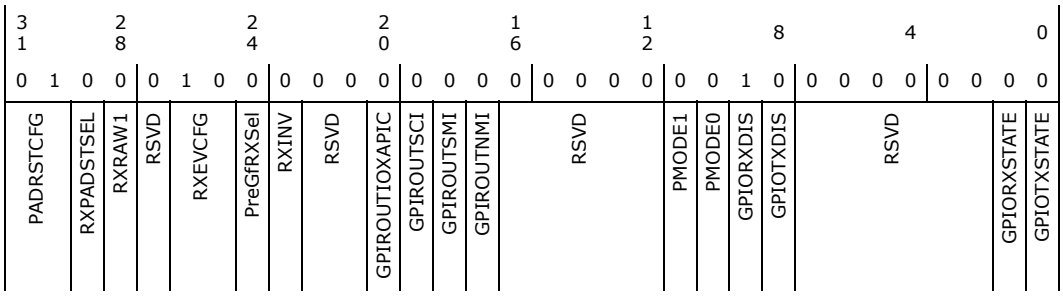
This register applies to GPP_B22.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000200h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.130 Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_22)—Offset 574h

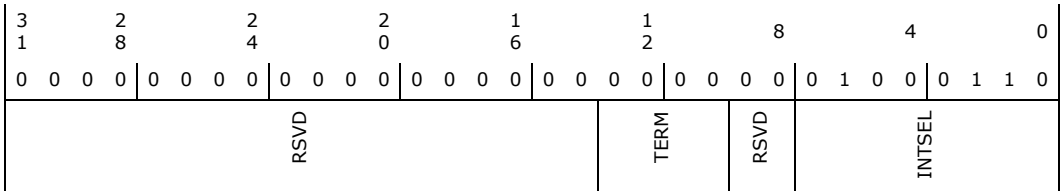
This register applies to GPP_B22.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 46h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	46h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.131 Pad Configuration DW0 (PAD_CFG_DW0_GPP_B_23)—Offset 578h

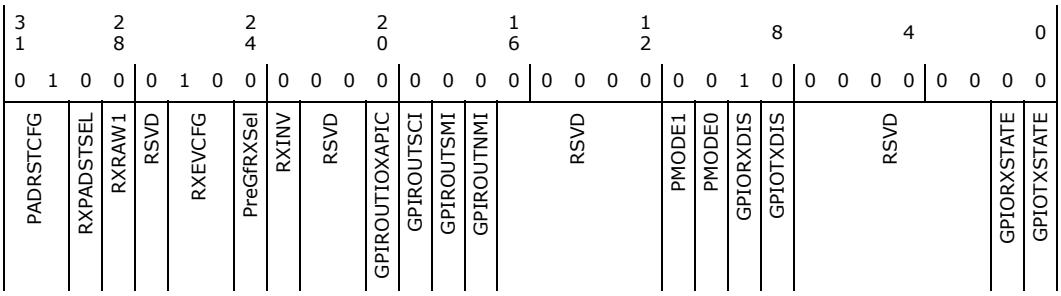
This register applies to GPP_B23.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000200h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.132 Pad Configuration DW1 (PAD_CFG_DW1_GPP_B_23)—Offset 57Ch

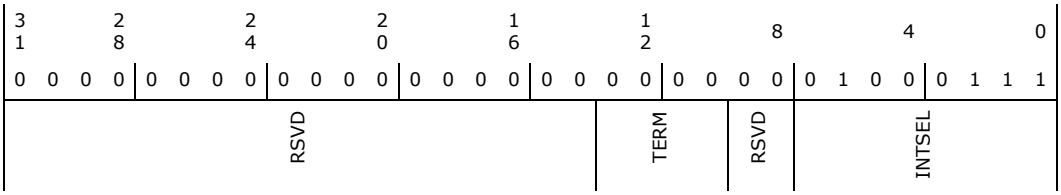
This register applies to GPP_B23.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 47h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	47h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.133 Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_0)—Offset 580h

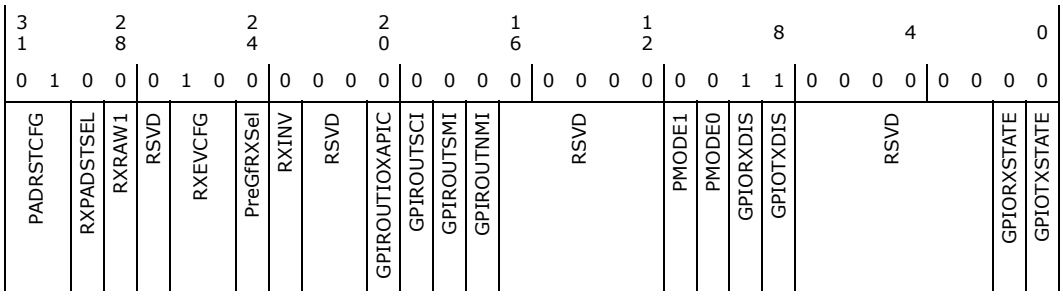
This register applies to GPP_F0.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.134 Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_0)—Offset 584h

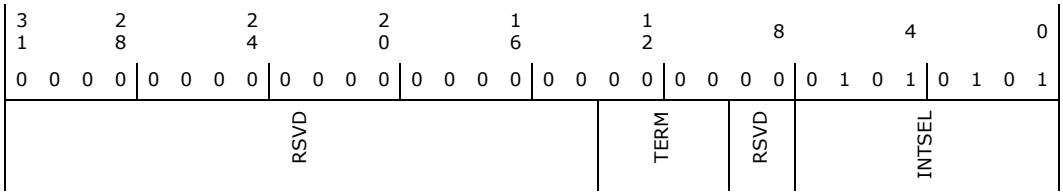
This register applies to GPP_F0.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 55h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	55h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.135 Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_1)—Offset 588h

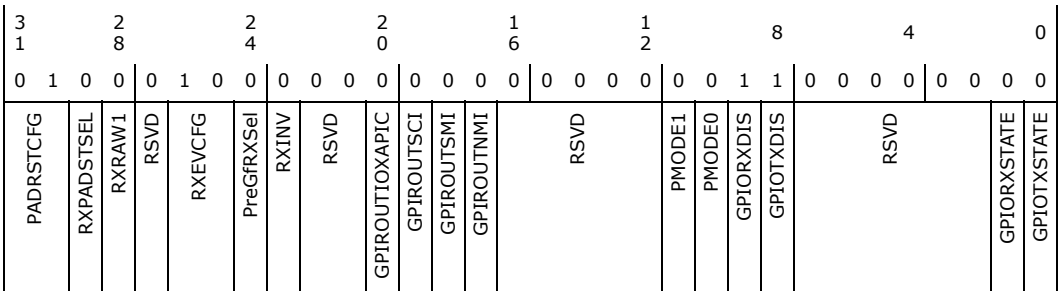
This register applies to GPP_F1.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.136 Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_1)—Offset 58Ch

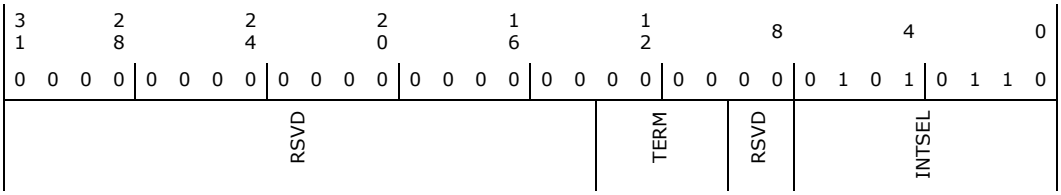
This register applies to GPP_F1.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 56h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	56h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.137 Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_2)—Offset 590h

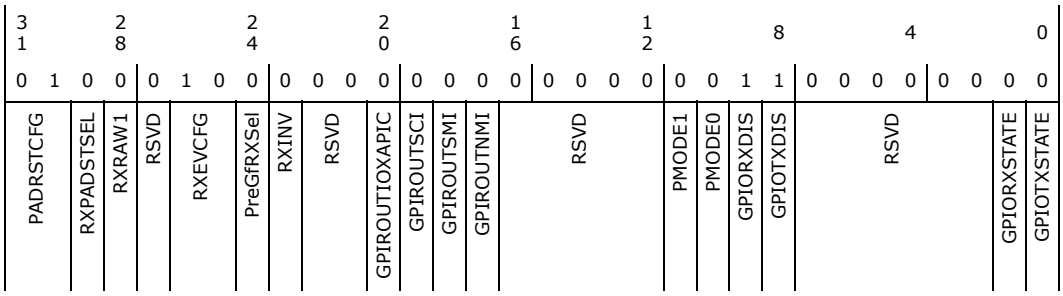
This register applies to GPP_F2.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.138 Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_2)—Offset 594h

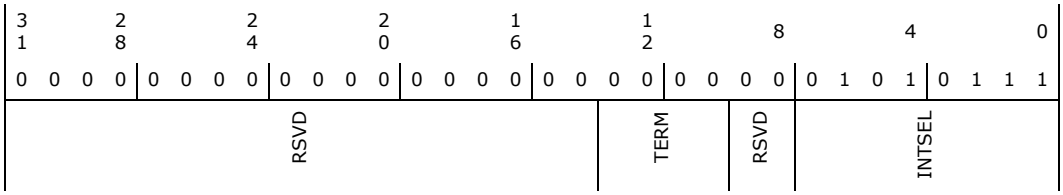
This register applies to GPP_F2.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 57h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	57h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.139 Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_3)—Offset 598h

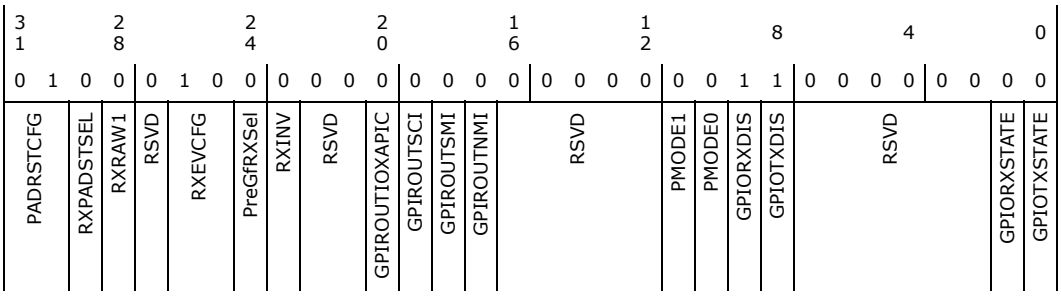
This register applies to GPP_F3.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.140 Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_3)—Offset 59Ch

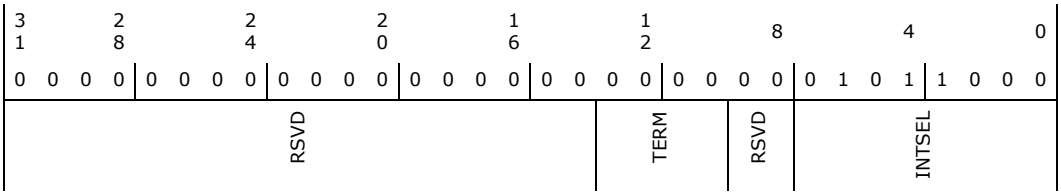
This register applies to GPP_F3.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 58h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	58h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.141 Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_4)—Offset 5A0h

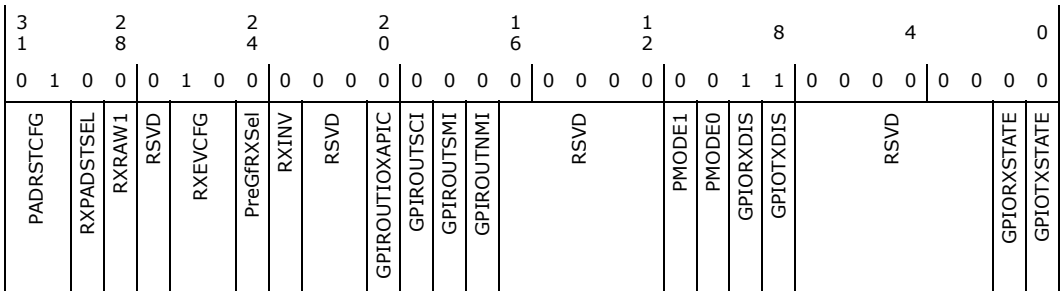
This register applies to GPP_F4.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.142 Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_4)—Offset 5A4h

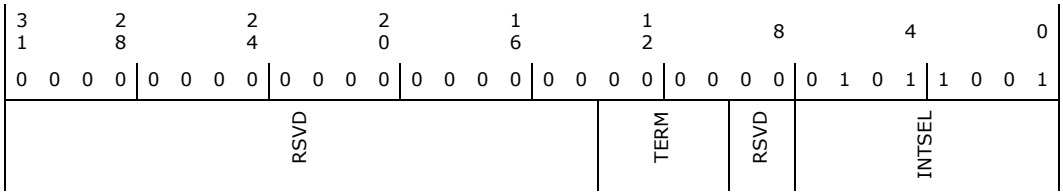
This register applies to GPP_F4.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 59h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	59h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.143 Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_5)—Offset 5A8h

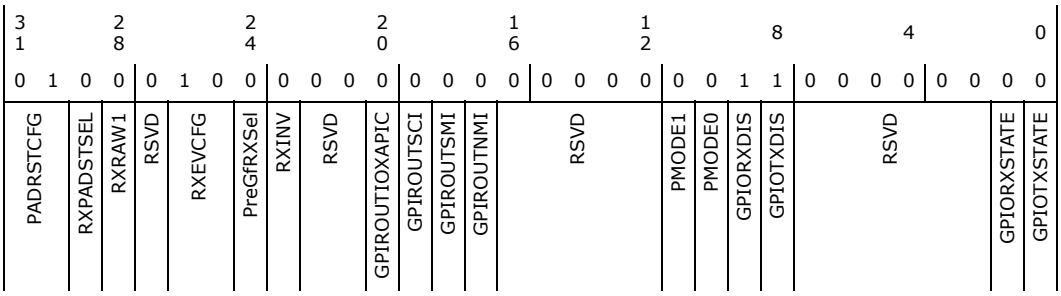
This register applies to GPP_F5.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.144 Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_5)—Offset 5ACh

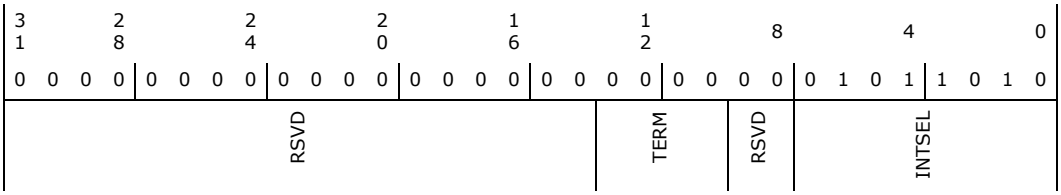
This register applies to GPP_F5.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 5Ah



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	5Ah RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.145 Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_6)—Offset 5B0h

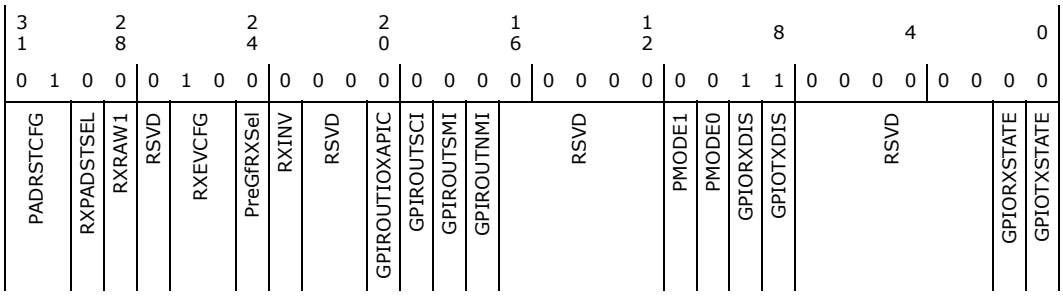
This register applies to GPP_F6.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.146 Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_6)—Offset 5B4h

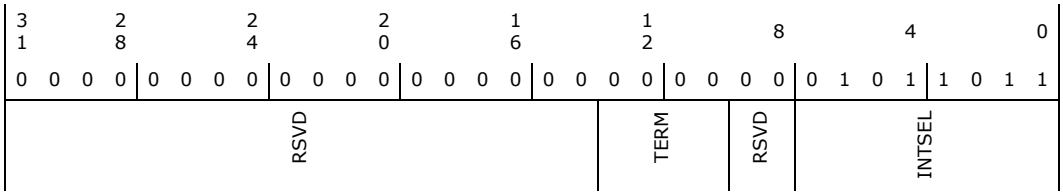
This register applies to GPP_F6.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 5Bh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	5Bh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.147 Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_7)—Offset 5B8h

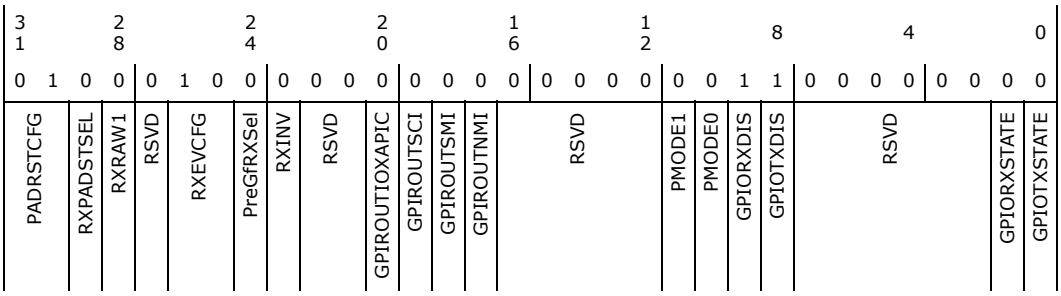
This register applies to GPP_F7.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.148 Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_7)—Offset 5BCh

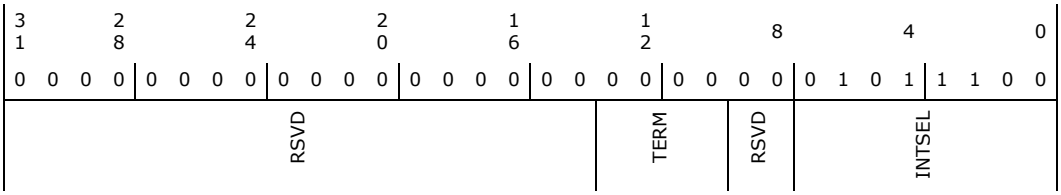
This register applies to GPP_F7.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 5Ch



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	5Ch RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.149 Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_8)—Offset 5C0h

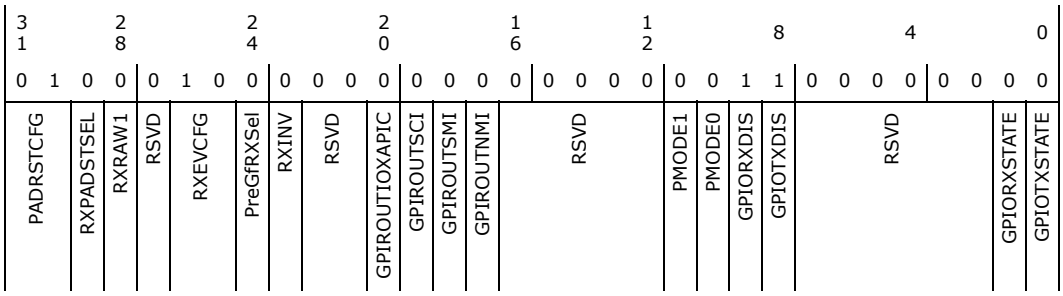
This register applies to GPP_F8.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.150 Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_8)—Offset 5C4h

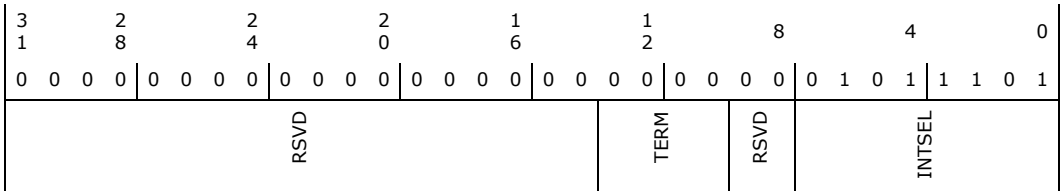
This register applies to GPP_F8.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 5Dh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	5Dh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.151 Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_9)—Offset 5C8h

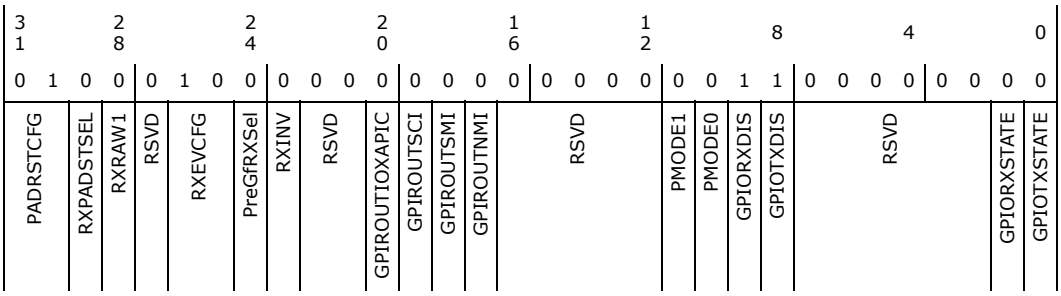
This register applies to GPP_F9.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.152 Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_9)—Offset 5CCh

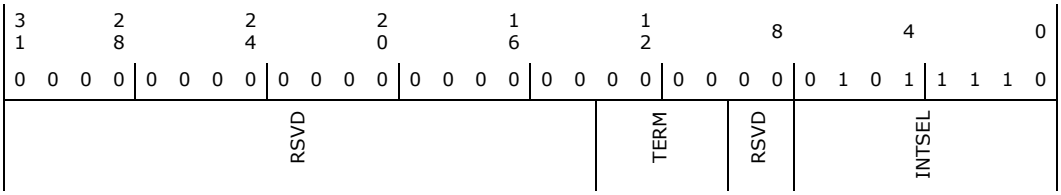
This register applies to GPP_F9.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 5Eh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	5Eh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.153 Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_10)—Offset 5D0h

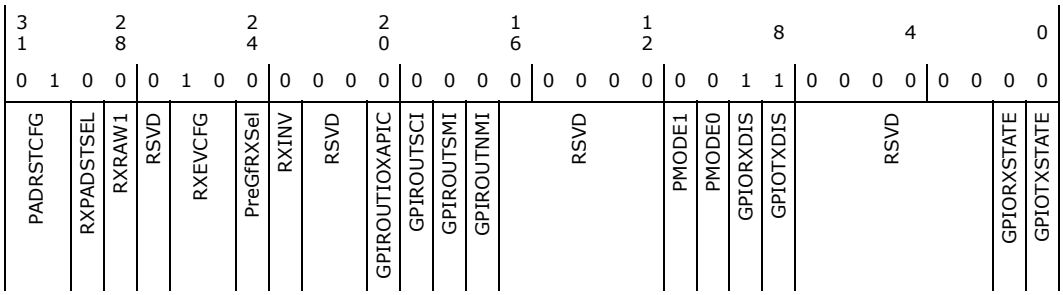
This register applies to GPP_F10.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.154 Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_10)—Offset 5D4h

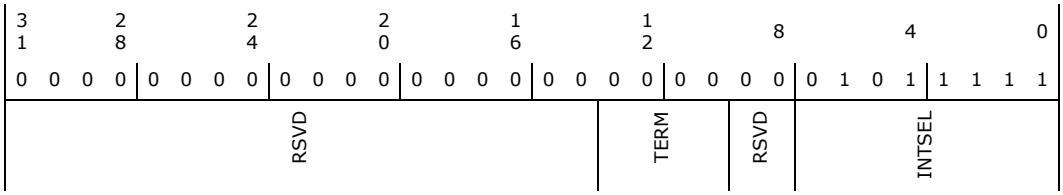
This register applies to GPP_F10.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 5Fh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	5Fh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.155 Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_11)—Offset 5D8h

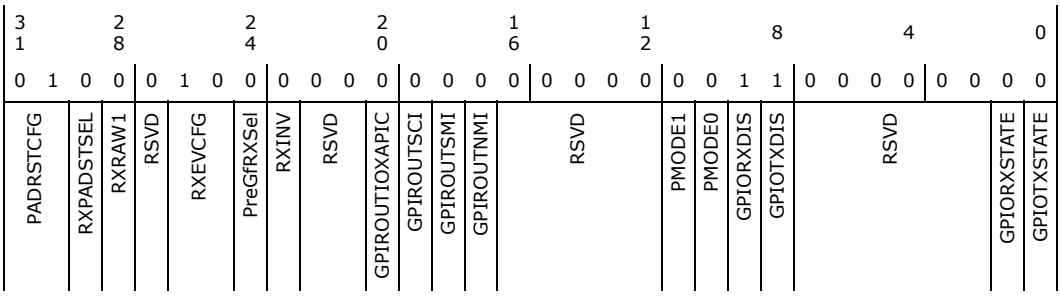
This register applies to GPP_F11.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.156 Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_11)—Offset 5DCh

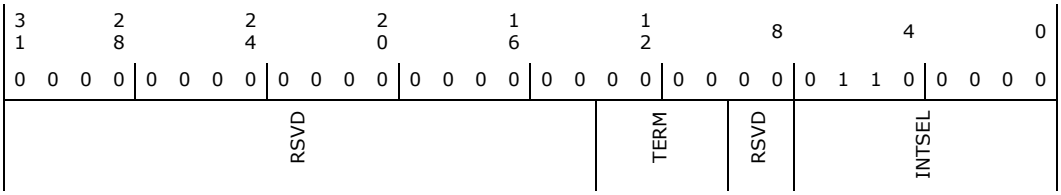
This register applies to GPP_F11.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 60h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	60h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.157 Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_12)—Offset 5E0h

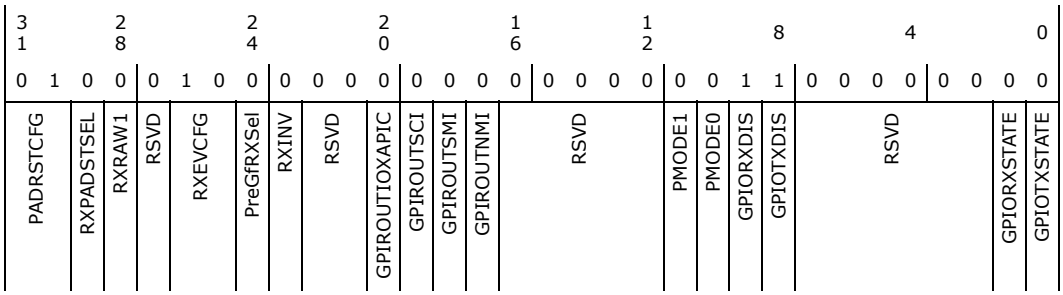
This register applies to GPP_F12.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.158 Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_12)—Offset 5E4h

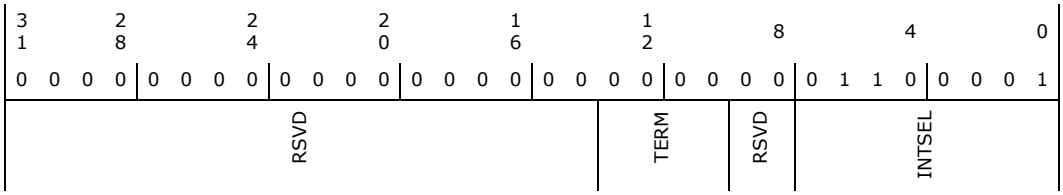
This register applies to GPP_F12.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 61h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	61h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.159 Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_13)—Offset 5E8h

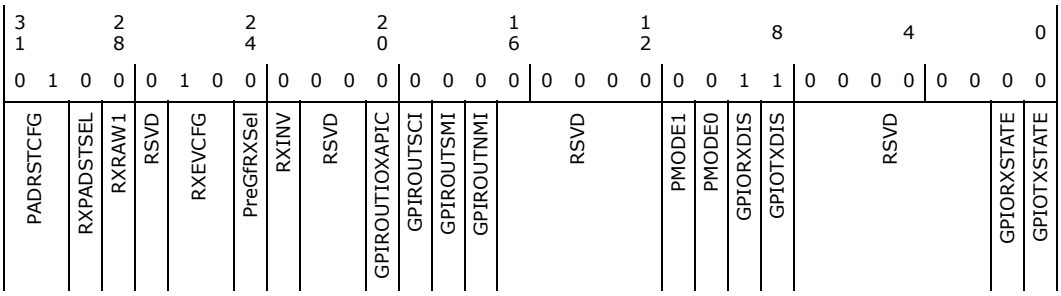
This register applies to GPP_F13.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.160 Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_13)—Offset 5ECh

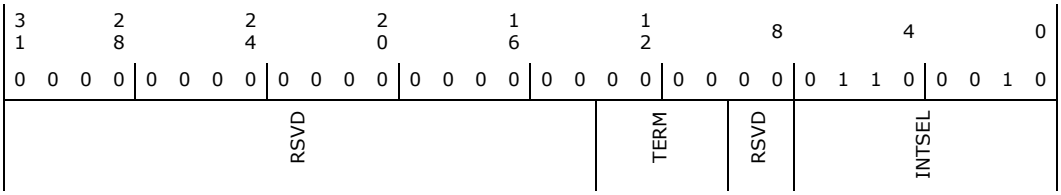
This register applies to GPP_F13.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 62h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	62h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.161 Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_14)—Offset 5F0h

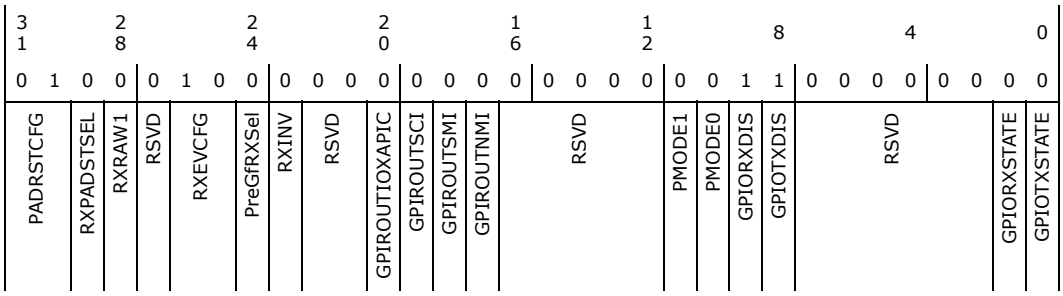
This register applies to GPP_F14.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.162 Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_14)—Offset 5F4h

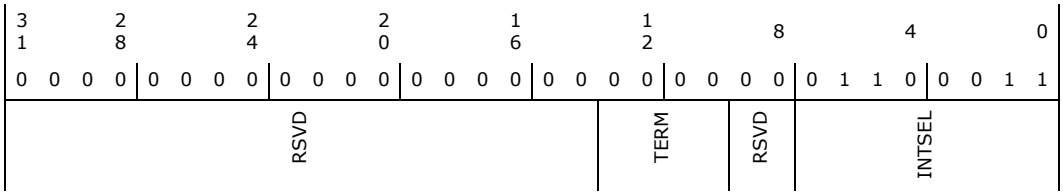
This register applies to GPP_F14.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 63h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	63h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.163 Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_15)—Offset 5F8h

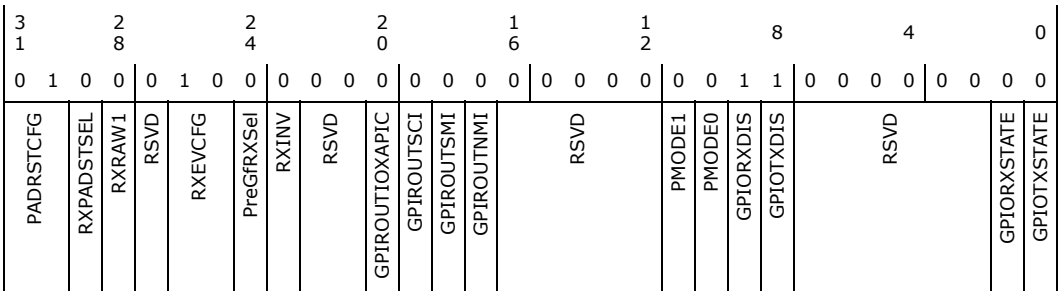
This register applies to GPP_F15.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.164 Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_15)—Offset 5FCh

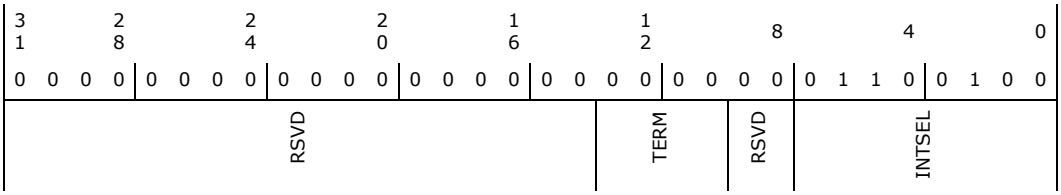
This register applies to GPP_F15.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 64h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	64h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.165 Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_16)—Offset 600h

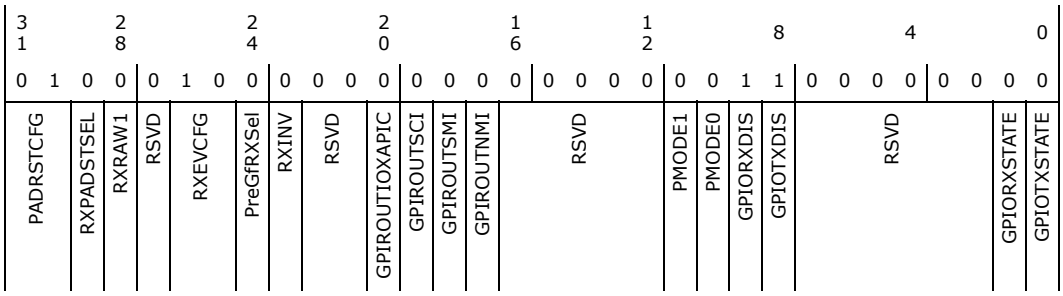
This register applies to GPP_F16.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.166 Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_16)—Offset 604h

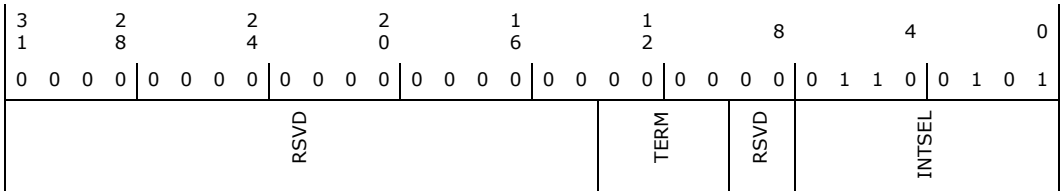
This register applies to GPP_F16.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 65h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	65h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.167 Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_17)—Offset 608h

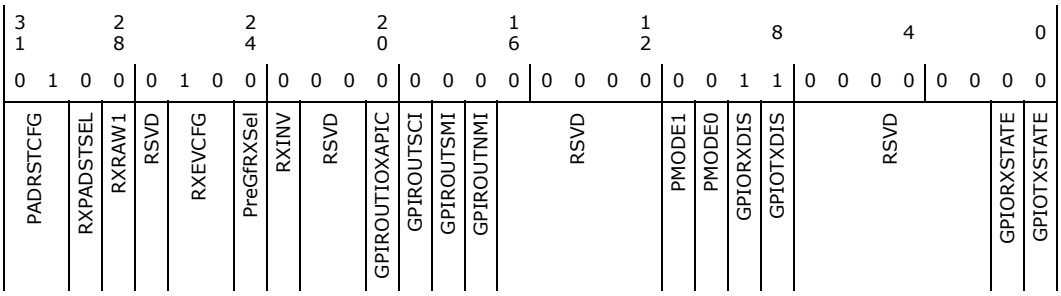
This register applies to GPP_F17.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.168 Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_17)—Offset 60Ch

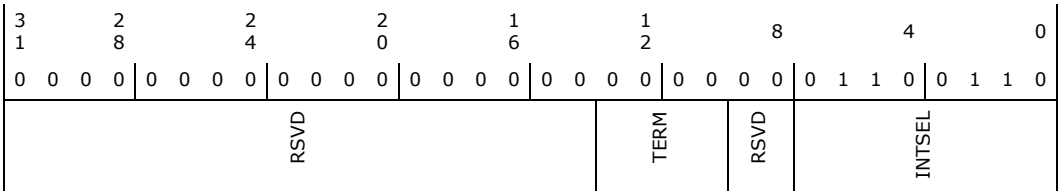
This register applies to GPP_F17.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 66h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	66h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.169 Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_18)—Offset 610h

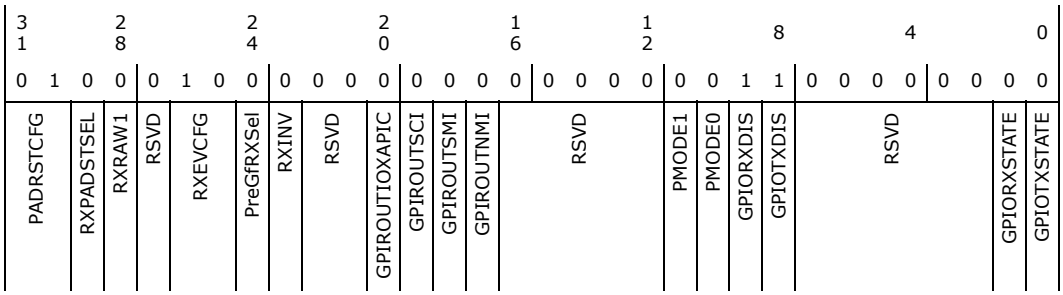
This register applies to GPP_F18.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.170 Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_18)—Offset 614h

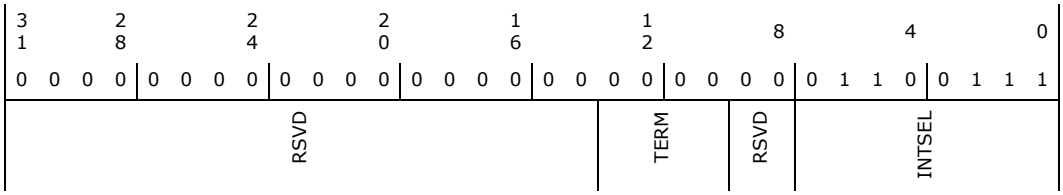
This register applies to GPP_F18.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 67h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	67h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.171 Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_19)—Offset 618h

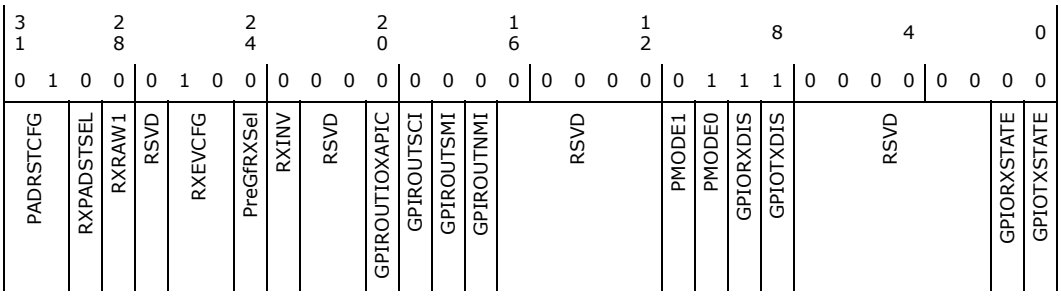
This register applies to GPP_F19.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.172 Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_19)—Offset 61Ch

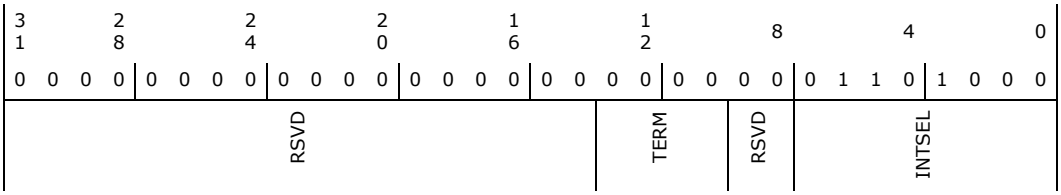
This register applies to GPP_F19.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 68h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	68h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.173 Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_20)—Offset 620h

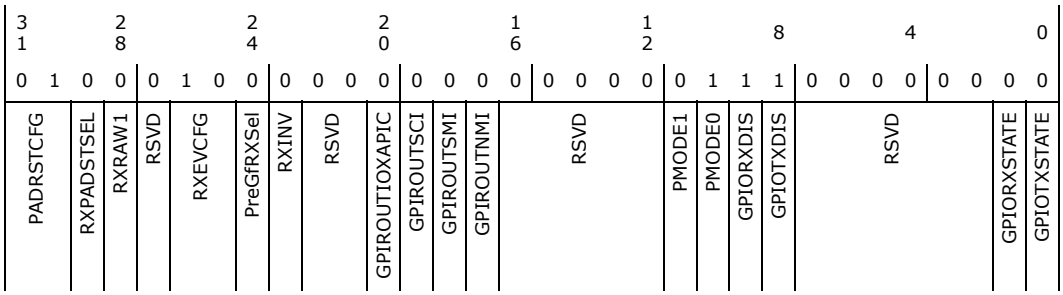
This register applies to GPP_F20.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.174 Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_20)—Offset 624h

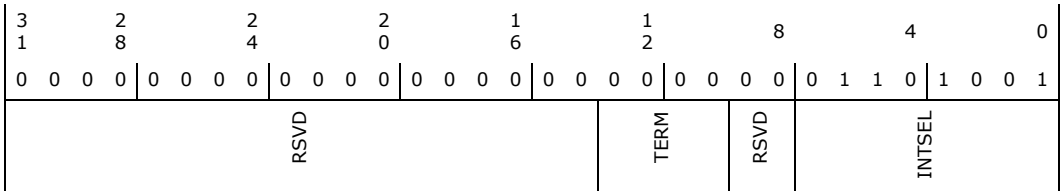
This register applies to GPP_F20.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 69h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	69h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.175 Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_21)—Offset 628h

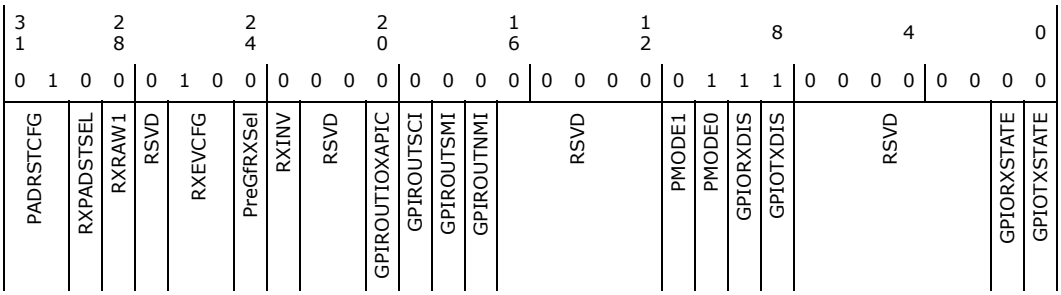
This register applies to GPP_F21.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.176 Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_21)—Offset 62Ch

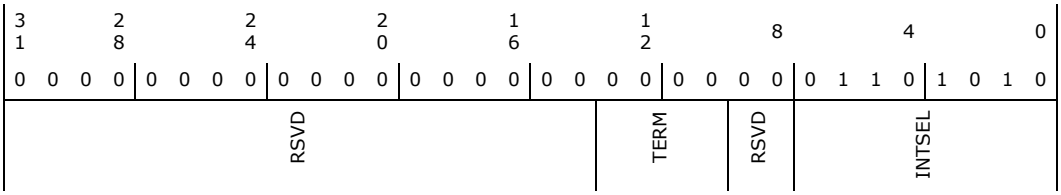
This register applies to GPP_F21.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 6Ah



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	6Ah RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.177 Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_22)—Offset 630h

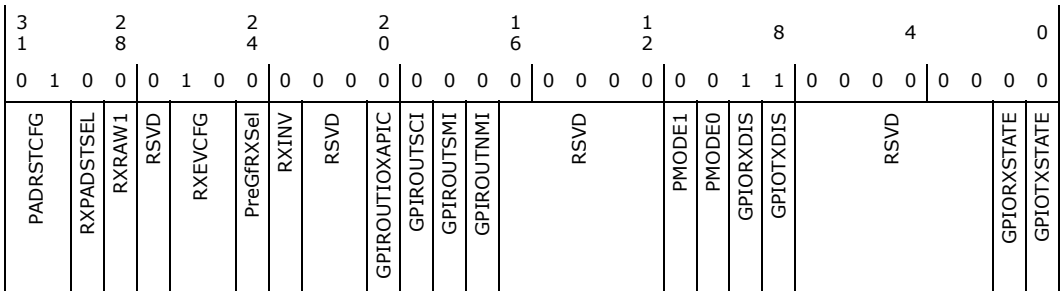
This register applies to GPP_F22.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ.
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.178 Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_22)—Offset 634h

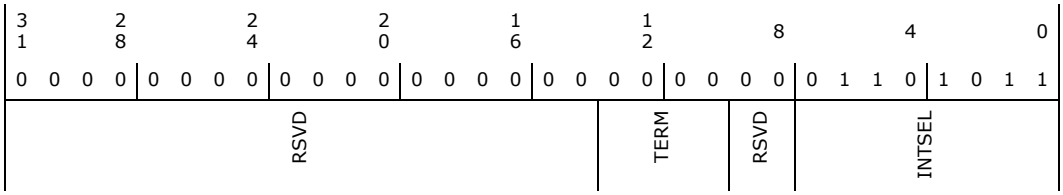
This register applies to GPP_F22 and has the same description as PAD_CFG_DW1_GPP_A_0. Exception: The default value of the INTSEL bit field in this register is 6Bh.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 6Bh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	6Bh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.6.179 Pad Configuration DW0 (PAD_CFG_DW0_GPP_F_23)—Offset 638h

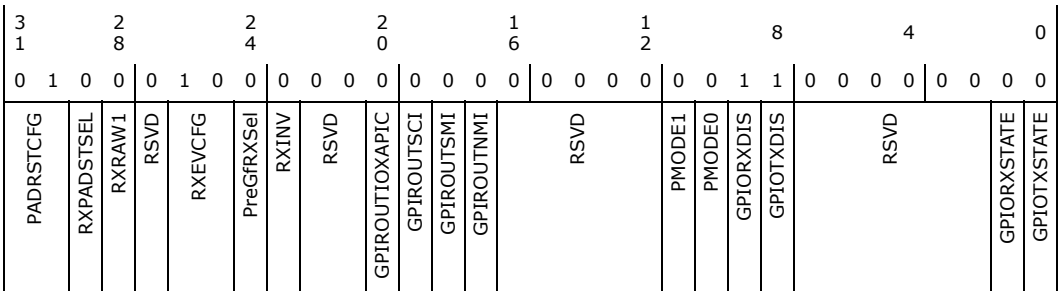
This register applies to GPP_F23.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.6.180 Pad Configuration DW1 (PAD_CFG_DW1_GPP_F_23)—Offset 63Ch

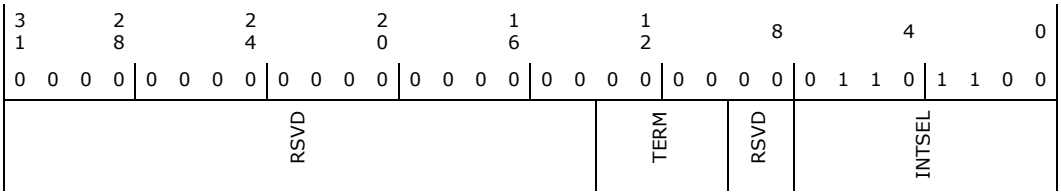
This register applies to GPP_F23.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 6Ch



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	6Ch RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max. IOxAPIC IRQ supported

18.7 GPIO Community 1 Registers Summary

Community 1 Registers are for GPP_C, GPP_D, and GPP_E groups.

Table 18-4. Summary of GPIO Community 1 Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
Ch	Fh	Pad Base Address (PADBAR)—Offset Ch	00000400h
10h	13h	Miscellaneous Configuration (MISCCFG)—Offset 10h	00043200h
20h	23h	Pad Ownership (PAD_OWN_GPP_C_0)—Offset 20h	00000000h
24h	27h	Pad Ownership (PAD_OWN_GPP_C_1)—Offset 24h	00000000h
28h	2Bh	Pad Ownership (PAD_OWN_GPP_C_2)—Offset 28h	00000000h
2Ch	2Fh	Pad Ownership (PAD_OWN_GPP_D_0)—Offset 2Ch	00000000h
30h	33h	Pad Ownership (PAD_OWN_GPP_D_1)—Offset 30h	00000000h
34h	37h	Pad Ownership (PAD_OWN_GPP_D_2)—Offset 34h	00000000h
38h	3Bh	Pad Ownership (PAD_OWN_GPP_E_0)—Offset 38h	00000000h
3Ch	3Fh	Pad Ownership (PAD_OWN_GPP_E_1)—Offset 3Ch	00000000h
60h	63	Pad Configuration Lock (PADCFGLOCK_GPP_C_0)—Offset 60h	00000000h
64h	67h	Pad Configuration Lock (PADCFGLOCKTX_GPP_C_0)—Offset 64h	00000000h
68h	6Bh	Pad Configuration Lock (PADCFGLOCK_GPP_D_0)—Offset 68h	00000000h
6Ch	6Fh	Pad Configuration Lock (PADCFGLOCKTX_GPP_D_0)—Offset 6Ch	00000000h
70h	73h	Pad Configuration Lock (PADCFGLOCK_GPP_E_0)—Offset 70h	00000000h
74h	77h	Pad Configuration Lock (PADCFGLOCKTX_GPP_E_0)—Offset 74h	00000000h
80h	83h	Host Software Pad Ownership (HOSTSW_OWN_GPP_C_0)—Offset 80h	00000000h
84h	87h	Host Software Pad Ownership (HOSTSW_OWN_GPP_D_0)—Offset 84h	00000000h
88h	8Bh	Host Software Pad Ownership (HOSTSW_OWN_GPP_E_0)—Offset 88h	00000000h
100h	103h	GPI Interrupt Status (GPI_IS_GPP_C_0)—Offset 100h	00000000h
104h	107h	GPI Interrupt Status (GPI_IS_GPP_D_0)—Offset 104h	00000000h
108h	10Bh	GPI Interrupt Status (GPI_IS_GPP_E_0)—Offset 108h	00000000h
110h	113h	GPI Interrupt Enable (GPI_IE_GPP_C_0)—Offset 110h	00000000h
114h	117h	GPI Interrupt Enable (GPI_IE_GPP_D_0)—Offset 114h	00000000h
118h	11Bh	GPI Interrupt Enable (GPI_IE_GPP_E_0)—Offset 118h	00000000h
120h	123h	GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_0)—Offset 120h	00000000h
124h	127h	GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_0)—Offset 124h	00000000h
128h	12Bh	GPI General Purpose Events Status (GPI_GPE_STS_GPP_E_0)—Offset 128h	00000000h
130h	133h	GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_0)—Offset 130h	00000000h
134h	137h	GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_0)—Offset 134h	00000000h
138h	13Bh	GPI General Purpose Events Enable (GPI_GPE_EN_GPP_E_0)—Offset 138h	00000000h
140h	143h	SMI Status (GPI_SMI_STS_GPP_C_0)—Offset 140h	00000000h
144h	147h	SMI Status (GPI_SMI_STS_GPP_D_0)—Offset 144h	00000000h
148h	14Bh	SMI Status (GPI_SMI_STS_GPP_E_0)—Offset 148h	00000000h
150h	153h	SMI Enable (GPI_SMI_EN_GPP_C_0)—Offset 150h	00000000h

Table 18-4. Summary of GPIO Community 1 Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
154h	157h	SMI Enable (GPI_SMI_EN_GPP_D_0)—Offset 154h	00000000h
158h	15Bh	SMI Enable (GPI_SMI_EN_GPP_E_0)—Offset 158h	00000000h
160h	163h	NMI Status (GPI_NMI_STS_GPP_C_0)—Offset 160h	00000000h
164h	167h	NMI Status (GPI_NMI_STS_GPP_D_0)—Offset 164h	00000000h
168h	16Bh	NMI Status (GPI_NMI_STS_GPP_E_0)—Offset 168h	00000000h
170h	173h	NMI Enable (GPI_NMI_EN_GPP_C_0)—Offset 170h	00000000h
174h	177h	NMI Enable (GPI_NMI_EN_GPP_D_0)—Offset 174h	00000000h
178h	17Bh	NMI Enable (GPI_NMI_EN_GPP_E_0)—Offset 178h	00000000h
204h	207h	PWM Control (PWMC)—Offset 204h	00000000h
20Ch	20Fh	GPIO Serial Blink Enable (GP_SER_BLINK)—Offset 20Ch	00000000h
210h	213h	GPIO Serial Blink Command/Status (GP_SER_CMDSTS)—Offset 210h	00000000h
214h	217h	GPIO Serial Blink Data (GP_SER_DATA)—Offset 214h	00000000h
400h	403h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_0)—Offset 400h	44000700h
404h	407h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_0)—Offset 404h	00000018h
408h	40Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_1)—Offset 408h	44000700h
40Ch	40Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_1)—Offset 40Ch	00000019h
410h	413h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_2)—Offset 410h	44000200h
414h	417h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_2)—Offset 414h	0000001Ah
418h	41Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_3)—Offset 418h	44000700h
41Ch	41Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_3)—Offset 41Ch	0000001Bh
420h	423h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_4)—Offset 420h	44000700h
424h	427h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_4)—Offset 424h	0000001Ch
428h	42Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_5)—Offset 428h	44000200h
42Ch	42Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_5)—Offset 42Ch	0000001Dh
430h	433h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_6)—Offset 430h	44000300h
434h	437h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_6)—Offset 434h	0000001Eh
438h	43Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_7)—Offset 438h	44000300h
43Ch	43Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_7)—Offset 43Ch	0000001Fh
440h	443h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_8)—Offset 440h	44000300h
444h	447h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_8)—Offset 444h	00000020h
448h	44Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_9)—Offset 448h	44000300h
44Ch	44Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_9)—Offset 44Ch	00000021h
450h	453h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_10)—Offset 450h	44000300h
454h	457h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_10)—Offset 454h	00000022h
458h	45Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_11)—Offset 458h	44000300h
45Ch	45Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_11)—Offset 45Ch	00000023h
460h	463h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_12)—Offset 460h	44000300h
464h	467h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_12)—Offset 464h	00000024h
468h	46Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_13)—Offset 468h	44000300h
46Ch	46Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_13)—Offset 46Ch	00000025h
470h	473h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_14)—Offset 470h	44000300h

Table 18-4. Summary of GPIO Community 1 Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
474h	477h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_14)—Offset 474h	00000026h
478h	47Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_15)—Offset 478h	44000300h
47Ch	47Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_15)—Offset 47Ch	00000027h
480h	483h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_16)—Offset 480h	44000300h
484h	487h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_16)—Offset 484h	00000028h
488h	48Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_17)—Offset 488h	44000300h
48Ch	48Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_17)—Offset 48Ch	00000029h
490h	493h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_18)—Offset 490h	44000300h
494h	497h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_18)—Offset 494h	0000002Ah
498h	49Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_19)—Offset 498h	44000300h
49Ch	49Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_19)—Offset 49Ch	0000002Bh
4A0h	4A3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_20)—Offset 4A0h	44000300h
4A4h	4A7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_20)—Offset 4A4h	0000002Ch
4A8h	4ABh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_21)—Offset 4A8h	44000300h
4ACh	4AFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_21)—Offset 4ACh	0000002Dh
4B0h	4B3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_22)—Offset 4B0h	44000300h
4B4h	4B7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_22)—Offset 4B4h	0000002Eh
4B8h	4BBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_23)—Offset 4B8h	44000300h
4BCh	4BFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_23)—Offset 4BCh	0000002Fh
4C0h	4C3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_0)—Offset 4C0h	44000300h
4C4h	4C7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_0)—Offset 4C4h	00000030h
4C8h	4CBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_1)—Offset 4C8h	44000300h
4CCh	4CFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_1)—Offset 4CCh	00000031h
4D0h	4D3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_2)—Offset 4D0h	44000300h
4D4h	4D7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_2)—Offset 4D4h	00000032h
4D8h	4DBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_3)—Offset 4D8h	44000300h
4DCh	4DFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_3)—Offset 4DCh	00000033h
4E0h	4E3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_4)—Offset 4E0h	44000300h
4E4h	4E7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_4)—Offset 4E4h	00000034h
4E8h	4EBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_5)—Offset 4E8h	44000300h
4ECh	4EFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_5)—Offset 4ECh	00000035h
4F0h	4F3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_6)—Offset 4F0h	44000300h
4F4h	4F7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_6)—Offset 4F4h	00000036h
4F8h	4FBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_7)—Offset 4F8h	44000300h
4FCh	4FFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_7)—Offset 4FCh	00000037h
500h	503h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_8)—Offset 500h	44000300h
504h	507h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_8)—Offset 504h	00000038h
508h	50Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_9)—Offset 508h	44000300h
50Ch	50Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_9)—Offset 50Ch	00000039h
510h	513h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_10)—Offset 510h	44000300h
514h	517h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_10)—Offset 514h	0000003Ah

Table 18-4. Summary of GPIO Community 1 Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
518h	51Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_11)—Offset 518h	44000300h
51Ch	51Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_11)—Offset 51Ch	0000003Bh
520h	523h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_12)—Offset 520h	44000300h
524h	527h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_12)—Offset 524h	0000003Ch
528h	52Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_13)—Offset 528h	44000300h
52Ch	52Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_13)—Offset 52Ch	0000003Dh
530h	533h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_14)—Offset 530h	44000300h
534h	537h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_14)—Offset 534h	0000003Eh
538h	53Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_15)—Offset 538h	44000300h
53Ch	53Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_15)—Offset 53Ch	0000003Fh
540h	543h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_16)—Offset 540h	44000300h
544h	547h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_16)—Offset 544h	00000040h
548h	54Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_17)—Offset 548h	44000300h
54Ch	54Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_17)—Offset 54Ch	00000041h
550h	553h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_18)—Offset 550h	44000300h
554h	557h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_18)—Offset 554h	00000042h
558h	55Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_19)—Offset 558h	44000300h
55Ch	55Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_19)—Offset 55Ch	00000043h
560h	563h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_20)—Offset 560h	44000300h
564h	567h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_20)—Offset 564h	00000044h
568h	56Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_21)—Offset 568h	44000300h
56Ch	56Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_21)—Offset 56Ch	00000045h
570h	573h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_22)—Offset 570h	44000300h
574h	577h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_22)—Offset 574h	00000046h
578h	57Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_23)—Offset 578h	44000300h
57Ch	57Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_23)—Offset 57Ch	00000047h
580h	583h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_0)—Offset 580h	44000300h
584h	587h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_0)—Offset 584h	00000048h
588h	58Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_1)—Offset 588h	44000300h
58Ch	58Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_1)—Offset 58Ch	00000049h
590h	593h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_2)—Offset 590h	44000300h
594h	597h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_2)—Offset 594h	0000004Ah
598h	59Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_3)—Offset 598h	44000300h
59Ch	59Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_3)—Offset 59Ch	0000004Bh
5A0h	5A3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_4)—Offset 5A0h	44000300h
5A4h	5A7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_4)—Offset 5A4h	0000004Ch
5A8h	5ABh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_5)—Offset 5A8h	44000300h
5ACh	5AFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_5)—Offset 5ACh	0000004Dh
5B0h	5B3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_6)—Offset 5B0h	44000300h
5B4h	5B7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_6)—Offset 5B4h	0000004Eh
5B8h	5BBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_7)—Offset 5B8h	44000300h

Table 18-4. Summary of GPIO Community 1 Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
5BCh	5BFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_7)—Offset 5BCh	0000004Fh
5C0h	5C3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_8)—Offset 5C0h	44000300h
5C4h	5C7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_8)—Offset 5C4h	00000050h
5C8h	5CBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_9)—Offset 5C8h	44000300h
5CCh	5CFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_9)—Offset 5CCh	00000051h
5D0h	5D3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_10)—Offset 5D0h	44000300h
5D4h	5D7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_10)—Offset 5D4h	00000052h
5D8h	5DBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_11)—Offset 5D8h	44000300h
5DCh	5DFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_11)—Offset 5DCh	00000053h
5E0h	5E3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_12)—Offset 5E0h	44000300h
5E4h	5E7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_12)—Offset 5E4h	00000054h

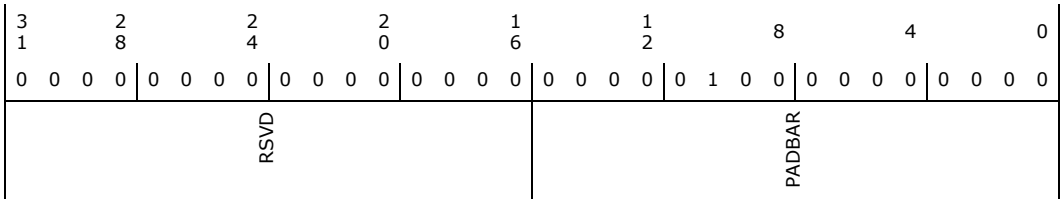
18.7.1 Pad Base Address (PADBAR)—Offset Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 400h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0h RO	Reserved
15:0	400h RO	**Pad Base Address (PADBAR):** This field provides the starting byte-align address of Pad0 register sets. It is meant for software to discover from where the very first Pad register (i.e. Pad0 register) starts to compute the next Pad address offsets.

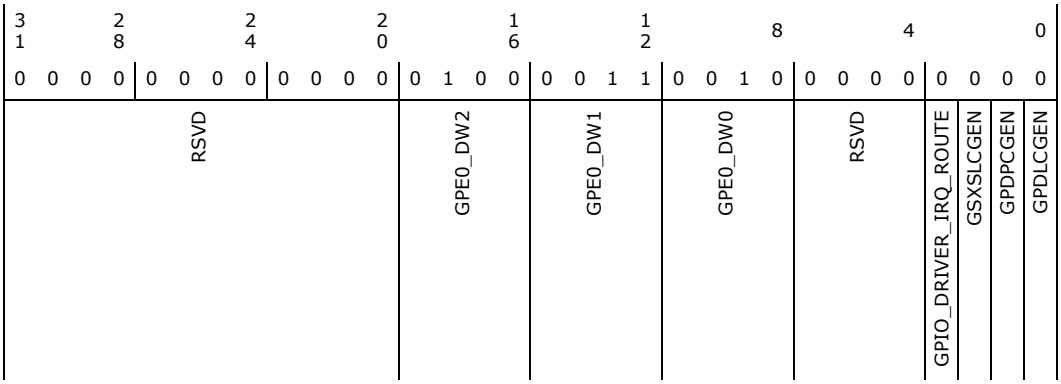
18.7.2 Miscellaneous Configuration (MISCCFG)—Offset 10h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 43200h



Bit Range	Default & Access	Field Name (ID): Description
31:20	0h RO	Reserved
19:16	4h RO	**GPIO Group to GPE_DW0 assignment encoding (GPE0_DW2):** This register assigns a specific GPIO Group to the ACPI GPE0[95:64]. 0h = GPP_A[23:0] mapped to GPE[87:64] and GPE[95:88] is not used 1h = GPP_B[23:0] mapped to GPE[87:64] and GPE[95:88] is not used 2h = GPP_C[23:0] mapped to GPE[87:64] and GPE[95:88] is not used] 3h = GPP_D[23:0] mapped to GPE[87:64] and GPE[95:88] is not used 4h = GPP_E[12:0] mapped to GPE[76:64] and GPE[95:77] is not used 5h = GPP_F[23:0] mapped to GPE[87:64] and GPE[95:88] is not used 6h = GPP_G[23:0] mapped to GPE[87:64] and GPE[95:88] is not used 7h = GPP_H[23:0] mapped to GPE[87:64] and GPE[95:88] is not used 8h = GPP_I[10:0] mapped to GPE[74:64] and GPE[95:75] is not used 9h = GPD[11:0] mapped to GPE[75:64] and GPE[95:76] is not used Ah= GPP_J[23:0] mapped to GPE[87:64] and GPE[95:88] not used Bh= GPP_K[11:0]mapped to GPE[75:64] and GPE[95:75] is not used
15:12	3h RO	**GPIO Group to GPE_DW0 assignment encoding (GPE0_DW1):** This register assigns a specific GPIO Group to the ACPI GPE0[63:32]. 0h = GPP_A[23:0] mapped to GPE[55:32] and GPE[63:56] is not used 1h = GPP_B[23:0] mapped to GPE[55:32] and GPE[63:56] is not used 2h = GPP_C[23:0] mapped to GPE[55:32] and GPE[63:56] is not used] 3h = GPP_D[23:0] mapped to GPE[55:32] and GPE[63:56] is not used 4h = GPP_E[12:0] mapped to GPE[44:32] and GPE[63:45] is not used 5h = GPP_F[23:0] mapped to GPE[55:32] and GPE[63:56] is not used 6h = GPP_G[23:0] mapped to GPE[55:32] and GPE[63:56] is not used 7h = GPP_H[23:0] mapped to GPE[55:32] and GPE[63:56] is not used 8h = GPP_I[10:0] mapped to GPE[42:32] and GPE[63:43] is not used 9h = GPD[11:0] mapped to GPE[43:32] and GPE[63:44] is not used Ah= GPP_J[23:0] mapped to GPE[87:64] and GPE[95:88] not used Bh= GPP_K[11:0]mapped to GPE[75:64] and GPE[95:75] is not used
11:8	2h RO	**GPIO Group to GPE_DW0 assignment encoding (GPE0_DW0):** This register assigns a specific GPIO Group to the ACPI GPE0[31:0]. 0h = GPP_A[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 1h = GPP_B[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 2h = GPP_C[23:0] mapped to GPE[23:0] and GPE[31:24] is not used] 3h = GPP_D[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 4h = GPP_E[12:0] mapped to GPE[12:0] and GPE[31:13] is not used 5h = GPP_F[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 6h = GPP_G[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 7h = GPP_H[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 8h = GPP_I[10:0] mapped to GPE[10:0] and GPE[31:11] is not used 9h = GPD[11:0] mapped to GPE[11:0] and GPE[31:12] is not used Ah= GPP_J[23:0] mapped to GPE[23:0] and GPE[31:24] not used Bh= GPP_K[11:0]mapped to GPE[11:0] and GPE[31:12] is not used

Bit Range	Default & Access	Field Name (ID): Description
7:4	0h RO	Reserved
3	0h RW	**GPIO Driver IRQ Route (GPIO_DRIVER_IRQ_ROUTE):** GPIO Driver IRQ_ROUTE[1:0]: Specifies the APIC IRQ globally for all pads within the current community (GPI_IS with corresponding GPI_IE enable). 0 = IRQ14 1 = IRQ15
2	0h RW	**GSX Static Local Clock Gating (GSXSLCGEN):** GSX Static Local Clock Gating (GSXSLCGEn) Specify whether the GSX controller should be statically clock gated for power saving if it is not enabled (even though the capability is available). 0 = Disable dynamic local clock gating 1 = Enable dynamic local clock gating
1	0h RW	**GPIO Dynamic Partition Clock Gating Enable (GPDPCGEN):** Specifies whether the GPIO Community should take part in partition clock gating 0 = Disable participation in dynamic partition clock gating. 1 = Enable participation in dynamic partition clock gating.
0	0h RW	**GPIO Dynamic Local Clock Gating Enable (GPDLCGEN):** Specifies whether the GPIO Community should perform local clock gating. 0 = Disable dynamic local clock gating. 1 = Enable dynamic local clock gating.

18.7.3 Pad Ownership (PAD_OWN_GPP_C_0)—Offset 20h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD	PAD_OWN_GPP_C_7			RSVD	PAD_OWN_GPP_C_6			RSVD	PAD_OWN_GPP_C_5			RSVD	PAD_OWN_GPP_C_4			RSVD	PAD_OWN_GPP_C_3			RSVD	PAD_OWN_GPP_C_2			RSVD	PAD_OWN_GPP_C_1			RSVD	PAD_OWN_GPP_C_0		

Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RO	**Pad Ownership (PAD_OWN_GPP_C_7):** Same description as bit 0, except that the bit field applies to GPP_C7.
27:26	0h RO	Reserved
25:24	0h RO	**Pad Ownership (PAD_OWN_GPP_C_6):** Same description as bit 0, except that the bit field applies to GPP_C6.
23:22	0h RO	Reserved
21:20	0h RO	**Pad Ownership (PAD_OWN_GPP_C_5):** Same description as bit 0, except that the bit field applies to GPP_C5.

Bit Range	Default & Access	Field Name (ID): Description
19:18	0h RO	Reserved
17:16	0h RO	**Pad Ownership (PAD_OWN_GPP_C_4):** Same description as bit 0, except that the bit field applies to GPP_C4.
15:14	0h RO	Reserved
13:12	0h RO	**Pad Ownership (PAD_OWN_GPP_C_3):** Same description as bit 0, except that the bit field applies to GPP_C3.
11:10	0h RO	Reserved
9:8	0h RO	**Pad Ownership (PAD_OWN_GPP_C_2):** Same description as bit 0, except that the bit field applies to GPP_C2.
7:6	0h RO	Reserved
5:4	0h RO	**Pad Ownership (PAD_OWN_GPP_C_1):** Same description as bit 0, except that the bit field applies to GPP_C1.
3:2	0h RO	Reserved
1:0	0h RO	**Pad Ownership (PAD_OWN_GPP_C_0):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = ME GPIO Mode. ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad

18.7.4 Pad Ownership (PAD_OWN_GPP_C_1)—Offset 24h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD	PAD_OWN_GPP_C_15			RSVD	PAD_OWN_GPP_C_14			RSVD	PAD_OWN_GPP_C_13			RSVD	PAD_OWN_GPP_C_12			RSVD	PAD_OWN_GPP_C_11			RSVD	PAD_OWN_GPP_C_10			RSVD	PAD_OWN_GPP_C_9			RSVD	PAD_OWN_GPP_C_8		

Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RO	**Pad Ownership (PAD_OWN_GPP_C_15):** Same description as bit 0, except that the bit field applies to GPP_C15.
27:26	0h RO	Reserved
25:24	0h RO	**Pad Ownership (PAD_OWN_GPP_C_14):** Same description as bit 0, except that the bit field applies to GPP_C14.
23:22	0h RO	Reserved
21:20	0h RO	**Pad Ownership (PAD_OWN_GPP_C_13):** Same description as bit 0, except that the bit field applies to GPP_C13.
19:18	0h RO	Reserved
17:16	0h RO	**Pad Ownership (PAD_OWN_GPP_C_12):** Same description as bit 0, except that the bit field applies to GPP_C12.
15:14	0h RO	Reserved
13:12	0h RO	**Pad Ownership (PAD_OWN_GPP_C_11):** Same description as bit 0, except that the bit field applies to GPP_C11.
11:10	0h RO	Reserved
9:8	0h RO	**Pad Ownership (PAD_OWN_GPP_C_10):** Same description as bit 0, except that the bit field applies to GPP_C10.
7:6	0h RO	Reserved
5:4	0h RO	**Pad Ownership (PAD_OWN_GPP_C_9):** Same description as bit 0, except that the bit field applies to GPP_C9.
3:2	0h RO	Reserved
1:0	0h RO	**Pad Ownership (PAD_OWN_GPP_C_8):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = ME GPIO Mode. ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad

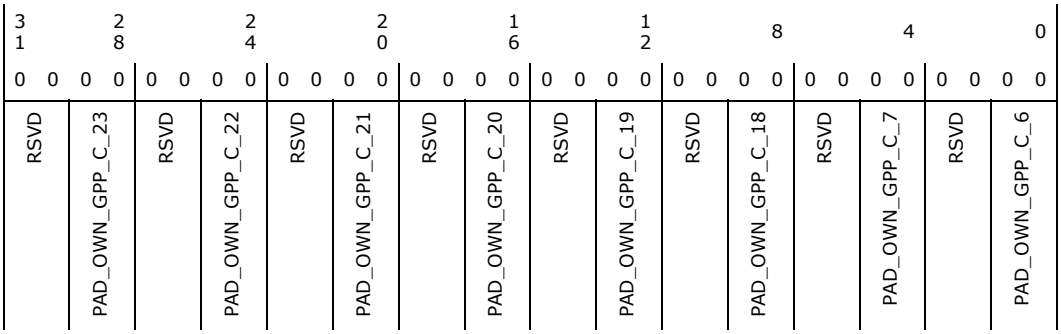
18.7.5 Pad Ownership (PAD_OWN_GPP_C_2)—Offset 28h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RO	**Pad Ownership (PAD_OWN_GPP_C_23):** Same description as bit 0, except that the bit field applies to GPP_C23.
27:26	0h RO	Reserved
25:24	0h RO	**Pad Ownership (PAD_OWN_GPP_C_22):** Same description as bit 0, except that the bit field applies to GPP_C22.
23:22	0h RO	Reserved
21:20	0h RO	**Pad Ownership (PAD_OWN_GPP_C_21):** Same description as bit 0, except that the bit field applies to GPP_C21.
19:18	0h RO	Reserved
17:16	0h RO	**Pad Ownership (PAD_OWN_GPP_C_20):** Same description as bit 0, except that the bit field applies to GPP_C20.
15:14	0h RO	Reserved
13:12	0h RO	**Pad Ownership (PAD_OWN_GPP_C_19):** Same description as bit 0, except that the bit field applies to GPP_C19.
11:10	0h RO	Reserved
9:8	0h RO	**Pad Ownership (PAD_OWN_GPP_C_18):** Same description as bit 0, except that the bit field applies to GPP_C18.
7:6	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
5:4	0h RO	**Pad Ownership (PAD_OWN_GPP_C_17):** Same description as bit 0, except that the bit field applies to GPP_C17.
3:2	0h RO	Reserved
1:0	0h RO	**Pad Ownership (PAD_OWN_GPP_C_16):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = ME GPIO Mode. ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad

18.7.6 Pad Ownership (PAD_OWN_GPP_D_0)—Offset 2Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD	PAD_OWN_GPP_D_7			RSVD	PAD_OWN_GPP_D_6			RSVD	PAD_OWN_GPP_D_5			RSVD	PAD_OWN_GPP_D_4			RSVD	PAD_OWN_GPP_D_3			RSVD	PAD_OWN_GPP_D_2			RSVD	PAD_OWN_GPP_D_1			RSVD	PAD_OWN_GPP_D_0		

Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RO	**Pad Ownership (PAD_OWN_GPP_D_7):** Same description as bit 0, except that the bit field applies to GPP_D7.
27:26	0h RO	Reserved
25:24	0h RO	**Pad Ownership (PAD_OWN_GPP_D_6):** Same description as bit 0, except that the bit field applies to GPP_D6.
23:22	0h RO	Reserved
21:20	0h RO	**Pad Ownership (PAD_OWN_GPP_D_5):** Same description as bit 0, except that the bit field applies to GPP_D5.
19:18	0h RO	Reserved
17:16	0h RO	**Pad Ownership (PAD_OWN_GPP_D_4):** Same description as bit 0, except that the bit field applies to GPP_D4.
15:14	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
13:12	0h RO	**Pad Ownership (PAD_OWN_GPP_D_3):** Same description as bit 0, except that the bit field applies to GPP_D3.
11:10	0h RO	Reserved
9:8	0h RO	**Pad Ownership (PAD_OWN_GPP_D_2):** Same description as bit 0, except that the bit field applies to GPP_D2.
7:6	0h RO	Reserved
5:4	0h RO	**Pad Ownership (PAD_OWN_GPP_D_1):** Same description as bit 0, except that the bit field applies to GPP_D1.
3:2	0h RO	Reserved
1:0	0h RO	**Pad Ownership (PAD_OWN_GPP_D_0):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = ME GPIO Mode. ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad

18.7.7 Pad Ownership (PAD_OWN_GPP_D_1)—Offset 30h

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD		PAD_OWN_GPP_D_15		RSVD		PAD_OWN_GPP_D_14		RSVD		PAD_OWN_GPP_D_13		RSVD		PAD_OWN_GPP_D_12		RSVD		PAD_OWN_GPP_D_11		RSVD		PAD_OWN_GPP_D_10		RSVD		PAD_OWN_GPP_D_9		RSVD		PAD_OWN_GPP_D_8	

Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RO	**Pad Ownership (PAD_OWN_GPP_D_15):** Same description as bit 0, except that the bit field applies to GPP_D15.
27:26	0h RO	Reserved
25:24	0h RO	**Pad Ownership (PAD_OWN_GPP_D_14):** Same description as bit 0, except that the bit field applies to GPP_D14.
23:22	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
21:20	0h RO	**Pad Ownership (PAD_OWN_GPP_D_13):** Same description as bit 0, except that the bit field applies to GPP_D13.
19:18	0h RO	Reserved
17:16	0h RO	**Pad Ownership (PAD_OWN_GPP_D_12):** Same description as bit 0, except that the bit field applies to GPP_D12.
15:14	0h RO	Reserved
13:12	0h RO	**Pad Ownership (PAD_OWN_GPP_D_11):** Same description as bit 0, except that the bit field applies to GPP_D11.
11:10	0h RO	Reserved
9:8	0h RO	**Pad Ownership (PAD_OWN_GPP_D_10):** Same description as bit 0, except that the bit field applies to GPP_D10.
7:6	0h RO	Reserved
5:4	0h RO	**Pad Ownership (PAD_OWN_GPP_D_9):** Same description as bit 0, except that the bit field applies to GPP_D9.
3:2	0h RO	Reserved
1:0	0h RO	**Pad Ownership (PAD_OWN_GPP_D_8):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = ME GPIO Mode. ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad

18.7.8 Pad Ownership (PAD_OWN_GPP_D_2)—Offset 34h

Same description as PAD_OWN_GPP_C_0, except that this register is for GPP_D[23:16].

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31:28		27:24		23:20		19:16		15:12		11:8		7:4		3:0	
0 0 0 0		0 0 0 0		0 0 0 0		0 0 0 0		0 0 0 0		0 0 0 0		0 0 0 0		0 0 0 0	
RSVD	PAD_OWN_GPP_D_23	RSVD	PAD_OWN_GPP_D_22	RSVD	PAD_OWN_GPP_D_21	RSVD	PAD_OWN_GPP_D_20	RSVD	PAD_OWN_GPP_D_19	RSVD	PAD_OWN_GPP_D_18	RSVD	PAD_OWN_GPP_D_7	RSVD	PAD_OWN_GPP_D_6

Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RO	**Pad Ownership (PAD_OWN_GPP_D_23):** Same description as bit 0, except that the bit field applies to GPP_D23.
27:26	0h RO	Reserved
25:24	0h RO	**Pad Ownership (PAD_OWN_GPP_D_22):** Same description as bit 0, except that the bit field applies to GPP_D22.
23:22	0h RO	Reserved
21:20	0h RO	**Pad Ownership (PAD_OWN_GPP_D_21):** Same description as bit 0, except that the bit field applies to GPP_D21.
19:18	0h RO	Reserved
17:16	0h RO	**Pad Ownership (PAD_OWN_GPP_D_20):** Same description as bit 0, except that the bit field applies to GPP_D20.
15:14	0h RO	Reserved
13:12	0h RO	**Pad Ownership (PAD_OWN_GPP_D_19):** Same description as bit 0, except that the bit field applies to GPP_D19.
11:10	0h RO	Reserved
9:8	0h RO	**Pad Ownership (PAD_OWN_GPP_D_18):** Same description as bit 0, except that the bit field applies to GPP_D18.
7:6	0h RO	Reserved
5:4	0h RO	**Pad Ownership (PAD_OWN_GPP_D_17):** Same description as bit 0, except that the bit field applies to GPP_D17.
3:2	0h RO	Reserved
1:0	0h RO	**Pad Ownership (PAD_OWN_GPP_D_16):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = ME GPIO Mode. ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad

18.7.9 Pad Ownership (PAD_OWN_GPP_E_0)—Offset 38h

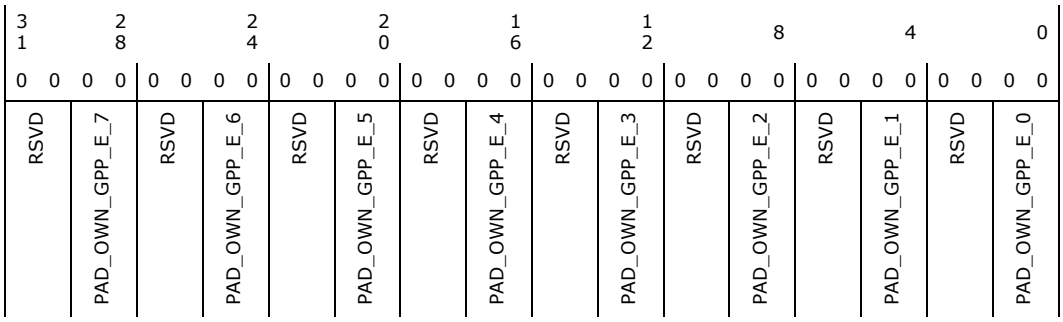
Same description as PAD_OWN_GPP_C_0, except that this register is for GPP_E[7:0].

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RO	**Pad Ownership (PAD_OWN_GPP_E_7):** Same description as bit 0, except that the bit field applies to GPP_E7.
27:26	0h RO	Reserved
25:24	0h RO	**Pad Ownership (PAD_OWN_GPP_E_6):** Same description as bit 0, except that the bit field applies to GPP_E6.
23:22	0h RO	Reserved
21:20	0h RO	**Pad Ownership (PAD_OWN_GPP_E_5):** Same description as bit 0, except that the bit field applies to GPP_E5.
19:18	0h RO	Reserved
17:16	0h RO	**Pad Ownership (PAD_OWN_GPP_E_4):** Same description as bit 0, except that the bit field applies to GPP_E4.
15:14	0h RO	Reserved
13:12	0h RO	**Pad Ownership (PAD_OWN_GPP_E_3):** Same description as bit 0, except that the bit field applies to GPP_E3.
11:10	0h RO	Reserved
9:8	0h RO	**Pad Ownership (PAD_OWN_GPP_E_2):** Same description as bit 0, except that the bit field applies to GPP_E2.
7:6	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
5:4	0h RO	**Pad Ownership (PAD_OWN_GPP_E_1):** Same description as bit 0, except that the bit field applies to GPP_E1.
3:2	0h RO	Reserved
1:0	0h RO	**Pad Ownership (PAD_OWN_GPP_E_0):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = ME GPIO Mode. ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad

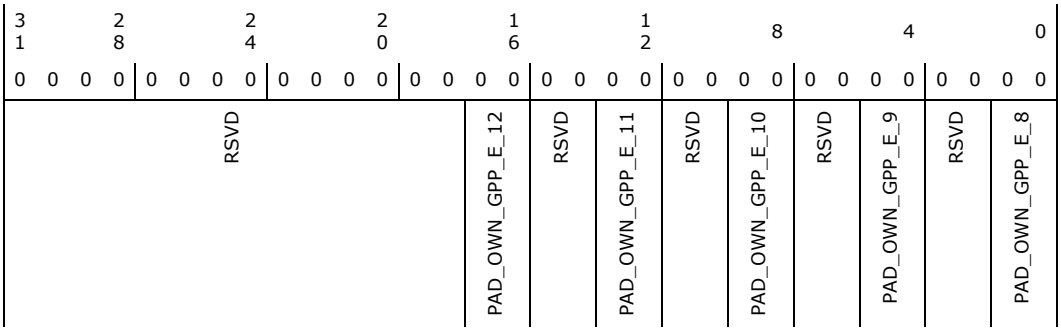
18.7.10 Pad Ownership (PAD_OWN_GPP_E_1)—Offset 3Ch

Same description as PAD_OWN_GPP_C_0, except that this register is for GPP_E[12:8].

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:18	0h RO	Reserved
17:16	0h RO	**Pad Ownership (PAD_OWN_GPP_E_12):** Same description as bit 0, except that the bit field applies to GPP_E12.
15:14	0h RO	Reserved
13:12	0h RO	**Pad Ownership (PAD_OWN_GPP_E_11):** Same description as bit 0, except that the bit field applies to GPP_E11.
11:10	0h RO	Reserved
9:8	0h RO	**Pad Ownership (PAD_OWN_GPP_E_10):** Same description as bit 0, except that the bit field applies to GPP_E10.
7:6	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
5:4	0h RO	**Pad Ownership (PAD_OWN_GPP_E_9):** Same description as bit 0, except that the bit field applies to GPP_E9.
3:2	0h RO	Reserved
1:0	0h RO	**Pad Ownership (PAD_OWN_GPP_E_8):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = ME GPIO Mode. ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad

18.7.11 Pad Configuration Lock (PADCFGLOCK_GPP_C_0)—Offset 60h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD								PADCFGLOCK_GPP_C_23	PADCFGLOCK_GPP_C_22	PADCFGLOCK_GPP_C_21	PADCFGLOCK_GPP_C_20	PADCFGLOCK_GPP_C_19	PADCFGLOCK_GPP_C_18	PADCFGLOCK_GPP_C_17	PADCFGLOCK_GPP_C_16	PADCFGLOCK_GPP_C_15	PADCFGLOCK_GPP_C_14	PADCFGLOCK_GPP_C_13	PADCFGLOCK_GPP_C_12	PADCFGLOCK_GPP_C_11	PADCFGLOCK_GPP_C_10	PADCFGLOCK_GPP_C_9	PADCFGLOCK_GPP_C_8	PADCFGLOCK_GPP_C_7	PADCFGLOCK_GPP_C_6	PADCFGLOCK_GPP_C_5	PADCFGLOCK_GPP_C_4	PADCFGLOCK_GPP_C_3	PADCFGLOCK_GPP_C_2	PADCFGLOCK_GPP_C_1	PADCFGLOCK_GPP_C_0

Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_C_23):** Applied to GPP_C23. Same description as bit 0.
22	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_C_22):** Applied to GPP_C22. Same description as bit 0.
21	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_C_21):** Applied to GPP_C21. Same description as bit 0.
20	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_C_20):** Applied to GPP_C20. Same description as bit 0.
19	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_C_19):** Applied to GPP_C19. Same description as bit 0.
18	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_C_18):** Applied to GPP_C18. Same description as bit 0.
17	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_C_17):** Applied to GPP_C17. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
16	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_C_16):** Applied to GPP_C16. Same description as bit 0.
15	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_C_15):** Applied to GPP_C15. Same description as bit 0.
14	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_C_14):** Applied to GPP_C14. Same description as bit 0.
13	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_C_13):** Applied to GPP_C13. Same description as bit 0.
12	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_C_12):** Applied to GPP_C12. Same description as bit 0.
11	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_C_11):** Applied to GPP_C11. Same description as bit 0.
10	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_C_10):** Applied to GPP_C10. Same description as bit 0.
9	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_C_9):** Applied to GPP_C9. Same description as bit 0.
8	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_C_8):** Applied to GPP_C8. Same description as bit 0.
7	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_C_7):** Applied to GPP_C7. Same description as bit 0.
6	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_C_6):** Applied to GPP_C6. Same description as bit 0.
5	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_C_5):** Applied to GPP_C5. Same description as bit 0.
4	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_C_4):** Applied to GPP_C4. Same description as bit 0.
3	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_C_3):** Applied to GPP_C3. Same description as bit 0.
2	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_C_2):** Applied to GPP_C2. Same description as bit 0.
1	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_C_1):** Applied to GPP_C1. Same description as bit 0.
0	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_C_0):** Pad Configuration Lock locks specific register fields in the GPP specific registers from being configured. The registers affected become Read-Only and software writes to these registers have no effect. 0 = Unlock 1 = Lock the following register fields as read-only (RO): - Pad Configuration registers (exclude GPIOTXState) - GPI_NMI_EN Register (if implemented) - GPI_SMI_EN Register (if implemented) - GPI_GPE_EN Register (if implemented) When PadCfgLock bit is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

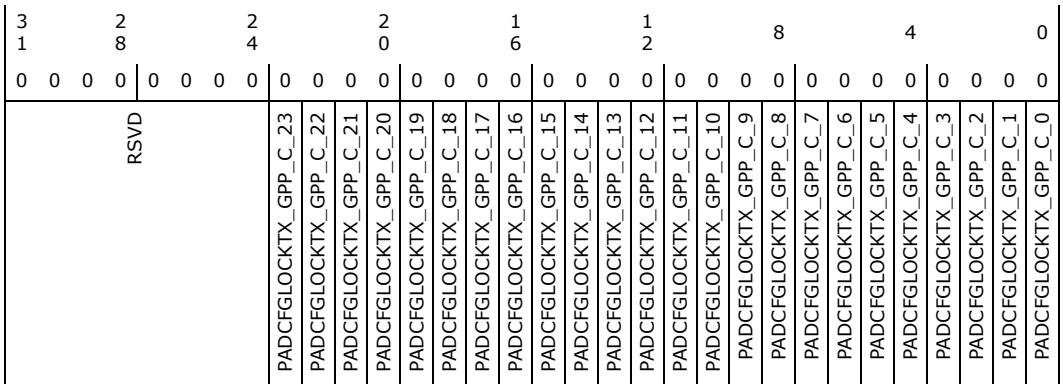
18.7.12 Pad Configuration Lock (PADCFGLOCKTX_GPP_C_0)—Offset 64h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_C_23):** Applied to GPP_C23. Same description as bit 0.
22	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_C_22):** Applied to GPP_C22. Same description as bit 0.
21	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_C_21):** Applied to GPP_C21. Same description as bit 0.
20	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_C_20):** Applied to GPP_C20. Same description as bit 0.
19	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_C_19):** Applied to GPP_C19. Same description as bit 0.
18	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_C_18):** Applied to GPP_C18. Same description as bit 0.
17	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_C_17):** Applied to GPP_C17. Same description as bit 0.
16	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_C_16):** Applied to GPP_C16. Same description as bit 0.
15	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_C_15):** Applied to GPP_C15. Same description as bit 0.
14	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_C_14):** Applied to GPP_C14. Same description as bit 0.
13	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_C_13):** Applied to GPP_C13. Same description as bit 0.
12	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_C_12):** Applied to GPP_C12. Same description as bit 0.
11	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_C_11):** Applied to GPP_C11. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
10	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_C_10):** Applied to GPP_C10. Same description as bit 0.
9	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_C_9):** Applied to GPP_C9. Same description as bit 0.
8	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_C_8):** Applied to GPP_C8. Same description as bit 0.
7	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_C_7):** Applied to GPP_C7. Same description as bit 0.
6	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_C_6):** Applied to GPP_C6. Same description as bit 0.
5	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_C_5):** Applied to GPP_C5. Same description as bit 0.
4	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_C_4):** Applied to GPP_C4. Same description as bit 0.
3	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_C_3):** Applied to GPP_C1. Same description as bit 0.
2	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_C_2):** Applied to GPP_C1. Same description as bit 0.
1	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_C_1):** Applied to GPP_C1. Same description as bit 0.
0	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_C_0):** PadCfgLockTx locks the GPIOTxState bit from being configured. The GPIOTxState register becomes Read-Only and software writes to the register have no effect. 0 = Unlock 1 = Locks the Pad Configuration GPIOTXState field as read-only (RO) When PadCfgLockTx is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

18.7.13 Pad Configuration Lock (PADCFGLOCK_GPP_D_0)—Offset 68h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD								PADCFGLOCK_GPP_D_23	PADCFGLOCK_GPP_D_22	PADCFGLOCK_GPP_D_21	PADCFGLOCK_GPP_D_20	PADCFGLOCK_GPP_D_19	PADCFGLOCK_GPP_D_18	PADCFGLOCK_GPP_D_17	PADCFGLOCK_GPP_D_16	PADCFGLOCK_GPP_D_15	PADCFGLOCK_GPP_D_14	PADCFGLOCK_GPP_D_13	PADCFGLOCK_GPP_D_12	PADCFGLOCK_GPP_D_11	PADCFGLOCK_GPP_D_10	PADCFGLOCK_GPP_D_9	PADCFGLOCK_GPP_D_8	PADCFGLOCK_GPP_D_7	PADCFGLOCK_GPP_D_6	PADCFGLOCK_GPP_D_5	PADCFGLOCK_GPP_D_4	PADCFGLOCK_GPP_D_3	PADCFGLOCK_GPP_D_2	PADCFGLOCK_GPP_D_1	PADCFGLOCK_GPP_D_0

Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_D_23):** Applied to GPP_D23. Same description as bit 0.
22	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_D_22):** Applied to GPP_D22. Same description as bit 0.
21	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_D_21):** Applied to GPP_D21. Same description as bit 0.
20	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_D_20):** Applied to GPP_D20. Same description as bit 0.
19	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_D_19):** Applied to GPP_D19. Same description as bit 0.
18	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_D_18):** Applied to GPP_D18. Same description as bit 0.
17	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_D_17):** Applied to GPP_D17. Same description as bit 0.
16	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_D_16):** Applied to GPP_D16. Same description as bit 0.
15	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_D_15):** Applied to GPP_D15. Same description as bit 0.
14	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_D_14):** Applied to GPP_D14. Same description as bit 0.
13	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_D_13):** Applied to GPP_D13. Same description as bit 0.
12	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_D_12):** Applied to GPP_D12. Same description as bit 0.
11	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_D_11):** Applied to GPP_D11. Same description as bit 0.
10	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_D_10):** Applied to GPP_D10. Same description as bit 0.
9	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_D_9):** Applied to GPP_D9. Same description as bit 0.
8	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_D_8):** Applied to GPP_D8. Same description as bit 0.
7	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_D_7):** Applied to GPP_D7. Same description as bit 0.
6	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_D_6):** Applied to GPP_D6. Same description as bit 0.
5	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_D_5):** Applied to GPP_D5. Same description as bit 0.
4	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_D_4):** Applied to GPP_D4. Same description as bit 0.
3	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_D_3):** Applied to GPP_D3. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
2	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_D_2):** Applied to GPP_D2. Same description as bit 0.
1	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_D_1):** Applied to GPP_D1. Same description as bit 0.
0	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_D_0):** Pad Configuration Lock locks specific register fields in the GPP specific registers from being configured. The registers affected become Read-Only and software writes to these registers have no effect. 0 = Unlock 1 = Lock the following register fields as read-only (RO): - Pad Configuration registers (exclude GPIOTXState) - GPI_NMI_EN Register (if implemented) - GPI_SMI_EN Register (if implemented) - GPI_GPE_EN Register (if implemented) When PadCfgLock bit is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

18.7.14 Pad Configuration Lock (PADCFGLOCKTX_GPP_D_0)—Offset 6Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD								PADCFGLOCKTX_GPP_D_23	PADCFGLOCKTX_GPP_D_22	PADCFGLOCKTX_GPP_D_21	PADCFGLOCKTX_GPP_D_20	PADCFGLOCKTX_GPP_D_19	PADCFGLOCKTX_GPP_D_18	PADCFGLOCKTX_GPP_D_17	PADCFGLOCKTX_GPP_D_16	PADCFGLOCKTX_GPP_D_15	PADCFGLOCKTX_GPP_D_14	PADCFGLOCKTX_GPP_D_13	PADCFGLOCKTX_GPP_D_12	PADCFGLOCKTX_GPP_D_11	PADCFGLOCKTX_GPP_D_10	PADCFGLOCKTX_GPP_D_9	PADCFGLOCKTX_GPP_D_8	PADCFGLOCKTX_GPP_D_7	PADCFGLOCKTX_GPP_D_6	PADCFGLOCKTX_GPP_D_5	PADCFGLOCKTX_GPP_D_4	PADCFGLOCKTX_GPP_D_3	PADCFGLOCKTX_GPP_D_2	PADCFGLOCKTX_GPP_D_1	PADCFGLOCKTX_GPP_D_0

Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_D_23):** Applied to GPP_D23. Same description as bit 0.
22	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_D_22):** Applied to GPP_D22. Same description as bit 0.
21	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_D_21):** Applied to GPP_D21. Same description as bit 0.
20	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_D_20):** Applied to GPP_D20. Same description as bit 0.
19	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_D_19):** Applied to GPP_D19. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
18	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_D_18):** Applied to GPP_D18. Same description as bit 0.
17	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_D_17):** Applied to GPP_D17. Same description as bit 0.
16	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_D_16):** Applied to GPP_D16. Same description as bit 0.
15	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_D_15):** Applied to GPP_D15. Same description as bit 0.
14	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_D_14):** Applied to GPP_D14. Same description as bit 0.
13	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_D_13):** Applied to GPP_D13. Same description as bit 0.
12	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_D_12):** Applied to GPP_D12. Same description as bit 0.
11	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_D_11):** Applied to GPP_D11. Same description as bit 0.
10	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_D_10):** Applied to GPP_D10. Same description as bit 0.
9	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_D_9):** Applied to GPP_D9. Same description as bit 0.
8	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_D_8):** Applied to GPP_D8. Same description as bit 0.
7	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_D_7):** Applied to GPP_D7. Same description as bit 0.
6	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_D_6):** Applied to GPP_D6. Same description as bit 0.
5	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_D_5):** Applied to GPP_D5. Same description as bit 0.
4	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_D_4):** Applied to GPP_D4. Same description as bit 0.
3	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_D_3):** Applied to GPP_D1. Same description as bit 0.
2	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_D_2):** Applied to GPP_D1. Same description as bit 0.
1	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_D_1):** Applied to GPP_D1. Same description as bit 0.
0	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_D_0):** PadCfgLockTx locks the GPIOTxState bit from being configured. The GPIOTxState register becomes Read-Only and software writes to the register have no effect. 0 = Unlock 1 = Locks the Pad Configuration GPIOTXState field as read-only (RO) When PadCfgLockTx is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

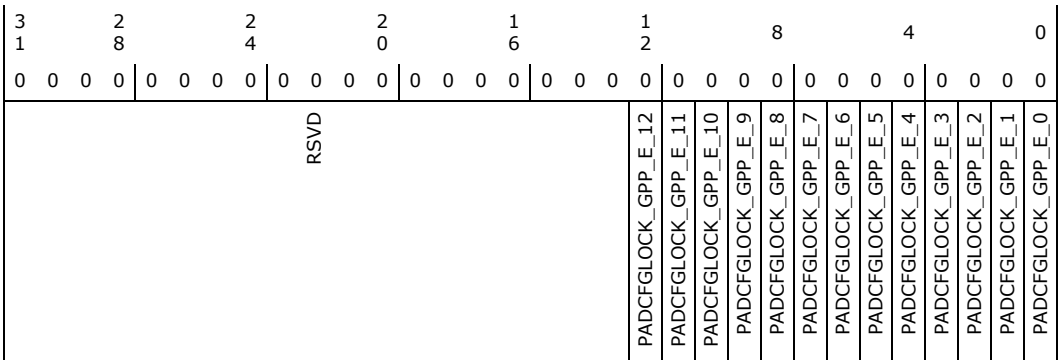
18.7.15 Pad Configuration Lock (PADCFGLOCK_GPP_E_0)—Offset 70h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:13	0h RO	Reserved
12	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_E_12):** Applied to GPP_E12. Same description as bit 0.
11	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_E_11):** Applied to GPP_E11. Same description as bit 0.
10	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_E_10):** Applied to GPP_E10. Same description as bit 0.
9	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_E_9):** Applied to GPP_E9. Same description as bit 0.
8	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_E_8):** Applied to GPP_E8. Same description as bit 0.
7	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_E_7):** Applied to GPP_E7. Same description as bit 0.
6	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_E_6):** Applied to GPP_E6. Same description as bit 0.
5	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_E_5):** Applied to GPP_E5. Same description as bit 0.
4	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_E_4):** Applied to GPP_E4. Same description as bit 0.
3	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_E_3):** Applied to GPP_E3. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
2	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_E_2):** Applied to GPP_E2. Same description as bit 0.
1	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_E_1):** Applied to GPP_E1. Same description as bit 0.
0	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_E_0):** Pad Configuration Lock locks specific register fields in the GPP specific registers from being configured. The registers affected become Read-Only and software writes to these registers have no effect. 0 = Unlock 1 = Lock the following register fields as read-only (RO): - Pad Configuration registers (exclude GPIOTXState) - GPI_NMI_EN Register (if implemented) - GPI_SMI_EN Register (if implemented) - GPI_GPE_EN Register (if implemented) When PadCfgLock bit is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

18.7.16 Pad Configuration Lock (PADCFGLOCKTX_GPP_E_0)—Offset 74h

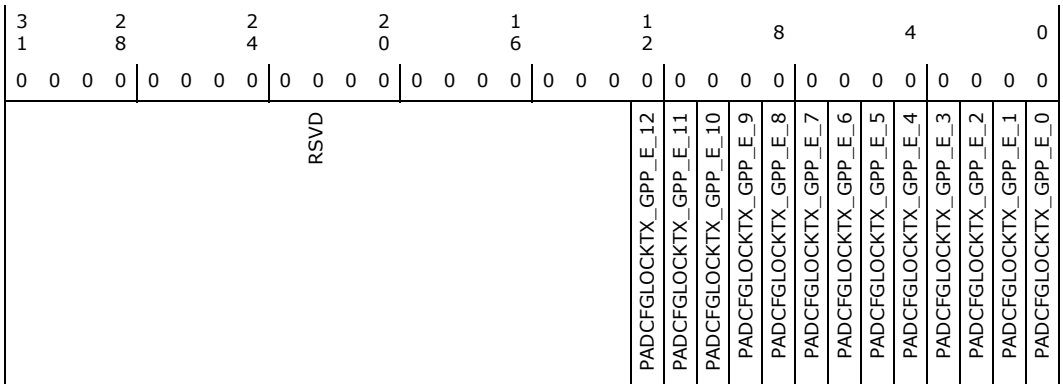
Same description as PADCFGLOCKTX_GPP_C_0 register, except this register applies to GPP_E group only.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:13	0h RO	Reserved
12	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_E_12):** Applied to GPP_E12. Same description as bit 0.
11	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_E_11):** Applied to GPP_E11. Same description as bit 0.
10	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_E_10):** Applied to GPP_E10. Same description as bit 0.
9	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_E_9):** Applied to GPP_E9. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
8	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_E_8):** Applied to GPP_E8. Same description as bit 0.
7	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_E_7):** Applied to GPP_E7. Same description as bit 0.
6	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_E_6):** Applied to GPP_E6. Same description as bit 0.
5	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_E_5):** Applied to GPP_E5. Same description as bit 0.
4	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_E_4):** Applied to GPP_E4. Same description as bit 0.
3	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_E_3):** Applied to GPP_E3. Same description as bit 0.
2	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_E_2):** Applied to GPP_E2. Same description as bit 0.
1	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_E_1):** Applied to GPP_E1. Same description as bit 0.
0	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_E_0):** PadCfgLockTx locks the GPIOTxState bit from being configured. The GPIOTxState register becomes Read-Only and software writes to the register have no effect. 0 = Unlock 1 = Locks the Pad Configuration GPIOTXState field as read-only (RO) When PadCfgLockTx is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

18.7.17 Host Software Pad Ownership (HOSTSW_OWN_GPP_C_0)—Offset 80h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	28	24	20	16	12	8	4	0
0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0
RSVD		HOSTSW_OWN_GPP_C_23	HOSTSW_OWN_GPP_C_22	HOSTSW_OWN_GPP_C_21	HOSTSW_OWN_GPP_C_20	HOSTSW_OWN_GPP_C_19	HOSTSW_OWN_GPP_C_18	HOSTSW_OWN_GPP_C_17	HOSTSW_OWN_GPP_C_16	HOSTSW_OWN_GPP_C_15	HOSTSW_OWN_GPP_C_14	HOSTSW_OWN_GPP_C_13	HOSTSW_OWN_GPP_C_12	HOSTSW_OWN_GPP_C_11	HOSTSW_OWN_GPP_C_10	HOSTSW_OWN_GPP_C_9	HOSTSW_OWN_GPP_C_8	HOSTSW_OWN_GPP_C_7	HOSTSW_OWN_GPP_C_6	HOSTSW_OWN_GPP_C_5	HOSTSW_OWN_GPP_C_4	HOSTSW_OWN_GPP_C_3	HOSTSW_OWN_GPP_C_2	HOSTSW_OWN_GPP_C_1	HOSTSW_OWN_GPP_C_0

Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_C_23):** Applied to GPP_C23. Same description as bit 0.
22	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_C_22):** Applied to GPP_C22. Same description as bit 0.
21	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_C_21):** Applied to GPP_C21. Same description as bit 0.
20	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_C_20):** Applied to GPP_C20. Same description as bit 0.
19	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_C_19):** Applied to GPP_C19. Same description as bit 0.
18	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_C_18):** Applied to GPP_C8. Same description as bit 0.
17	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_C_17):** Applied to GPP_C17. Same description as bit 0.
16	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_C_16):** Applied to GPP_C16. Same description as bit 0.
15	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_C_15):** Applied to GPP_C15. Same description as bit 0.
14	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_C_14):** Applied to GPP_C14. Same description as bit 0.
13	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_C_13):** Applied to GPP_C13. Same description as bit 0.
12	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_C_12):** Applied to GPP_C12. Same description as bit 0.
11	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_C_11):** Applied to GPP_C11. Same description as bit 0.
10	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_C_10):** Applied to GPP_C10. Same description as bit 0.
9	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_C_9):** Applied to GPP_C9. Same description as bit 0.
8	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_C_8):** Applied to GPP_C8. Same description as bit 0.
7	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_C_7):** Applied to GPP_C7. Same description as bit 0.
6	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_C_6):** Applied to GPP_C6. Same description as bit 0.
5	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_C_5):** Applied to GPP_C5. Same description as bit 0.
4	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_C_4):** Applied to GPP_C4. Same description as bit 0.
3	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_C_3):** Applied to GPP_C3. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
2	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_C_2):** Applied to GPP_C2. Same description as bit 0.
1	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_C_1):** Applied to GPP_C1. Same description as bit 0.
0	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_C_0):** This register determines the appropriate host status bit update when a pad is under host ownership. 0 = ACPI Mode. GPIO input event updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. GPI_STS update is masked. 1 = GPIO Driver Mode. GPIO input event updates are limited to GPI_STS. GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS updates are masked.

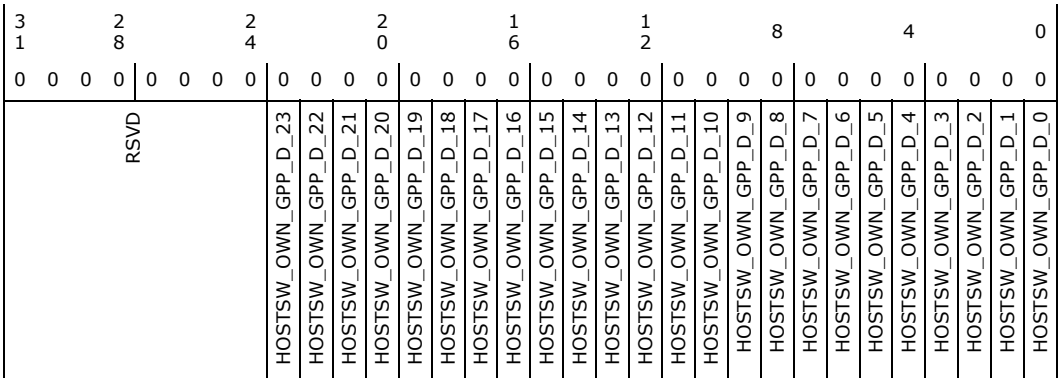
18.7.18 Host Software Pad Ownership (HOSTSW_OWN_GPP_D_0)—Offset 84h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_D_23):** Applied to GPP_D23. Same description as bit 0.
22	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_D_22):** Applied to GPP_D22. Same description as bit 0.
21	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_D_21):** Applied to GPP_D21. Same description as bit 0.
20	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_D_20):** Applied to GPP_D20. Same description as bit 0.
19	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_D_19):** Applied to GPP_D19. Same description as bit 0.
18	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_D_18):** Applied to GPP_D8. Same description as bit 0.
17	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_D_17):** Applied to GPP_D17. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
16	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_D_16):** Applied to GPP_D16. Same description as bit 0.
15	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_D_15):** Applied to GPP_D15. Same description as bit 0.
14	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_D_14):** Applied to GPP_D14. Same description as bit 0.
13	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_D_13):** Applied to GPP_D13. Same description as bit 0.
12	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_D_12):** Applied to GPP_D12. Same description as bit 0.
11	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_D_11):** Applied to GPP_D11. Same description as bit 0.
10	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_D_10):** Applied to GPP_D10. Same description as bit 0.
9	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_D_9):** Applied to GPP_D9. Same description as bit 0.
8	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_D_8):** Applied to GPP_D8. Same description as bit 0.
7	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_D_7):** Applied to GPP_D7. Same description as bit 0.
6	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_D_6):** Applied to GPP_D6. Same description as bit 0.
5	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_D_5):** Applied to GPP_D5. Same description as bit 0.
4	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_D_4):** Applied to GPP_D4. Same description as bit 0.
3	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_D_3):** Applied to GPP_D3. Same description as bit 0.
2	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_D_2):** Applied to GPP_D2. Same description as bit 0.
1	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_D_1):** Applied to GPP_D1. Same description as bit 0.
0	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_D_0):** This register determines the appropriate host status bit update when a pad is under host ownership. 0 = ACPI Mode. GPIO input event updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. GPI_STS update is masked. 1 = GPIO Driver Mode. GPIO input event updates are limited to GPI_STS. GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS updates are masked.

18.7.19 Host Software Pad Ownership (HOSTSW_OWN_GPP_E_0)—Offset 88h

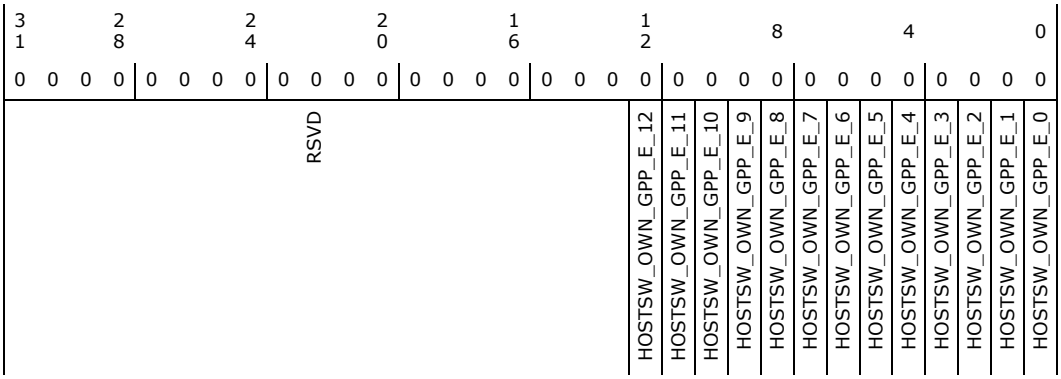
Same description as HOSTSW_OWN_GPP_C_0 register, except that this register applies to GPP_E group only.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:13	0h RO	Reserved
12	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_E_12):** Applied to GPP_E12. Same description as bit 0.
11	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_E_11):** Applied to GPP_E11. Same description as bit 0.
10	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_E_10):** Applied to GPP_E10. Same description as bit 0.
9	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_E_9):** Applied to GPP_E9. Same description as bit 0.
8	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_E_8):** Applied to GPP_E8. Same description as bit 0.
7	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_E_7):** Applied to GPP_E7. Same description as bit 0.
6	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_E_6):** Applied to GPP_E6. Same description as bit 0.
5	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_E_5):** Applied to GPP_E5. Same description as bit 0.
4	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_E_4):** Applied to GPP_E4. Same description as bit 0.
3	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_E_3):** Applied to GPP_E3. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
2	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_E_2):** Applied to GPP_E2. Same description as bit 0.
1	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_E_1):** Applied to GPP_E1. Same description as bit 0.
0	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_E_0):** This register determines the appropriate host status bit update when a pad is under host ownership. 0 = ACPI Mode. GPIO input event updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. GPI_STS update is masked. 1 = GPIO Driver Mode. GPIO input event updates are limited to GPI_STS. GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS updates are masked.

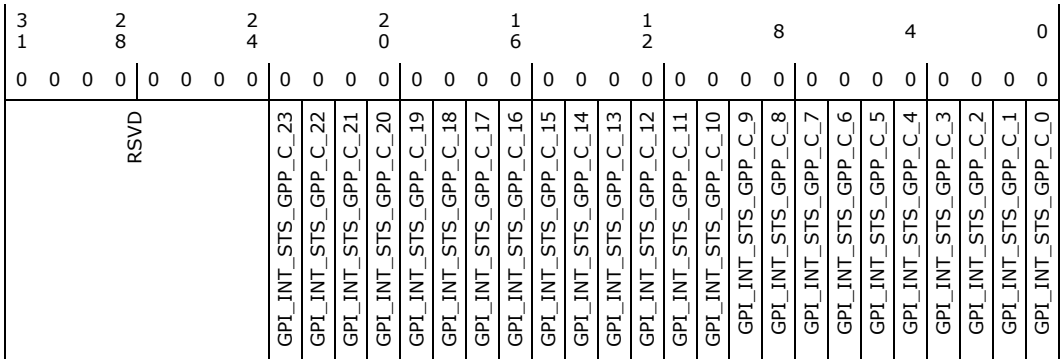
18.7.20 GPI Interrupt Status (GPI_IS_GPP_C_0)—Offset 100h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_C_23):** Applied to GPP_C23. Same description as bit 0.
22	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_C_22):** Applied to GPP_C22. Same description as bit 0.
21	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_C_21):** Applied to GPP_C21. Same description as bit 0.
20	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_C_20):** Applied to GPP_C20. Same description as bit 0.
19	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_C_19):** Applied to GPP_C19. Same description as bit 0.
18	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_C_18):** Applied to GPP_C18. Same description as bit 0.
17	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_C_17):** Applied to GPP_C17. Same description as bit 0.
16	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_C_16):** Applied to GPP_C16. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
15	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_C_15):** Applied to GPP_C15. Same description as bit 0.
14	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_C_14):** Applied to GPP_C14. Same description as bit 0.
13	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_C_13):** Applied to GPP_C13. Same description as bit 0.
12	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_C_12):** Applied to GPP_C12. Same description as bit 0.
11	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_C_11):** Applied to GPP_C11. Same description as bit 0.
10	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_C_10):** Applied to GPP_C10. Same description as bit 0.
9	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_C_9):** Applied to GPP_C9. Same description as bit 0.
8	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_C_8):** Applied to GPP_C8. Same description as bit 0.
7	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_C_7):** Applied to GPP_C7. Same description as bit 0.
6	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_C_6):** Applied to GPP_C6. Same description as bit 0.
5	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_C_5):** Applied to GPP_C5. Same description as bit 0.
4	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_C_4):** Applied to GPP_C4. Same description as bit 0.
3	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_C_3):** Applied to GPP_C3. Same description as bit 0.
2	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_C_2):** Applied to GPP_C2. Same description as bit 0.
1	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_C_1):** Applied to GPP_C1. Same description as bit 0.
0	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_C_0):** This bit is set to '1' by hardware when either an edge or a level event is detected (See RxEdCfg RxInv) on pad and all the following conditions are true: - The corresponding pad is used in GPIO input mode. - HOSTSW_OWN = 1 (i.e., Host GPIO Driver Mode). Writing a value of 1 will clear the bit while writing a value of 0 has no effect. 0 = No interrupt 1 = Interrupt asserts The state of GPI_INT_EN[x] does not prevent the setting of GPI_INT_STS[x].

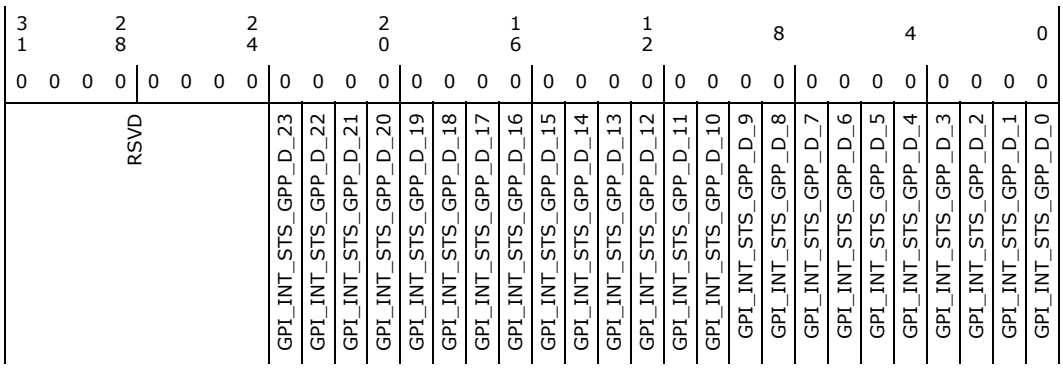
18.7.21 GPI Interrupt Status (GPI_IS_GPP_D_0)—Offset 104h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_D_23):** Applied to GPP_D23. Same description as bit 0.
22	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_D_22):** Applied to GPP_D22. Same description as bit 0.
21	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_D_21):** Applied to GPP_D21. Same description as bit 0.
20	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_D_20):** Applied to GPP_D20. Same description as bit 0.
19	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_D_19):** Applied to GPP_D19. Same description as bit 0.
18	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_D_18):** Applied to GPP_D18. Same description as bit 0.
17	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_D_17):** Applied to GPP_D17. Same description as bit 0.
16	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_D_16):** Applied to GPP_D16. Same description as bit 0.
15	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_D_15):** Applied to GPP_D15. Same description as bit 0.
14	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_D_14):** Applied to GPP_D14. Same description as bit 0.
13	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_D_13):** Applied to GPP_D13. Same description as bit 0.
12	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_D_12):** Applied to GPP_D12. Same description as bit 0.
11	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_D_11):** Applied to GPP_D11. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
10	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_D_10):** Applied to GPP_D10. Same description as bit 0.
9	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_D_9):** Applied to GPP_D9. Same description as bit 0.
8	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_D_8):** Applied to GPP_D8. Same description as bit 0.
7	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_D_7):** Applied to GPP_D7. Same description as bit 0.
6	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_D_6):** Applied to GPP_D6. Same description as bit 0.
5	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_D_5):** Applied to GPP_D5. Same description as bit 0.
4	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_D_4):** Applied to GPP_D4. Same description as bit 0.
3	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_D_3):** Applied to GPP_D3. Same description as bit 0.
2	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_D_2):** Applied to GPP_D2. Same description as bit 0.
1	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_D_1):** Applied to GPP_D1. Same description as bit 0.
0	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_D_0):** This bit is set to '1' by hardware when either an edge or a level event is detected (See RxEdCfg RxInv) on pad and all the following conditions are true: - The corresponding pad is used in GPIO input mode. - HOSTSW_OWN = 1 (i.e., Host GPIO Driver Mode). Writing a value of 1 will clear the bit while writing a value of 0 has no effect. 0 = No interrupt 1 = Interrupt asserts The state of GPI_INT_EN[x] does not prevent the setting of GPI_INT_STS[x].

18.7.22 GPI Interrupt Status (GPI_IS_GPP_E_0)—Offset 108h

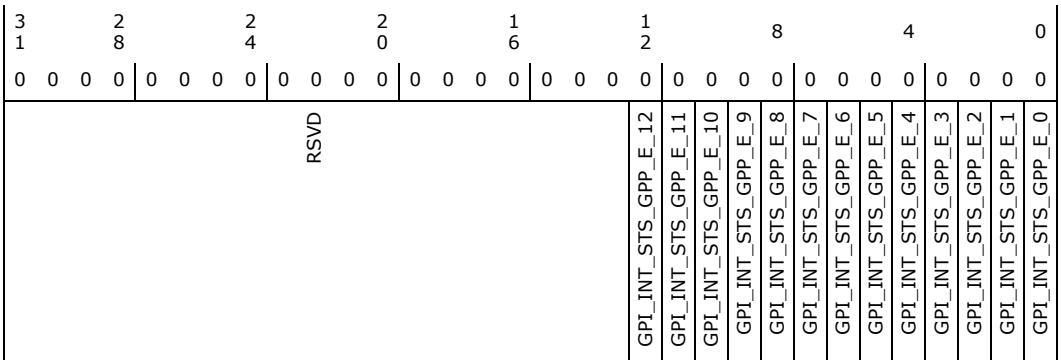
Same description as GPI_IS_GPP_C_0 register, except that this register applies to GPP_E group only.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:13	0h RO	Reserved
12	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_E_12):** Applied to GPP_E12. Same description as bit 0.
11	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_E_11):** Applied to GPP_E11. Same description as bit 0.
10	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_E_10):** Applied to GPP_E10. Same description as bit 0.
9	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_E_9):** Applied to GPP_E9. Same description as bit 0.
8	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_E_8):** Applied to GPP_E8. Same description as bit 0.
7	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_E_7):** Applied to GPP_E7. Same description as bit 0.
6	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_E_6):** Applied to GPP_E6. Same description as bit 0.
5	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_E_5):** Applied to GPP_E5. Same description as bit 0.
4	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_E_4):** Applied to GPP_E4. Same description as bit 0.
3	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_E_3):** Applied to GPP_E3. Same description as bit 0.
2	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_E_2):** Applied to GPP_E2. Same description as bit 0.
1	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_E_1):** Applied to GPP_E1. Same description as bit 0.
0	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_D_0):** This bit is set to '1' by hardware when either an edge or a level event is detected (See RxEdCfg RxInv) on pad and all the following conditions are true: - The corresponding pad is used in GPIO input mode. - HOSTSW_OWN = 1 (i.e., Host GPIO Driver Mode). Writing a value of 1 will clear the bit while writing a value of 0 has no effect. 0 = No interrupt 1 = Interrupt asserts The state of GPI_INT_EN[x] does not prevent the setting of GPI_INT_STS[x].

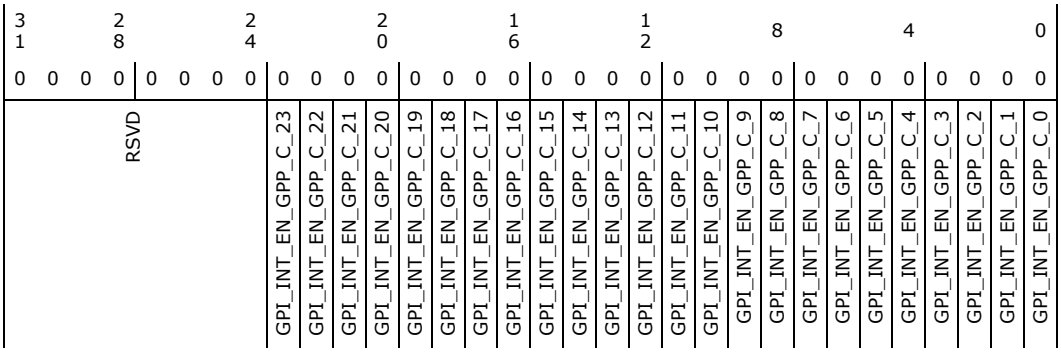
18.7.23 GPI Interrupt Enable (GPI_IE_GPP_C_0)—Offset 110h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_C_23):** Applied to GPP_C23. Same description as bit 0.
22	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_C_22):** Applied to GPP_C22. Same description as bit 0.
21	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_C_21):** Applied to GPP_C21. Same description as bit 0.
20	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_C_20):** Applied to GPP_C20. Same description as bit 0.
19	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_C_19):** Applied to GPP_C19. Same description as bit 0.
18	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_C_18):** Applied to GPP_C18. Same description as bit 0.
17	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_C_17):** Applied to GPP_C17. Same description as bit 0.
16	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_C_16):** Applied to GPP_C16. Same description as bit 0.
15	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_C_15):** Applied to GPP_C15. Same description as bit 0.
14	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_C_14):** Applied to GPP_C14. Same description as bit 0.
13	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_C_13):** Applied to GPP_C13. Same description as bit 0.
12	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_C_12):** Applied to GPP_C12. Same description as bit 0.
11	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_C_11):** Applied to GPP_C11. Same description as bit 0.
10	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_C_10):** Applied to GPP_C10. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
9	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_C_9):** Applied to GPP_C9. Same description as bit 0.
8	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_C_8):** Applied to GPP_C8. Same description as bit 0.
7	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_C_7):** Applied to GPP_C7. Same description as bit 0.
6	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_C_6):** Applied to GPP_C6. Same description as bit 0.
5	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_C_5):** Applied to GPP_C5. Same description as bit 0.
4	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_C_4):** Applied to GPP_C4. Same description as bit 0.
3	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_C_3):** Applied to GPP_C3. Same description as bit 0.
2	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_C_2):** Applied to GPP_C2. Same description as bit 0.
1	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_C_1):** Applied to GPP_C1. Same description as bit 0.
0	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_C_0):** This bit is used to enable/ disable the generation of APIC interrupt when the corresponding GPI_INT_STS bit is set. 0 = disable interrupt generation 1 = enable interrupt generation Refer to GPI_IRQ_ROUTE for host GPIO Driver Mode interrupt routing.

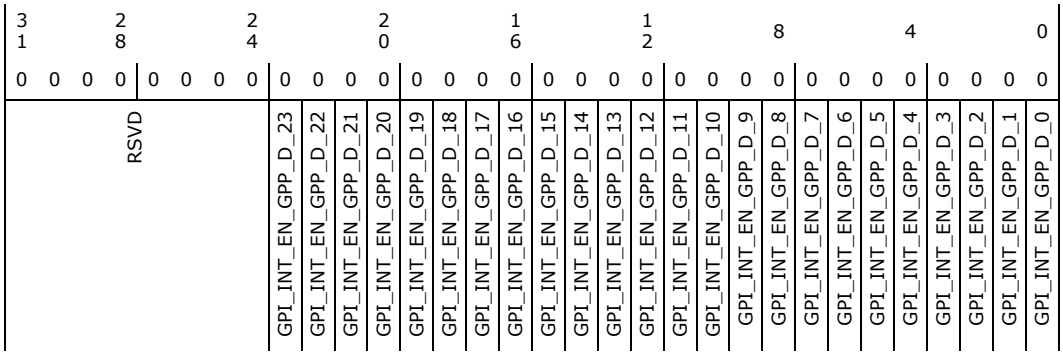
18.7.24 GPI Interrupt Enable (GPI_IE_GPP_D_0)—Offset 114h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_D_23):** Applied to GPP_D23. Same description as bit 0.
22	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_D_22):** Applied to GPP_D22. Same description as bit 0.
21	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_D_21):** Applied to GPP_D21. Same description as bit 0.
20	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_D_20):** Applied to GPP_D20. Same description as bit 0.
19	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_D_19):** Applied to GPP_D19. Same description as bit 0.
18	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_D_18):** Applied to GPP_D18. Same description as bit 0.
17	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_D_17):** Applied to GPP_D17. Same description as bit 0.
16	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_D_16):** Applied to GPP_D16. Same description as bit 0.
15	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_D_15):** Applied to GPP_D15. Same description as bit 0.
14	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_D_14):** Applied to GPP_D14. Same description as bit 0.
13	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_D_13):** Applied to GPP_D13. Same description as bit 0.
12	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_D_12):** Applied to GPP_D12. Same description as bit 0.
11	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_D_11):** Applied to GPP_D11. Same description as bit 0.
10	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_D_10):** Applied to GPP_D10. Same description as bit 0.
9	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_D_9):** Applied to GPP_D9. Same description as bit 0.
8	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_D_8):** Applied to GPP_D8. Same description as bit 0.
7	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_D_7):** Applied to GPP_D7. Same description as bit 0.
6	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_D_6):** Applied to GPP_D6. Same description as bit 0.
5	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_D_5):** Applied to GPP_D5. Same description as bit 0.
4	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_D_4):** Applied to GPP_D4. Same description as bit 0.
3	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_D_3):** Applied to GPP_D3. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
2	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_D_2):** Applied to GPP_D2. Same description as bit 0.
1	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_D_1):** Applied to GPP_D1. Same description as bit 0.
0	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_D_0):** This bit is used to enable/ disable the generation of APIC interrupt when the corresponding GPI_INT_STS bit is set. 0 = disable interrupt generation 1 = enable interrupt generation Refer to GPI_IRQ_ROUTE for host GPIO Driver Mode interrupt routing.

18.7.25 GPI Interrupt Enable (GPI_IE_GPP_E_0)—Offset 118h

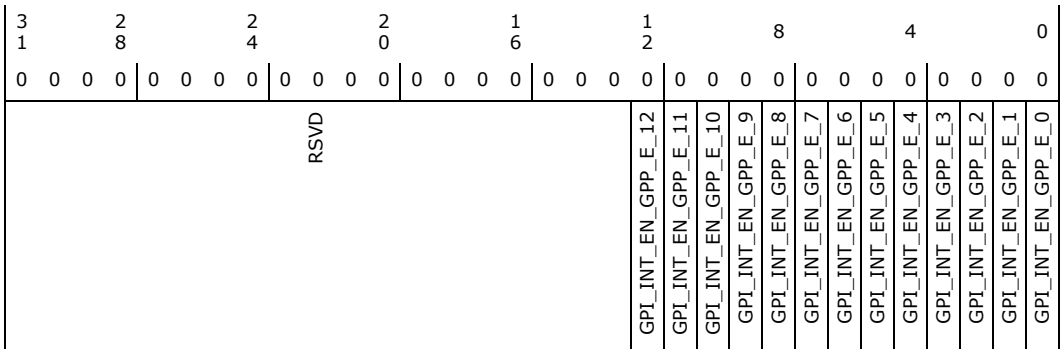
Same description as GPI_IE_GPP_C_0 register, except that this register is for GPP_E group only.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:13	0h RO	Reserved
12	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_E_12):** Applied to GPP_E12. Same description as bit 0.
11	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_E_11):** Applied to GPP_E11. Same description as bit 0.
10	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_E_10):** Applied to GPP_E10. Same description as bit 0.
9	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_E_9):** Applied to GPP_E9. Same description as bit 0.
8	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_E_8):** Applied to GPP_E8. Same description as bit 0.
7	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_E_7):** Applied to GPP_E7. Same description as bit 0.
6	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_E_6):** Applied to GPP_E6. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
5	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_E_5):** Applied to GPP_E5. Same description as bit 0.
4	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_E_4):** Applied to GPP_E4. Same description as bit 0.
3	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_E_3):** Applied to GPP_E3. Same description as bit 0.
2	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_E_2):** Applied to GPP_E2. Same description as bit 0.
1	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_E_1):** Applied to GPP_E1. Same description as bit 0.
0	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_E_0):** This bit is used to enable/ disable the generation of APIC interrupt when the corresponding GPI_INT_STS bit is set. 0 = disable interrupt generation 1 = enable interrupt generation Refer to GPI_IRQ_ROUTE for host GPIO Driver Mode interrupt routing.

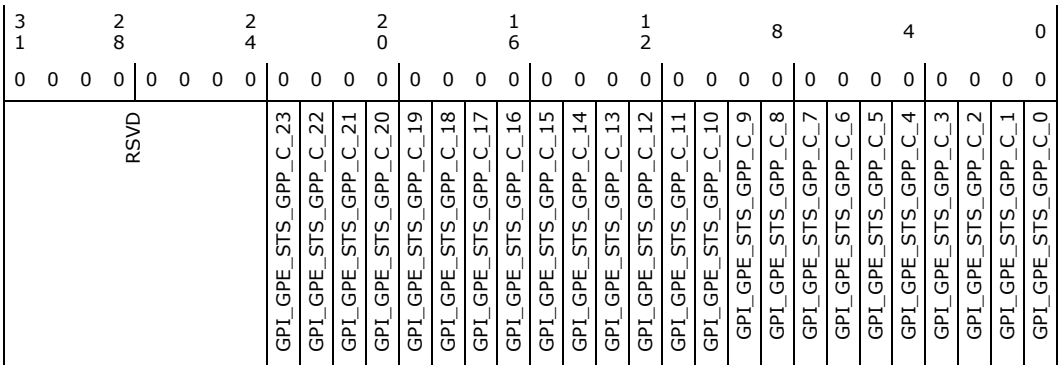
18.7.26 GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_0)—Offset 120h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_23):** Applied to GPP_C23. Same description as bit 0.
22	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_22):** Applied to GPP_C22. Same description as bit 0.
21	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_21):** Applied to GPP_C21. Same description as bit 0.
20	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_20):** Applied to GPP_C20. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
19	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_19):** Applied to GPP_C19. Same description as bit 0.
18	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_18):** Applied to GPP_C18. Same description as bit 0.
17	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_17):** Applied to GPP_C17. Same description as bit 0.
16	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_16):** Applied to GPP_C16. Same description as bit 0.
15	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_15):** Applied to GPP_C15. Same description as bit 0.
14	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_14):** Applied to GPP_C14. Same description as bit 0.
13	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_13):** Applied to GPP_C13. Same description as bit 0.
12	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_12):** Applied to GPP_C12. Same description as bit 0.
11	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_11):** Applied to GPP_C11. Same description as bit 0.
10	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_10):** Applied to GPP_C10. Same description as bit 0.
9	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_9):** Applied to GPP_C9. Same description as bit 0.
8	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_8):** Applied to GPP_C8. Same description as bit 0.
7	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_7):** Applied to GPP_C7. Same description as bit 0.
6	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_6):** Applied to GPP_C6. Same description as bit 0.
5	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_5):** Applied to GPP_C5. Same description as bit 0.
4	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_4):** Applied to GPP_C4. Same description as bit 0.
3	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_3):** Applied to GPP_C3. Same description as bit 0.
2	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_2):** Applied to GPP_C2. Same description as bit 0.
1	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_1):** Applied to GPP_C1. Same description as bit 0.
0	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_C_0):** These bits are set any time the corresponding GPIO pad is set up as an input, under host ownership and the corresponding GPIO signal is high (or low if the corresponding RXINV bit is set). If the corresponding enable bit is set in the GPI_GPE_EN register, then when the GPI_GPE_STS bit is set: - If the system is in an S3-S5 state, the event will also wake the system. - If the system is in an S0 state (or upon waking back to an S0 state), an SCI will be caused, depending on the GPIRoutSCI bit for the corresponding pad. These bits are sticky bits and are cleared by writing a 1 back to this bit position. The state of GPI_GPE_EN does not prevent the setting of GPI_GPE_STS.

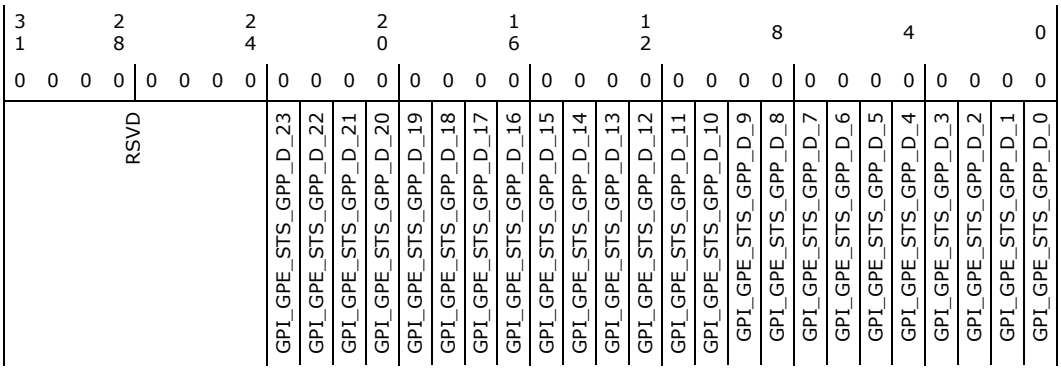
18.7.27 GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_0)—Offset 124h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_23):** Applied to GPP_D23. Same description as bit 0.
22	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_22):** Applied to GPP_D22. Same description as bit 0.
21	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_21):** Applied to GPP_D21. Same description as bit 0.
20	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_20):** Applied to GPP_D20. Same description as bit 0.
19	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_19):** Applied to GPP_D19. Same description as bit 0.
18	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_18):** Applied to GPP_D18. Same description as bit 0.
17	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_17):** Applied to GPP_D17. Same description as bit 0.
16	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_16):** Applied to GPP_D16. Same description as bit 0.
15	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_15):** Applied to GPP_D15. Same description as bit 0.
14	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_14):** Applied to GPP_D14. Same description as bit 0.
13	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_13):** Applied to GPP_D13. Same description as bit 0.
12	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_12):** Applied to GPP_D12. Same description as bit 0.
11	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_11):** Applied to GPP_D11. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
10	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_10):** Applied to GPP_D10. Same description as bit 0.
9	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_9):** Applied to GPP_D9. Same description as bit 0.
8	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_8):** Applied to GPP_D8. Same description as bit 0.
7	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_7):** Applied to GPP_D7. Same description as bit 0.
6	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_6):** Applied to GPP_D6. Same description as bit 0.
5	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_5):** Applied to GPP_D5. Same description as bit 0.
4	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_4):** Applied to GPP_D4. Same description as bit 0.
3	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_3):** Applied to GPP_D3. Same description as bit 0.
2	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_2):** Applied to GPP_D2. Same description as bit 0.
1	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_1):** Applied to GPP_D1. Same description as bit 0.
0	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_D_0):** These bits are set any time the corresponding GPIO pad is set up as an input, under host ownership and the corresponding GPIO signal is high (or low if the corresponding RXINV bit is set). If the corresponding enable bit is set in the GPI_GPE_EN register, then when the GPI_GPE_STS bit is set: - If the system is in an S3-S5 state, the event will also wake the system. - If the system is in an S0 state (or upon waking back to an S0 state), an SCI will be caused, depending on the GPIRoutSCI bit for the corresponding pad. These bits are sticky bits and are cleared by writing a 1 back to this bit position. The state of GPI_GPE_EN does not prevent the setting of GPI_GPE_STS.

18.7.28 GPI General Purpose Events Status (GPI_GPE_STS_GPP_E_0)—Offset 128h

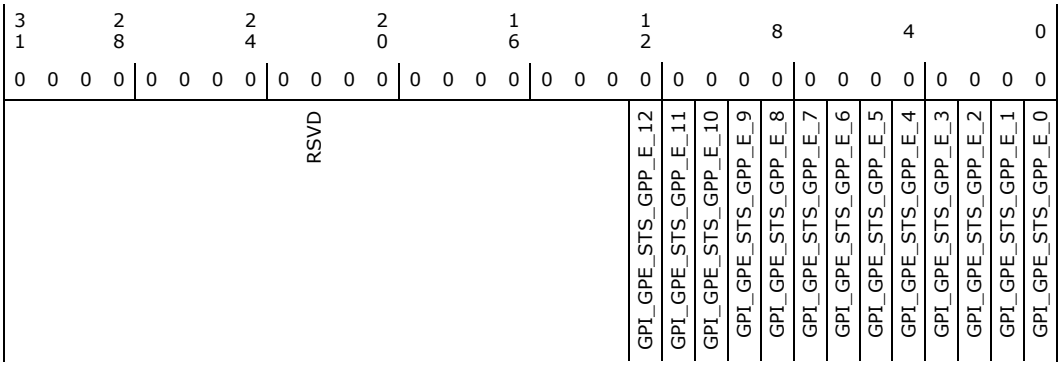
Same description as GPI_GPE_STS_GPP_C_0 register, except that this register is for GPP_E group only.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:13	0h RO	Reserved
12	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_E_12):** Applied to GPP_E12. Same description as bit 0.
11	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_E_11):** Applied to GPP_E11. Same description as bit 0.
10	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_E_10):** Applied to GPP_E10. Same description as bit 0.
9	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_E_9):** Applied to GPP_E9. Same description as bit 0.
8	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_E_8):** Applied to GPP_E8. Same description as bit 0.
7	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_E_7):** Applied to GPP_E7. Same description as bit 0.
6	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_E_6):** Applied to GPP_E6. Same description as bit 0.
5	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_E_5):** Applied to GPP_E5. Same description as bit 0.
4	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_E_4):** Applied to GPP_E4. Same description as bit 0.
3	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_E_3):** Applied to GPP_E3. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
2	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_E_2):** Applied to GPP_E2. Same description as bit 0.
1	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_E_1):** Applied to GPP_E1. Same description as bit 0.
0	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_E_0):** These bits are set any time the corresponding GPIO pad is set up as an input, under host ownership and the corresponding GPIO signal is high (or low if the corresponding RXINV bit is set). If the corresponding enable bit is set in the GPI_GPE_EN register, then when the GPI_GPE_STS bit is set: - If the system is in an S3-S5 state, the event will also wake the system. - If the system is in an S0 state (or upon waking back to an S0 state), an SCI will be caused, depending on the GPIRoutSCI bit for the corresponding pad. These bits are sticky bits and are cleared by writing a 1 back to this bit position. The state of GPI_GPE_EN does not prevent the setting of GPI_GPE_STS.

18.7.29 GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_0)—Offset 130h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD								GPI_GPE_EN_GPP_C_23	GPI_GPE_EN_GPP_C_22	GPI_GPE_EN_GPP_C_21	GPI_GPE_EN_GPP_C_20	GPI_GPE_EN_GPP_C_19	GPI_GPE_EN_GPP_C_18	GPI_GPE_EN_GPP_C_17	GPI_GPE_EN_GPP_C_16	GPI_GPE_EN_GPP_C_15	GPI_GPE_EN_GPP_C_14	GPI_GPE_EN_GPP_C_13	GPI_GPE_EN_GPP_C_12	GPI_GPE_EN_GPP_C_11	GPI_GPE_EN_GPP_C_10	GPI_GPE_EN_GPP_C_9	GPI_GPE_EN_GPP_C_8	GPI_GPE_EN_GPP_C_7	GPI_GPE_EN_GPP_C_6	GPI_GPE_EN_GPP_C_5	GPI_GPE_EN_GPP_C_4	GPI_GPE_EN_GPP_C_3	GPI_GPE_EN_GPP_C_2	GPI_GPE_EN_GPP_C_1	GPI_GPE_EN_GPP_C_0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_23):** Applied to GPP_C23. Same description as bit 0.
22	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_22):** Applied to GPP_C22. Same description as bit 0.
21	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_21):** Applied to GPP_C21. Same description as bit 0.
20	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_20):** Applied to GPP_C20. Same description as bit 0.
19	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_19):** Applied to GPP_C19. Same description as bit 0.
18	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_18):** Applied to GPP_C18. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_17):** Applied to GPP_C17. Same description as bit 0.
16	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_16):** Applied to GPP_C16. Same description as bit 0.
15	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_15):** Applied to GPP_C15. Same description as bit 0.
14	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_14):** Applied to GPP_C14. Same description as bit 0.
13	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_13):** Applied to GPP_C13. Same description as bit 0.
12	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_12):** Applied to GPP_C12. Same description as bit 0.
11	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_11):** Applied to GPP_C11. Same description as bit 0.
10	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_10):** Applied to GPP_C10. Same description as bit 0.
9	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_9):** Applied to GPP_C9. Same description as bit 0.
8	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_8):** Applied to GPP_C8. Same description as bit 0.
7	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_7):** Applied to GPP_C7. Same description as bit 0.
6	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_6):** Applied to GPP_C6. Same description as bit 0.
5	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_5):** Applied to GPP_C5. Same description as bit 0.
4	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_4):** Applied to GPP_C4. Same description as bit 0.
3	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_3):** Applied to GPP_C3. Same description as bit 0.
2	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_2):** Applied to GPP_C2. Same description as bit 0.
1	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_1):** Applied to GPP_C1. Same description as bit 0.
0	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_C_0):** This bit is used to enable/disable the generation of GPE to cause SCI and/or wake when the corresponding GPI_GPE_STS bit is set. 0 = disable GPE generation 1 = enable GPE generation ***Note:*** The pad must also be routed for GPE functionality in order for GPE to be generated, i.e. the corresponding GPIRoutSCI must be set to '1'.

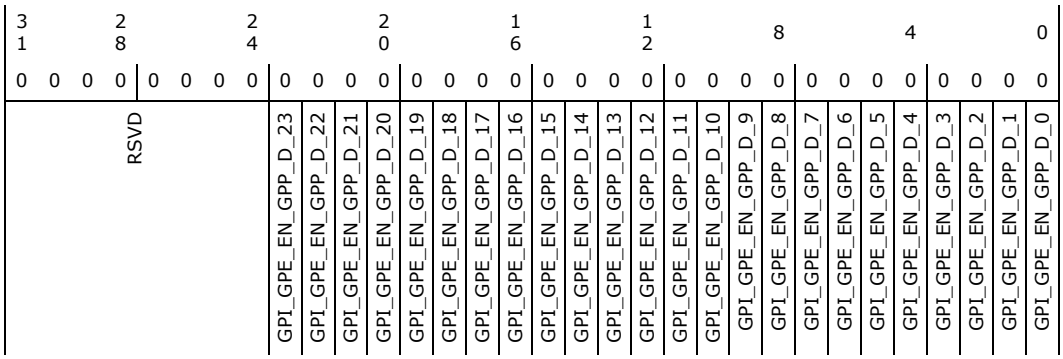
18.7.30 GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_0)—Offset 134h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_23):** Applied to GPP_D23. Same description as bit 0.
22	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_22):** Applied to GPP_D22. Same description as bit 0.
21	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_21):** Applied to GPP_D21. Same description as bit 0.
20	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_20):** Applied to GPP_D20. Same description as bit 0.
19	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_19):** Applied to GPP_D19. Same description as bit 0.
18	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_18):** Applied to GPP_D18. Same description as bit 0.
17	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_17):** Applied to GPP_D17. Same description as bit 0.
16	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_16):** Applied to GPP_D16. Same description as bit 0.
15	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_15):** Applied to GPP_D15. Same description as bit 0.
14	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_14):** Applied to GPP_D14. Same description as bit 0.
13	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_13):** Applied to GPP_D13. Same description as bit 0.
12	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_12):** Applied to GPP_D12. Same description as bit 0.
11	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_11):** Applied to GPP_D11. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
10	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_10):** Applied to GPP_D10. Same description as bit 0.
9	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_9):** Applied to GPP_D9. Same description as bit 0.
8	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_8):** Applied to GPP_D8. Same description as bit 0.
7	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_7):** Applied to GPP_D7. Same description as bit 0.
6	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_6):** Applied to GPP_D6. Same description as bit 0.
5	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_5):** Applied to GPP_D5. Same description as bit 0.
4	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_4):** Applied to GPP_D4. Same description as bit 0.
3	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_3):** Applied to GPP_D3. Same description as bit 0.
2	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_2):** Applied to GPP_D2. Same description as bit 0.
1	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_1):** Applied to GPP_D1. Same description as bit 0.
0	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_D_0):** This bit is used to enable/disable the generation of GPE to cause SCI and/or wake when the corresponding GPI_GPE_STS bit is set. 0 = disable GPE generation 1 = enable GPE generation ***Note:*** The pad must also be routed for GPE functionality in order for GPE to be generated, i.e. the corresponding GPIRoutSCI must be set to '1'.

18.7.31 GPI General Purpose Events Enable (GPI_GPE_EN_GPP_E_0)—Offset 138h

Same description as GPI_GPE_EN_GPP_C_0 register, except that this register is for GPP_E group only.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																				GPI_GPE_EN_GPP_E_12	GPI_GPE_EN_GPP_E_11	GPI_GPE_EN_GPP_E_10	GPI_GPE_EN_GPP_E_9	GPI_GPE_EN_GPP_E_8	GPI_GPE_EN_GPP_E_7	GPI_GPE_EN_GPP_E_6	GPI_GPE_EN_GPP_E_5	GPI_GPE_EN_GPP_E_4	GPI_GPE_EN_GPP_E_3	GPI_GPE_EN_GPP_E_2	GPI_GPE_EN_GPP_E_1	GPI_GPE_EN_GPP_E_0

Bit Range	Default & Access	Field Name (ID): Description
31:13	0h RO	Reserved
12	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_E_12):** Applied to GPP_E12. Same description as bit 0.
11	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_E_11):** Applied to GPP_E11. Same description as bit 0.
10	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_E_10):** Applied to GPP_E10. Same description as bit 0.
9	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_E_9):** Applied to GPP_E9. Same description as bit 0.
8	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_E_8):** Applied to GPP_E8. Same description as bit 0.
7	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_E_7):** Applied to GPP_E7. Same description as bit 0.
6	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_E_6):** Applied to GPP_E6. Same description as bit 0.
5	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_E_5):** Applied to GPP_E5. Same description as bit 0.
4	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_E_4):** Applied to GPP_E4. Same description as bit 0.
3	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_E_3):** Applied to GPP_E3. Same description as bit 0.
2	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_E_2):** Applied to GPP_E2. Same description as bit 0.
1	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_E_1):** Applied to GPP_E1. Same description as bit 0.
0	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_E_0):** This bit is used to enable/disable the generation of GPE to cause SCI and/or wake when the corresponding GPI_GPE_STS bit is set. 0 = disable GPE generation 1 = enable GPE generation ***Note:*** The pad must also be routed for GPE functionality in order for GPE to be generated, i.e., the corresponding GPIRoutSCI must be set to '1'.

18.7.32 SMI Status (GPI_SMI_STS_GPP_C_0)—Offset 140h

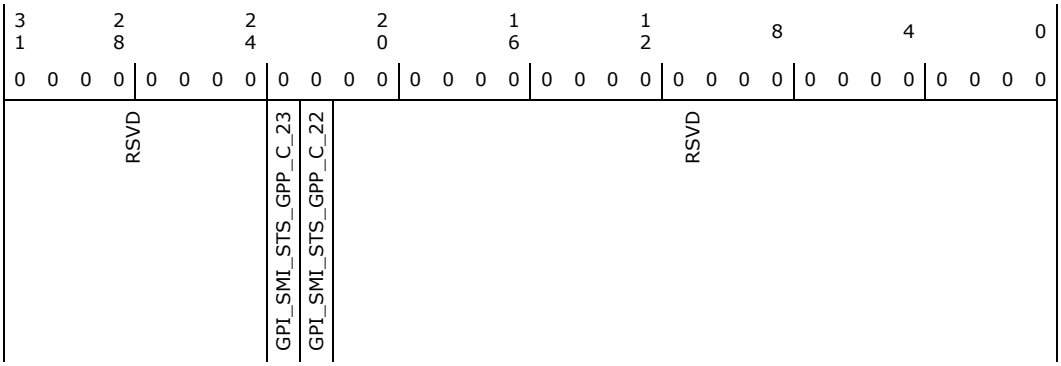
Register bits in this register are implemented for GPP_C signals that have SMI capability only. Other bits are reserved and RO.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW1C	**GPI SMI Status (GPI_SMI_STS_GPP_C_23):** Same description as bit 22.
22	0h RW1C	**GPI SMI Status (GPI_SMI_STS_GPP_C_22):** This bit is set to 1 by hardware when a level event (See RxEdCfg,RxInv) is detected, and all the following conditions are true: - The corresponding pad is used in GPIO input mode - The corresponding PAD_OWN[2:0] is '000' (i.e. ACPI GPIO Mode). If the following conditions are true, then an SMI will be generated if the GPI_SMI_STS bit is set: 1. The corresponding bit in the GPI_SMI_EN register is set 2. The corresponding pad's GPIROUTSMI is set Writing a value of 1 will clear the bit while writing a value of 0 has no effect. 0 = There is no SMI event 1 = There is an SMI event The state of GPI_SMI_EN does not prevent the setting of GPI_SMI_STS. Defaults for these bits are dependent on the state of the GPI pads.
21:0	0h RO	Reserved.

18.7.33 SMI Status (GPI_SMI_STS_GPP_D_0)—Offset 144h

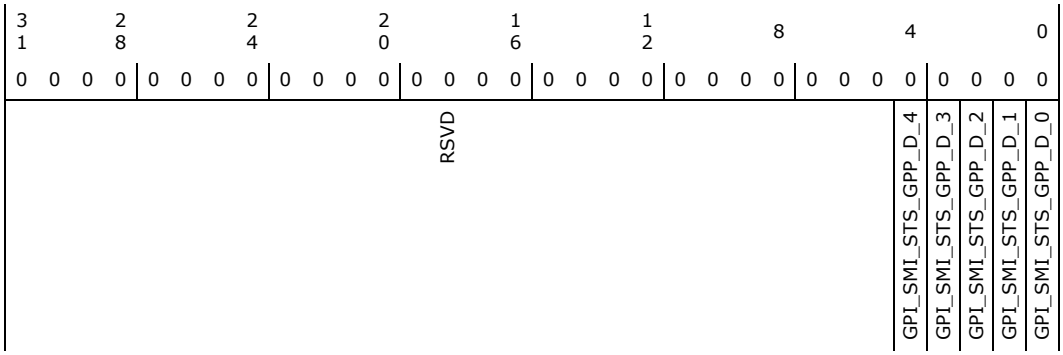
Register bits in this register are implemented for GPP_D signals that have SMI capability only. Other bits are reserved and RO.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:5	0h RO	Reserved
4	0h RW1C	**GPI SMI Status (GPI_SMI_STS_GPP_D_4):** Same description as bit 0.
3	0h RW1C	**GPI SMI Status (GPI_SMI_STS_GPP_D_3):** Same description as bit 0.
2	0h RW1C	**GPI SMI Status (GPI_SMI_STS_GPP_D_2):** Same description as bit 0.
1	0h RW1C	**GPI SMI Status (GPI_SMI_STS_GPP_D_1):** Same description as bit 0.
0	0h RW1C	**GPI SMI Status (GPI_SMI_STS_GPP_D_0):** This bit is set to 1 by hardware when a level event (See RxEdCfg,RxInv) is detected, and all the following conditions are true: - The corresponding pad is used in GPIO input mode. - The corresponding PAD_OWN[2:0] is '000' (i.e. ACPI GPIO Mode). If the following conditions are true, then an SMI will be generated if the GPI_SMI_STS bit is set: 1. The corresponding bit in the GPI_SMI_EN register is set. 2. The corresponding pad's GPIROUTSMI is set. Writing a value of 1 will clear the bit while writing a value of 0 has no effect. 0 = There is no SMI event. 1 = There is an SMI event. The state of GPI_SMI_EN does not prevent the setting of GPI_SMI_STS. Defaults for these bits are dependent on the state of the GPI pads.

18.7.34 SMI Status (GPI_SMI_STS_GPP_E_0)—Offset 148h

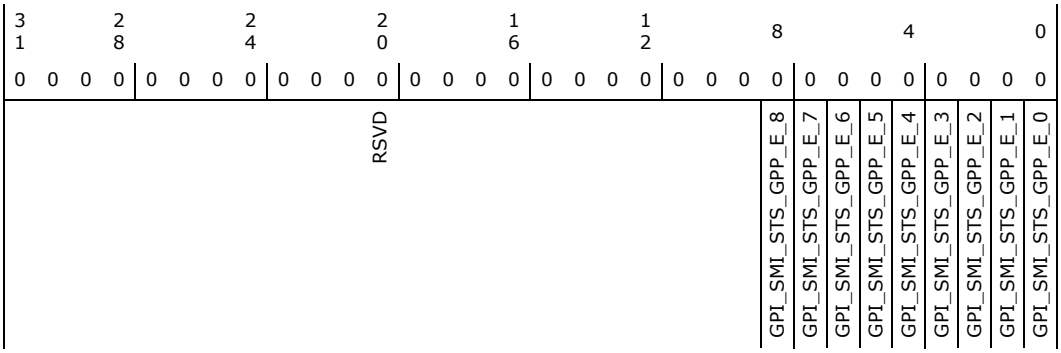
Register bits in this register are implemented for GPP_E signals that have SMI capability only. Other bits are reserved and RO.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:9	0h RO	Reserved
8	0h RW1C	**GPI SMI Status (GPI_SMI_STS_GPP_E_8):** Same description as bit 0.
7	0h RW1C	**GPI SMI Status (GPI_SMI_STS_GPP_E_7):** Same description as bit 0.
6	0h RW1C	**GPI SMI Status (GPI_SMI_STS_GPP_E_6):** Same description as bit 0.
5	0h RW1C	**GPI SMI Status (GPI_SMI_STS_GPP_E_5):** Same description as bit 0.
4	0h RW1C	**GPI SMI Status (GPI_SMI_STS_GPP_E_4):** Same description as bit 0.
3	0h RW1C	**GPI SMI Status (GPI_SMI_STS_GPP_E_3):** Same description as bit 0.
2	0h RW1C	**GPI SMI Status (GPI_SMI_STS_GPP_E_2):** Same description as bit 0.
1	0h RW1C	**GPI SMI Status (GPI_SMI_STS_GPP_E_1):** Same description as bit 0.
0	0h RW1C	**GPI SMI Status (GPI_SMI_STS_GPP_E_0):** This bit is set to 1 by hardware when a level event (See RxEdCfg,RxInv) is detected, and all the following conditions are true: - The corresponding pad is used in GPIO input mode. - The corresponding PAD_OWN[2:0] is '000' (i.e., ACPI GPIO Mode). If the following conditions are true, then an SMI will be generated if the GPI_SMI_STS bit is set: 1. The corresponding bit in the GPI_SMI_EN register is set. 2. The corresponding pad's GPIROUTSMI is set. Writing a value of 1 will clear the bit while writing a value of 0 has no effect. 0 = There is no SMI event. 1 = There is an SMI event. The state of GPI_SMI_EN does not prevent the setting of GPI_SMI_STS. Defaults for these bits are dependent on the state of the GPI pads.

18.7.35 SMI Enable (GPI_SMI_EN_GPP_C_0)—Offset 150h

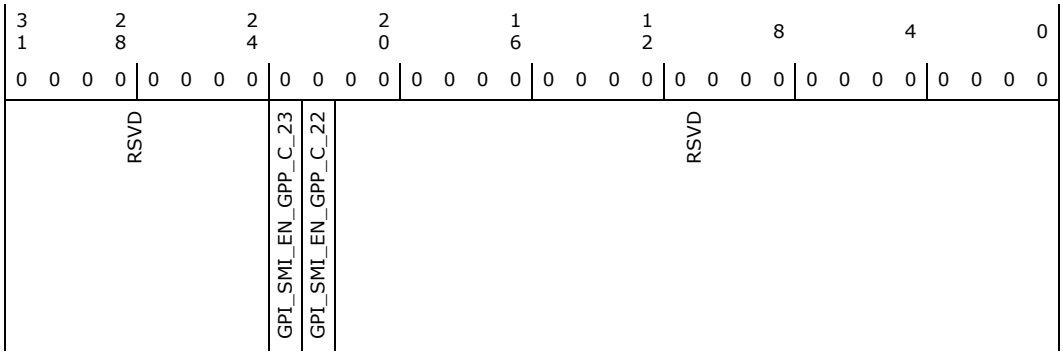
Register bits in this register are implemented for GPP_C signals that have SMI capability only. Other bits are reserved and RO.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**GPI SMI Enable (GPI_SMI_EN_GPP_C_23):** Same description as bit 22.
22	0h RW	**GPI SMI Enable (GPI_SMI_EN_GPP_C_22):** This bit is used to enable/disable the generation of SMI when the corresponding GPI_SMI_STS bit is set. The pad must also be routed for SMI functionality in order for SMI to be generated i.e., the corresponding GPIROUTSMI must be set to 1. 0 = disable SMI generation 1 = enable SMI generation ***Note:*** Each of the bit is lock-able dependent on the associated Pad Configuration register PadCfgLock setting. E.g. if the PadCfgLock of pad0 is 1, bit0 of this bit is locked down to read-only.
21:0	0h RO	Reserved

18.7.36 SMI Enable (GPI_SMI_EN_GPP_D_0)—Offset 154h

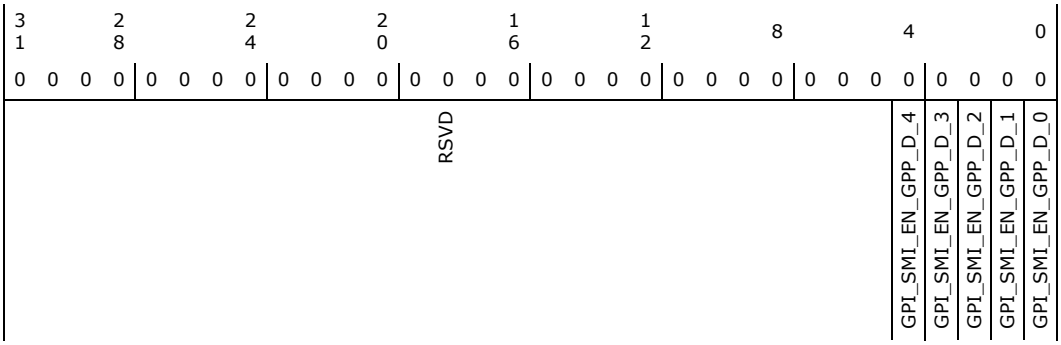
Register bits in this register are implemented for GPP_D signals that have SMI capability only. Other bits are reserved and RO.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:5	0h RO	Reserved
4	0h RW	**GPI SMI Enable (GPI_SMI_EN_GPP_D_4):** Same description as bit 0.
3	0h RW	**GPI SMI Enable (GPI_SMI_EN_GPP_D_3):** Same description as bit 0.
2	0h RW	**GPI SMI Enable (GPI_SMI_EN_GPP_D_2):** Same description as bit 0.
1	0h RW	**GPI SMI Enable (GPI_SMI_EN_GPP_D_1):** Same description as bit 0.
0	0h RW	**GPI SMI Enable (GPI_SMI_EN_GPP_D_0):** This bit is used to enable/disable the generation of SMI when the corresponding GPI_SMI_STS bit is set. The pad must also be routed for SMI functionality in order for SMI to be generated, i.e. the corresponding GPIROUTSMI must be set to 1. 0 = disable SMI generation 1 = enable SMI generation ***Note:*** Each of the bit is lock-able dependent on the associated Pad Configuration register PadCfgLock setting. E.g. if the PadCfgLock of pad0 is 1, bit0 of this bit is locked down to read-only.

18.7.37 SMI Enable (GPI_SMI_EN_GPP_E_0)—Offset 158h

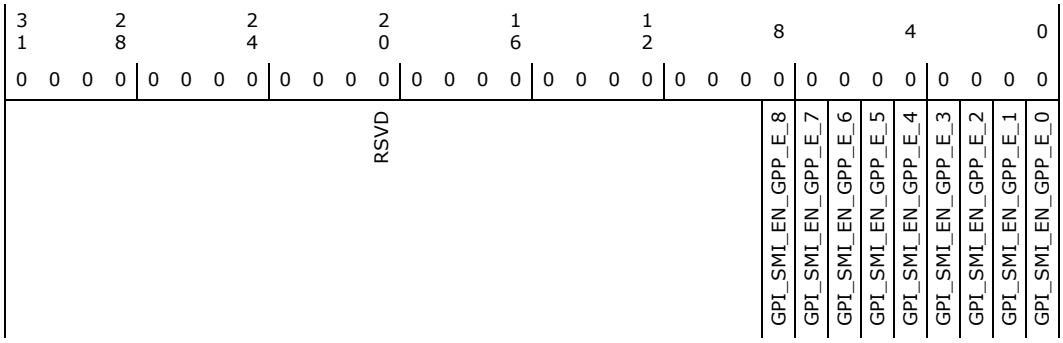
Register bits in this register are implemented for GPP_E signals that have SMI capability only. Other bits are reserved and RO.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:9	0h RO	Reserved
8	0h RW	**GPI SMI Enable (GPI_SMI_EN_GPP_E_8):** Same description as bit 0.
7	0h RW	**GPI SMI Enable (GPI_SMI_EN_GPP_E_7):** Same description as bit 0.
6	0h RW	**GPI SMI Enable (GPI_SMI_EN_GPP_E_6):** Same description as bit 0.
5	0h RW	**GPI SMI Enable (GPI_SMI_EN_GPP_E_5):** Same description as bit 0.
4	0h RW	**GPI SMI Enable (GPI_SMI_EN_GPP_E_4):** Same description as bit 0.
3	0h RW	**GPI SMI Enable (GPI_SMI_EN_GPP_E_3):** Same description as bit 0.
2	0h RW	**GPI SMI Enable (GPI_SMI_EN_GPP_E_2):** Same description as bit 0.
1	0h RW	**GPI SMI Enable (GPI_SMI_EN_GPP_E_1):** Same description as bit 0.
0	0h RW	**GPI SMI Enable (GPI_SMI_EN_GPP_E_0):** This bit is used to enable/disable the generation of SMI when the corresponding GPI_SMI_STS bit is set. The pad must also be routed for SMI functionality in order for SMI to be generated i.e., the corresponding GPIROUTSMI must be set to 1. 0 = disable SMI generation 1 = enable SMI generation ***Note:*** Each of the bit is lock-able dependent on the associated Pad Configuration register PadCfgLock setting. For example, if the PadCfgLock of pad0 is 1, bit0 of this bit is locked down to read-only.

18.7.38 NMI Status (GPI_NMI_STS_GPP_C_0)—Offset 160h

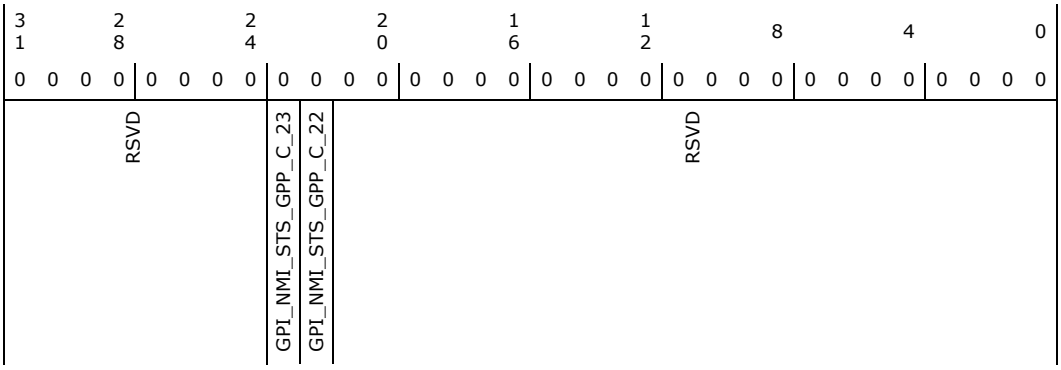
Register bits in this register are implemented for GPP_C signals that have NMI capability only. Other bits are reserved and RO.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW1C	**GPI NMI Status (GPI_NMI_STS_GPP_C_23):** Same description as bit 22.
22	0h RW1C	**GPI NMI Status (GPI_NMI_STS_GPP_C_22):** This bit is set to 1 by hardware when an edge event is detected (See RxEdCfg, RxInv) on pad and all the following conditions are true: - The corresponding pad is used in GPIO input mode (PMode). - The corresponding GPIONMIRout is set to 1, i.e., programmed to route as NMI - The corresponding GPIOOwn[2:0] is '000' (i.e., ACPI GPIO Mode). - The corresponding GPI_NMI_EN is set. Writing a value of 1 will clear the bit while writing a value of 0 has no effect. 0 = There is no NMI event. 1 = There is an NMI event.
21:0	0h RO	Reserved

18.7.39 NMI Status (GPI_NMI_STS_GPP_D_0)—Offset 164h

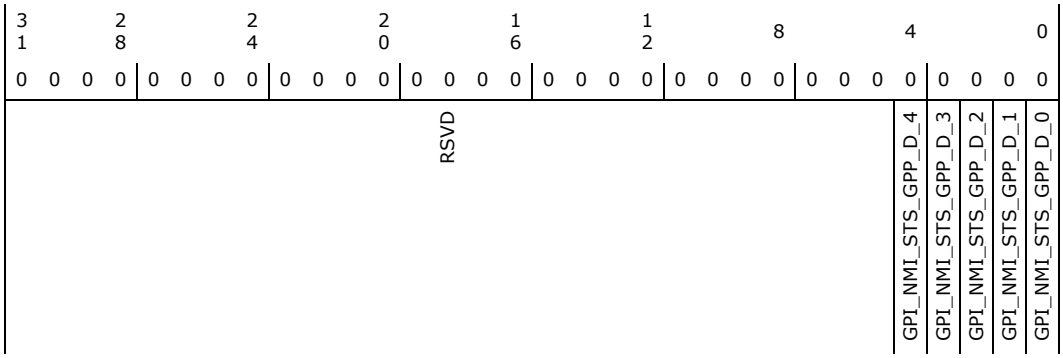
Register bits in this register are implemented for GPP_D signals that have NMI capability only. Other bits are reserved and RO.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:5	0h RO	Reserved
4	0h RW1C	**GPI NMI Status (GPI_NMI_STS_GPP_D_4):** Same description as bit 0.
3	0h RW1C	**GPI NMI Status (GPI_NMI_STS_GPP_D_3):** Same description as bit 0.
2	0h RW1C	**GPI NMI Status (GPI_NMI_STS_GPP_D_2):** Same description as bit 0.
1	0h RW1C	**GPI NMI Status (GPI_NMI_STS_GPP_D_1):** Same description as bit 0.
0	0h RW1C	**GPI NMI Status (GPI_NMI_STS_GPP_D_0):** This bit is set to 1 by hardware when an edge event is detected (See RxEdCfg, RxInv) on pad and all the following conditions are true: - The corresponding pad is used in GPIO input mode (PMode) - The corresponding GPIONMIRout is set to 1, i.e., programmed to route as NMI - The corresponding GPIOOwn[2:0] is '000' (i.e., ACPI GPIO Mode). - The corresponding GPI_NMI_EN is set Writing a value of 1 will clear the bit while writing a value of 0 has no effect. 0 = There is no NMI event 1 = There is an NMI event

18.7.40 NMI Status (GPI_NMI_STS_GPP_E_0)—Offset 168h

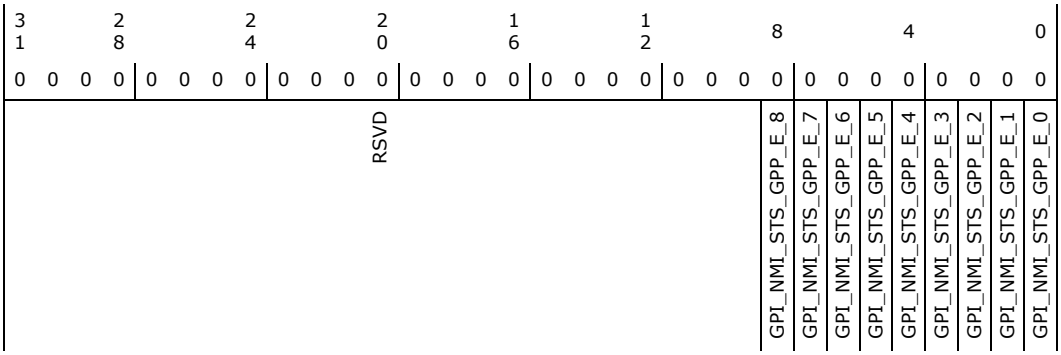
Register bits in this register are implemented for GPP_E signals that have NMI capability only. Other bits are reserved and RO.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:9	0h RO	Reserved
8	0h RW1C	**GPI NMI Status (GPI_NMI_STS_GPP_E_8):** Same description as bit 0.
7	0h RW1C	**GPI NMI Status (GPI_NMI_STS_GPP_E_7):** Same description as bit 0.
6	0h RW1C	**GPI NMI Status (GPI_NMI_STS_GPP_E_6):** Same description as bit 0.
5	0h RW1C	**GPI NMI Status (GPI_NMI_STS_GPP_E_5):** Same description as bit 0.
4	0h RW1C	**GPI NMI Status (GPI_NMI_STS_GPP_E_4):** Same description as bit 0.
3	0h RW1C	**GPI NMI Status (GPI_NMI_STS_GPP_E_3):** Same description as bit 0.
2	0h RW1C	**GPI NMI Status (GPI_NMI_STS_GPP_E_2):** Same description as bit 0.
1	0h RW1C	**GPI NMI Status (GPI_NMI_STS_GPP_E_1):** Same description as bit 0.
0	0h RW1C	**GPI NMI Status (GPI_NMI_STS_GPP_E_0):** This bit is set to 1 by hardware when an edge event is detected (See RxEdCfg, RxInv) on pad and all the following conditions are true: - The corresponding pad is used in GPIO input mode (PMode) - The corresponding GPIONMIRout is set to 1, i.e. programmed to route as NMI - The corresponding GPIOOwn[2:0] is '000' (i.e., ACPI GPIO Mode). - The corresponding GPI_NMI_EN is set Writing a value of 1 will clear the bit while writing a value of 0 has no effect. 0 = There is no NMI event 1 = There is an NMI event

18.7.41 NMI Enable (GPI_NMI_EN_GPP_C_0)—Offset 170h

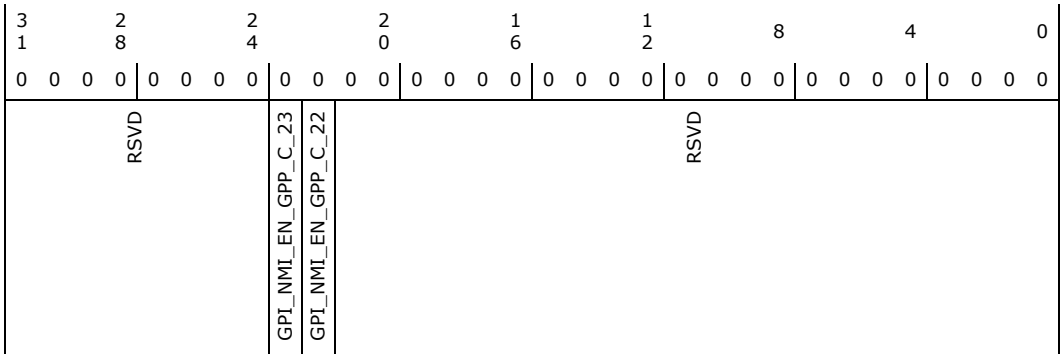
Register bits in this register are implemented for GPP_C signals that have NMI capability only. Other bits are reserved and RO.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**GPI NMI Enable (GPI_NMI_EN_GPP_C_23):** Same description as bit 22.
22	0h RW	**GPI NMI Enable (GPI_NMI_EN_GPP_C_22):** This bit is used to enable/disable the generation of NMI when the corresponding GPI_NMI_STS bit is set and its GPIROUTNMI is set. 0 = disable NMI generation 1 = enable NMI generation Each bit is lock-able dependent on the associated Pad Configuration register PadCfgLock setting. E.g. if the PadCfgLock of pad0 is '1', bit0 of this bit is locked down to read-only.
21:0	0h RO	Reserved

18.7.42 NMI Enable (GPI_NMI_EN_GPP_D_0)—Offset 174h

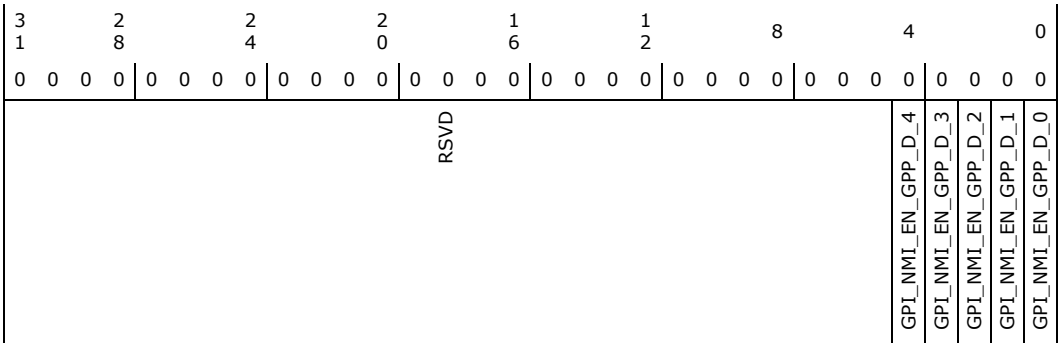
Register bits in this register are implemented for GPP_D signals that have NMI capability only. Other bits are reserved and RO.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:5	0h RO	Reserved
4	0h RW	**GPI NMI Enable (GPI_NMI_EN_GPP_D_4):** Same description as bit 0.
3	0h RW	**GPI NMI Enable (GPI_NMI_EN_GPP_D_3):** Same description as bit 0.
2	0h RW	**GPI NMI Enable (GPI_NMI_EN_GPP_D_2):** Same description as bit 0.
1	0h RW	**GPI NMI Enable (GPI_NMI_EN_GPP_D_1):** Same description as bit 0.
0	0h RW	**GPI NMI Enable (GPI_NMI_EN_GPP_D_0):** This bit is used to enable/disable the generation of NMI when the corresponding GPI_NMI_STS bit is set and its GPIROUTNMI is set. 0 = disable NMI generation 1 = enable NMI generation Each bit is lock-able dependent on the associated Pad Configuration register PadCfgLock setting. E.g. if the PadCfgLock of pad0 is '1', bit0 of this bit is locked down to read-only.

18.7.43 NMI Enable (GPI_NMI_EN_GPP_E_0)—Offset 178h

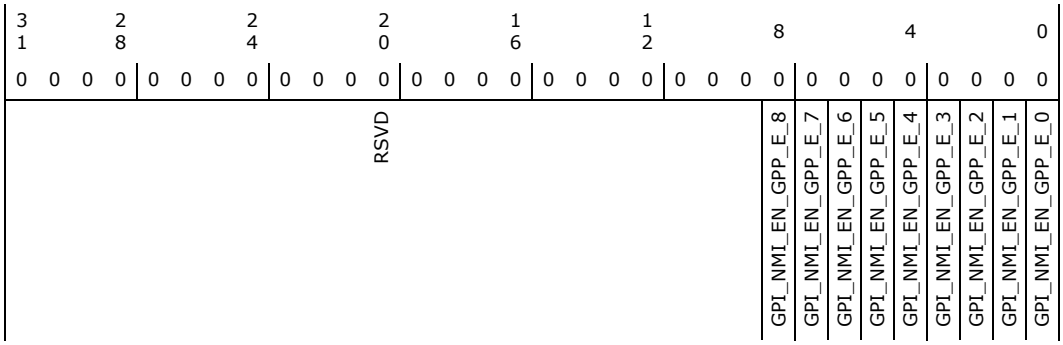
Register bits in this register are implemented for GPP_E signals that have NMI capability only. Other bits are reserved and RO.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:9	0h RO	Reserved
8	0h RW	**GPI NMI Enable (GPI_NMI_EN_GPP_E_8):** Same description as bit 0.
7	0h RW	**GPI NMI Enable (GPI_NMI_EN_GPP_E_7):** Same description as bit 0.
6	0h RW	**GPI NMI Enable (GPI_NMI_EN_GPP_E_6):** Same description as bit 0.
5	0h RW	**GPI NMI Enable (GPI_NMI_EN_GPP_E_5):** Same description as bit 0.
4	0h RW	**GPI NMI Enable (GPI_NMI_EN_GPP_E_4):** Same description as bit 0.
3	0h RW	**GPI NMI Enable (GPI_NMI_EN_GPP_E_3):** Same description as bit 0.
2	0h RW	**GPI NMI Enable (GPI_NMI_EN_GPP_E_2):** Same description as bit 0.
1	0h RW	**GPI NMI Enable (GPI_NMI_EN_GPP_E_1):** Same description as bit 0.
0	0h RW	**GPI NMI Enable (GPI_NMI_EN_GPP_E_0):** NMI enable This bit is used to enable/ disable the generation of NMI when the corresponding GPI_NMI_STS bit is set and its GPIROUTNMI is set. 0 = disable NMI generation 1 = enable NMI generation Each bit is lock-able dependent on the associated Pad Configuration register PadCfgLock setting. E.g. if the PadCfgLock of pad0 is '1', bit0 of this bit is locked down to read-only.

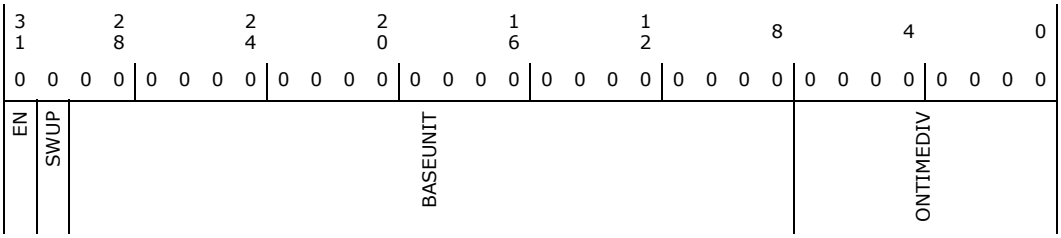
18.7.44 PWM Control (PWMC)—Offset 204h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31	0h RW	**Enable (EN):** 0 = Disable PWM Output 1 = Enable PWM Output
30	0h RW	**Software Update (SWUP):** Indication that there is an update to PWM settings pending. SW sets this bit to 1 when updating the PWM_base_unit or PWM_on_time_divisor fields. The PWM module will apply the new settings at the end of the current cycle and reset this bit. 0 = No updates pending 1 = Update pending
29:8	0h RW	**Base Unit (BASEUNIT):** Unsigned 8 integer bits, 14 fraction bits. Used to determine PWM output frequency. The PWM base frequency for the PCH is 32.768 kHz.
7:0	0h RW	**On Time Divisor (ONTIMEDIV):** On Time Divisor (OnTimeDiv) PWM duty cycle = PWM_on-time_divisor/256.

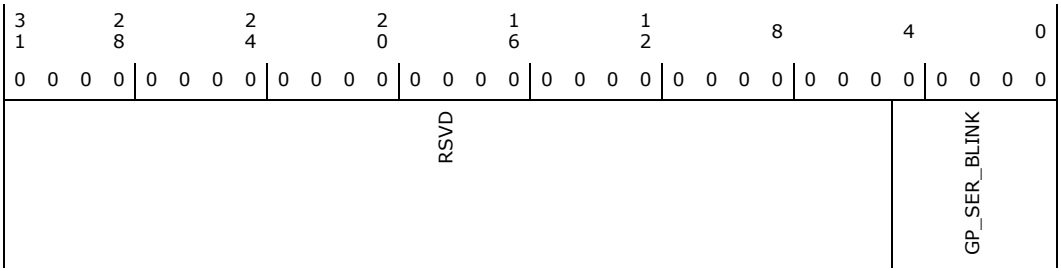
18.7.45 GPIO Serial Blink Enable (GP_SER_BLINK)—Offset 20Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:5	0h RO	Reserved
4:0	0h RW	**GP SER BLINK (GP_SER_BLINK):** The setting of this bit has no effect if the corresponding GPIO is programmed as an input, if the corresponding GPIO has the PWM enabled, or if Serial Blink capability does not exist. This bit should be set to a 1 before output buffer is enabled. When set to a '0', the corresponding GPIO will function normally. This bit should be set to a 1 while the corresponding PMode bit is set to 0h (GPIO Mode). Setting the PMode bit to other value (non-GPIO Mode) after the GP_SER_BLINK bit ensures PCH will not drive a 1 on the pin as an output. When this corresponding bit is set to a 1 and the pin is configured to output mode, the serial blink capability is enabled and the programmed message is serialized out through an open-drain buffer configuration. The value of the corresponding GPIOTxState bit remains unchanged and does not impact the serial blink capability in any way. Writes to this register have no effect when the corresponding pin is configured in native mode and the read value returned is undefined. Bit0 = GPP_D0 Bit1 = GPP_D1 Bit2 = GPP_D2 Bit3 = GPP_D3 Bit4 = GPP_D4

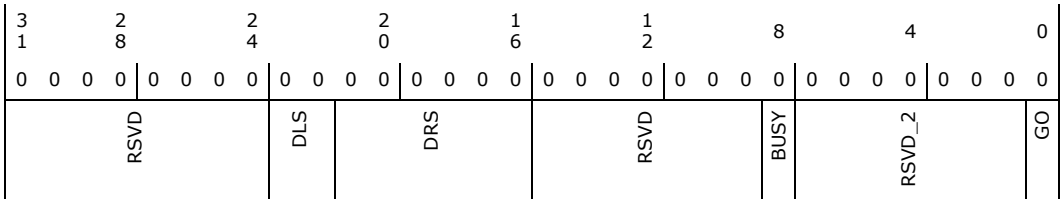
18.7.46 GPIO Serial Blink Command/Status (GP_SER_CMDSTS)—Offset 210h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23:22	0h RW	**Data Length Select (DLS):** This read/write field determines the number of bytes to serialize on GPIO 00: Serialize bits 7:0 of GP_GB_DATA (1 byte) 01: Serialize bits 15:0 of GP_GB_DATA (2 bytes) 10: Undefined - Software must not write this value 11: Serialize bits 31:0 of GP_GB_DATA (4 bytes) Software should not modify the value in this register unless the Busy bit
21:16	0h RW	**Data Rate Select (DRS):** Data Rate Select (DRS): This read/write field selects the number of 333.34 ns (4 clock periods 12 MHz clock) time intervals to count between Manchester data transitions. The default of 8h results in a 2666.67 ns minimum time between transitions. A value of 0h in this register produces undefined behavior. Software should not modify the value in this register unless the Busy bit is clear.
15:9	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
8	0h RO	**Busy (BUSY):** Busy: This read-only status bit is the hardware indication that a serialization is in progress. Hardware sets this bit to 1 based on the Go bit being set. Hardware clears this bit when the Go bit is cleared by the hardware.
7:1	0h RO	**Reserved (RSVD_2)**
0	0h RW	**Go (GO):** Go: This bit is set to 1 by software to start the serialization process. Hardware clears the bit after the serialized data is sent. Writes of 0 to this register have no effect. Software should not write this bit to 1 unless the Busy status bit is cleared.

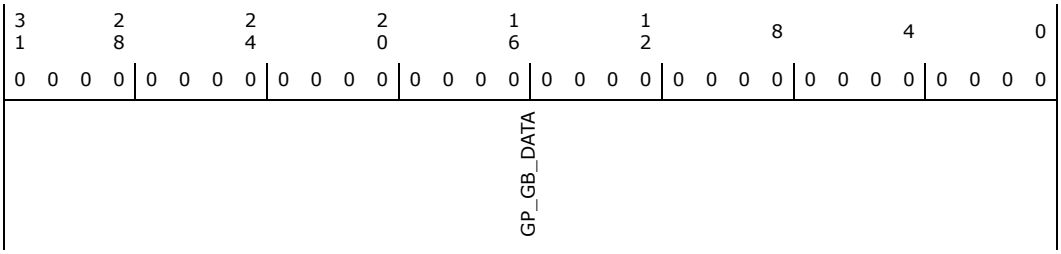
18.7.47 GPIO Serial Blink Data (GP_SER_DATA)—Offset 214h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:0	0h RW	**GP Serial Blink Data (GP_GB_DATA):** GP_GB_DATA This read-write register contains the data serialized out. The number of bits shifted out is selected through the DLS field in the GP_GB_CMDSTS register. This register should not be modified by software when the Busy bit is set.

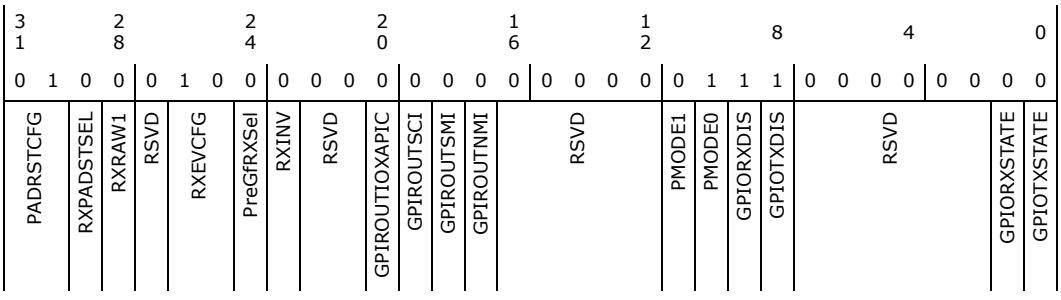
18.7.48 Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_0)—Offset 400h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.7.49 Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_0)—Offset 404h

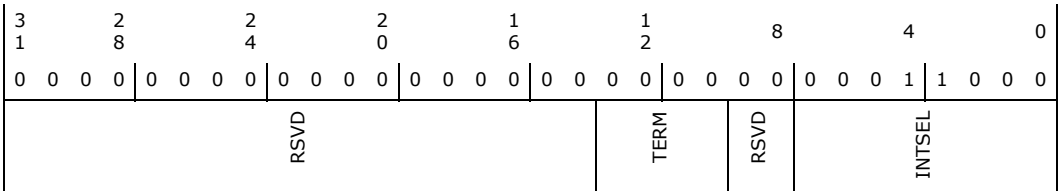
This register applies to GPP_C0.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 18h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	18h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.7.50 Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_1)—Offset 408h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG		PreGfRXSel	RXINV	RSVD		GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD							PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD					GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

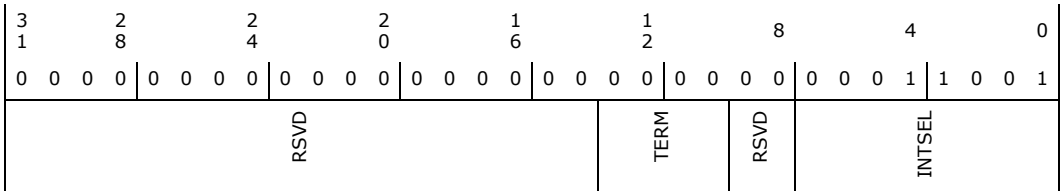
18.7.51 Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_1)—Offset 40Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 19h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	19h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

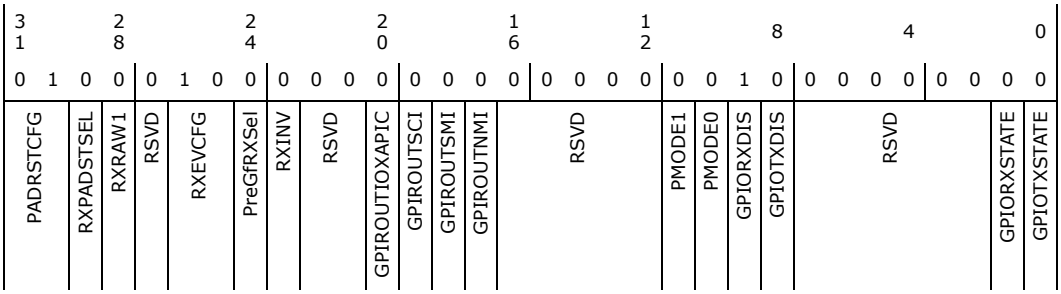
18.7.52 Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_2)—Offset 410h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000200h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).

Bit Range	Default & Access	Field Name (ID): Description
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

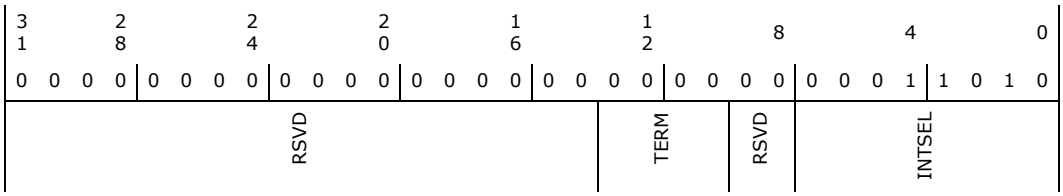
18.7.53 Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_2)—Offset 414h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Ah



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Ah RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

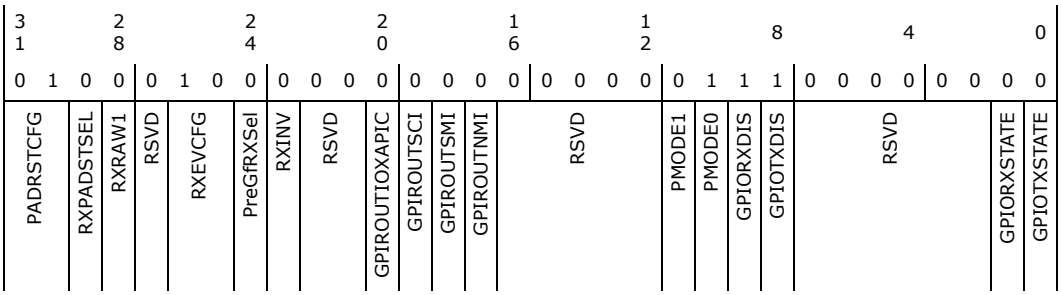
18.7.54 Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_3)—Offset 418h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	**Reserved**
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

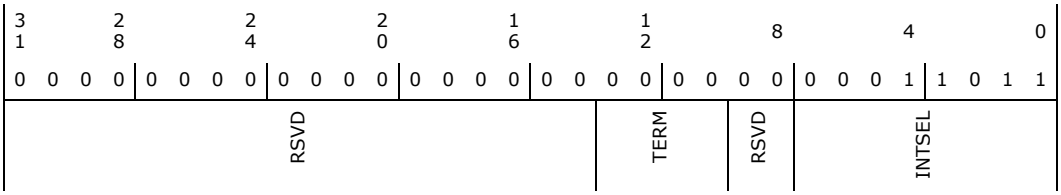
18.7.55 Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_3)—Offset 41Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Bh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Bh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

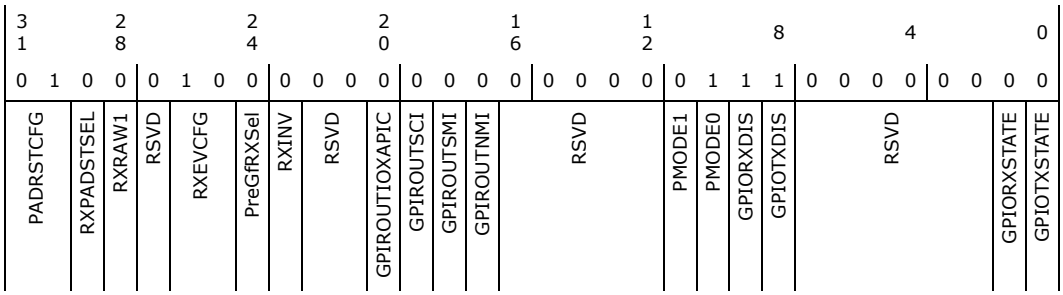
18.7.56 Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_4)—Offset 420h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

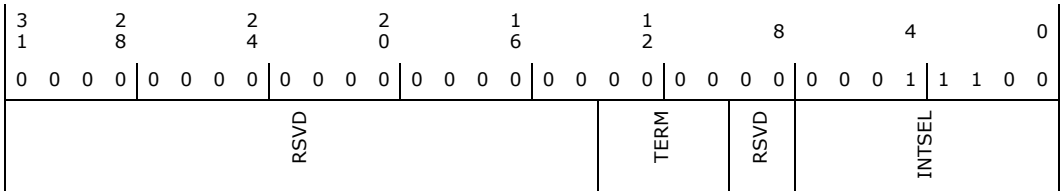
18.7.57 Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_4)—Offset 424h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Ch



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Ch RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

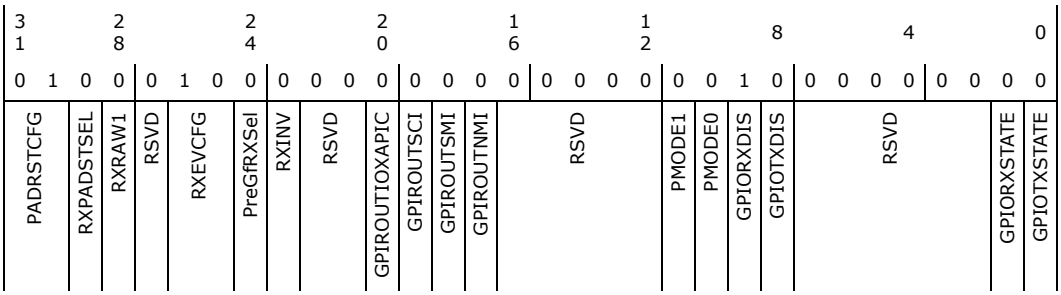
18.7.58 Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_5)—Offset 428h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000200h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

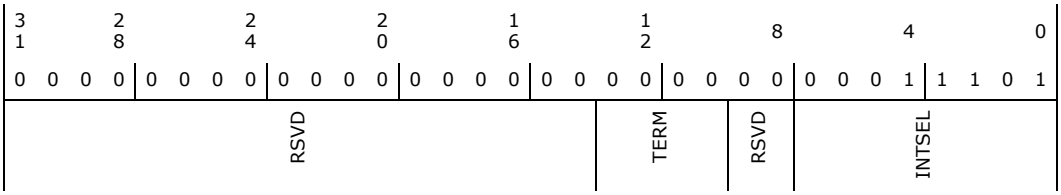
18.7.59 Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_5)—Offset 42Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Dh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Dh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

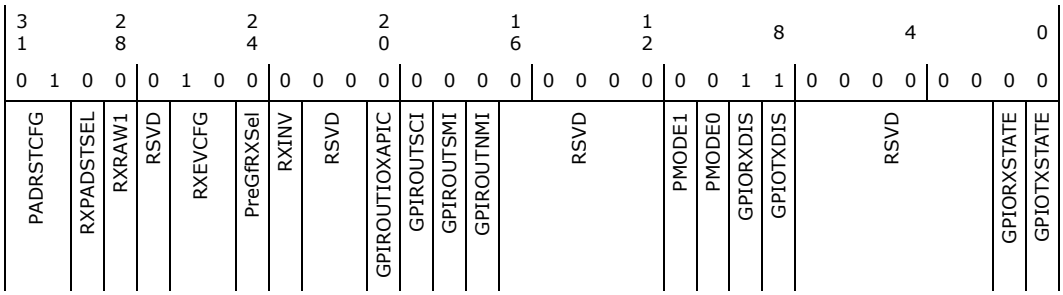
18.7.60 Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_6)—Offset 430h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

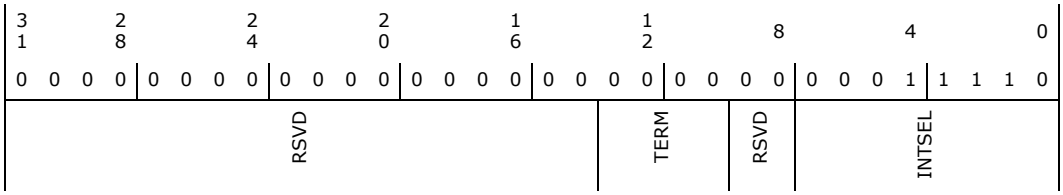
18.7.61 Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_6)—Offset 434h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Eh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Eh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

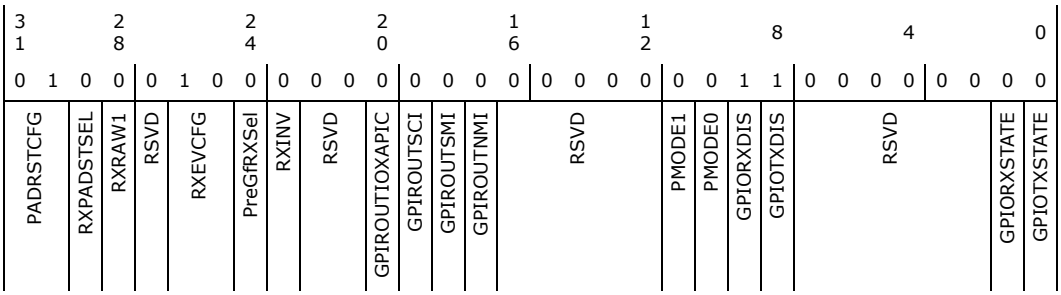
18.7.62 Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_7)—Offset 438h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.

Bit Range	Default & Access	Field Name (ID): Description
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

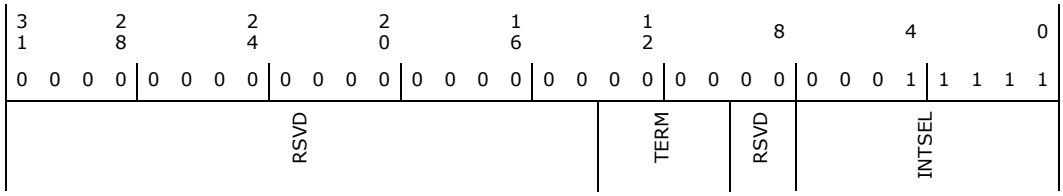
18.7.63 Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_7)—Offset 43Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Fh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Fh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

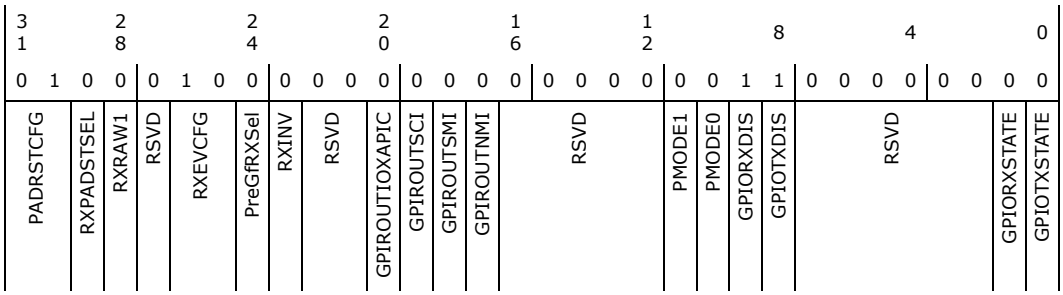
18.7.64 Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_8)—Offset 440h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

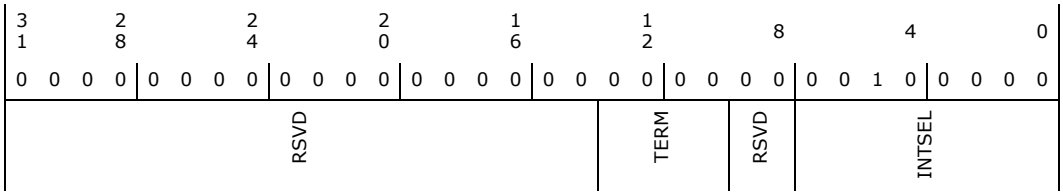
18.7.65 Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_8)—Offset 444h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 20h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	20h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

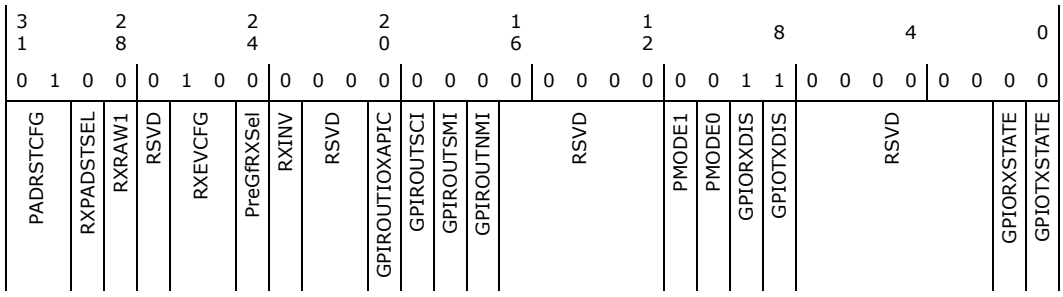
18.7.66 Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_9)—Offset 448h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

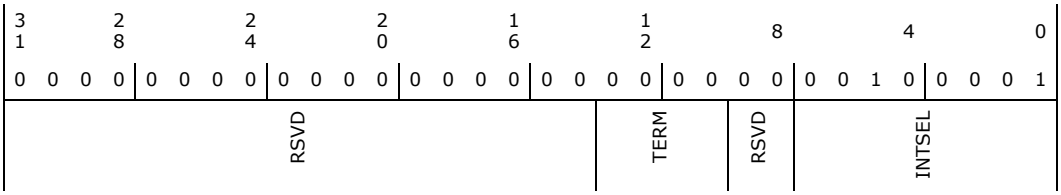
18.7.67 Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_9)—Offset 44Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 21h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	21h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

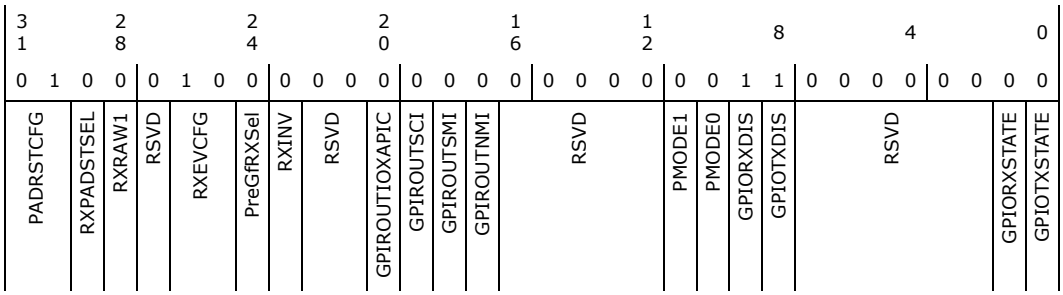
18.7.68 Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_10)—Offset 450h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

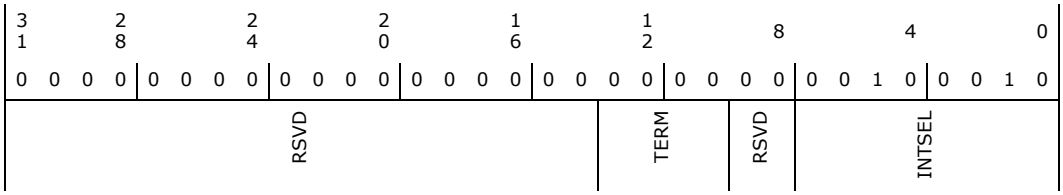
18.7.69 Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_10)—Offset 454h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 22h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	22h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

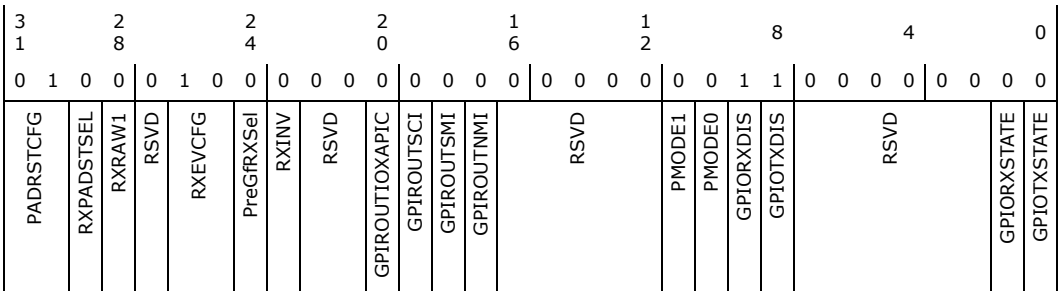
18.7.70 Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_11)—Offset 458h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

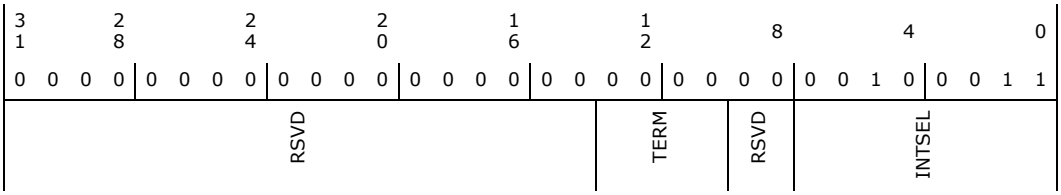
18.7.71 Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_11)—Offset 45Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 23h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	23h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

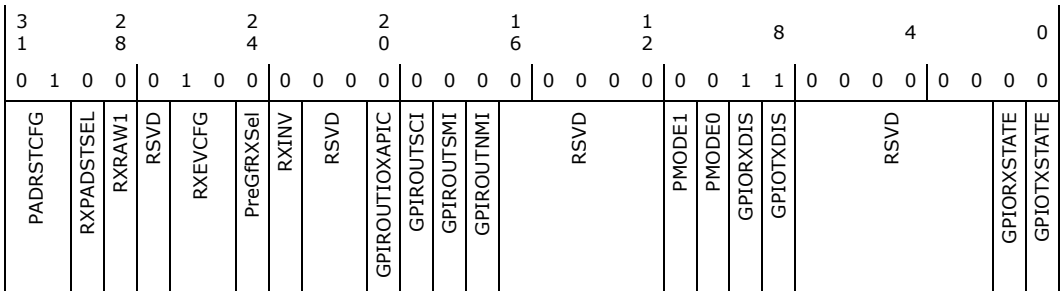
18.7.72 Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_12)—Offset 460h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.7.73 Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_12)—Offset 464h

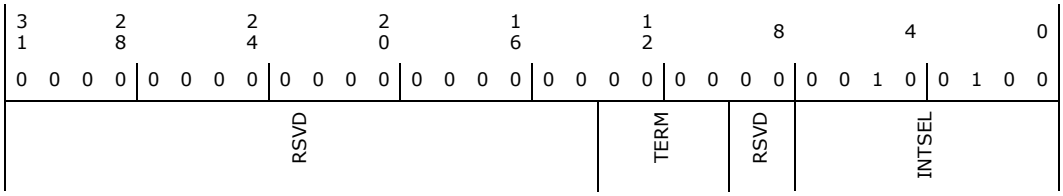
Same description as PAD_CFG_DW1_GPP_C_0 Exception: The default value of the INTSEL bit field in this register is 24h.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 24h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	24h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

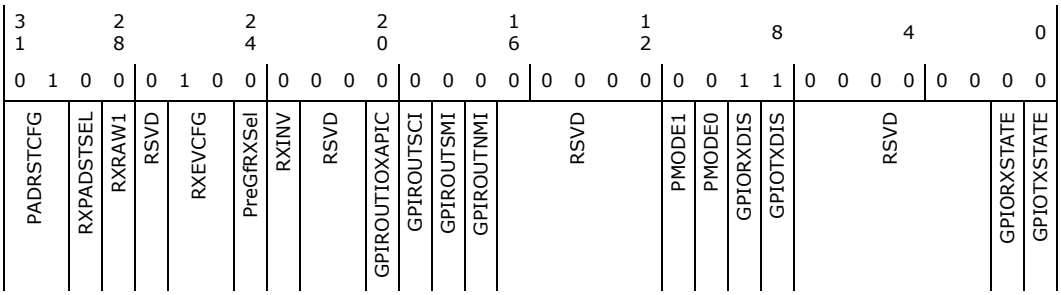
18.7.74 Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_13)—Offset 468h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

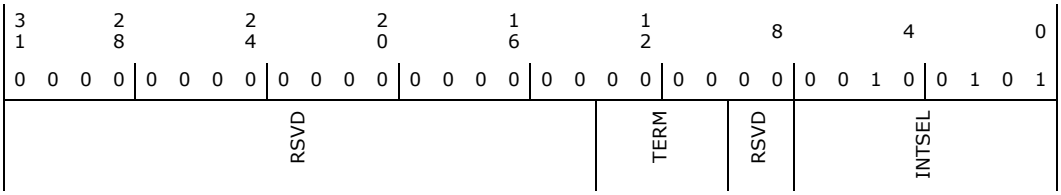
18.7.75 Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_13)—Offset 46Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 25h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	25h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

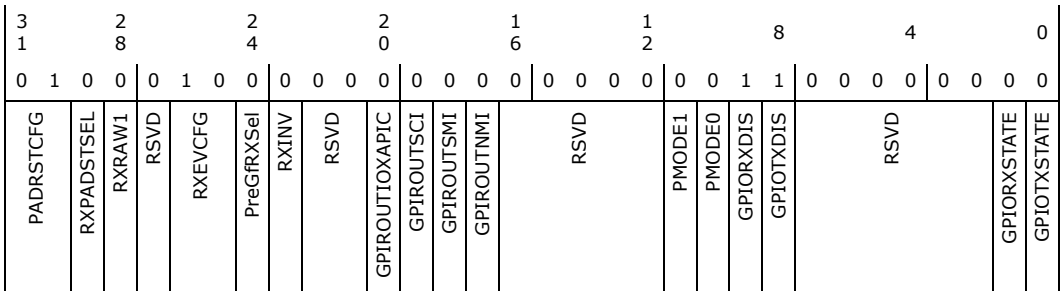
18.7.76 Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_14)—Offset 470h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

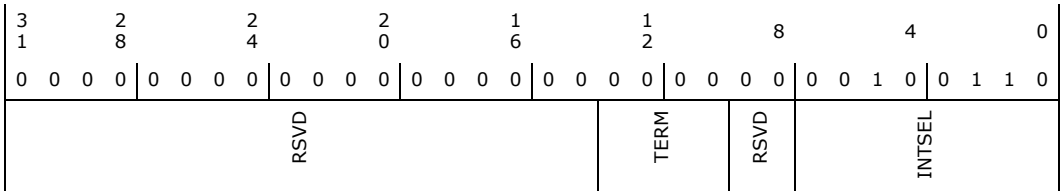
18.7.77 Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_14)—Offset 474h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 26h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	26h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

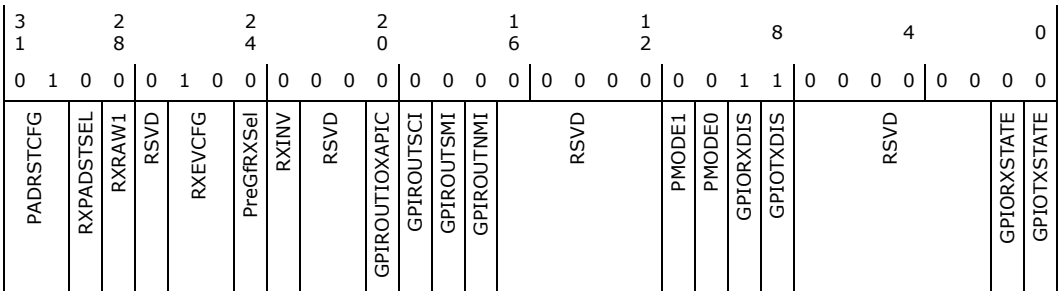
18.7.78 Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_15)—Offset 478h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

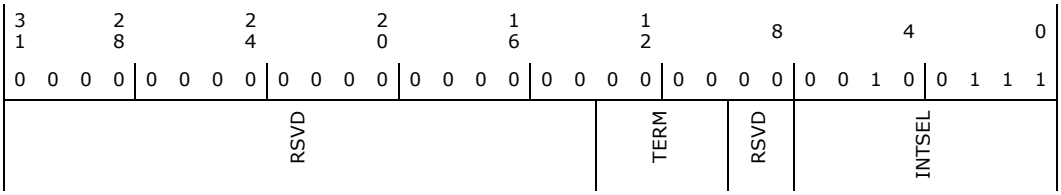
18.7.79 Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_15)—Offset 47Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 27h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	27h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

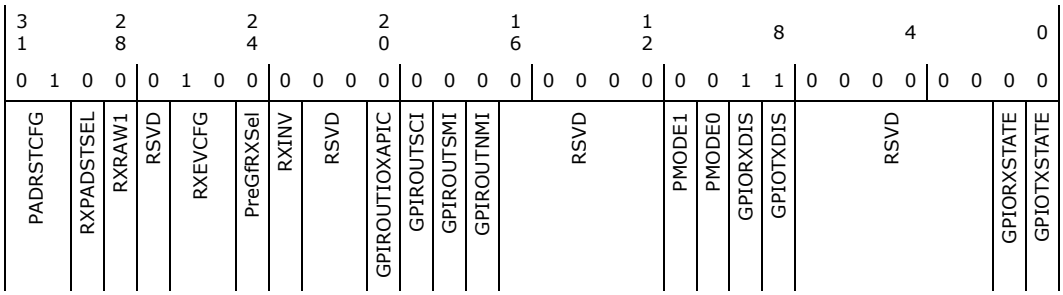
18.7.80 Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_16)—Offset 480h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

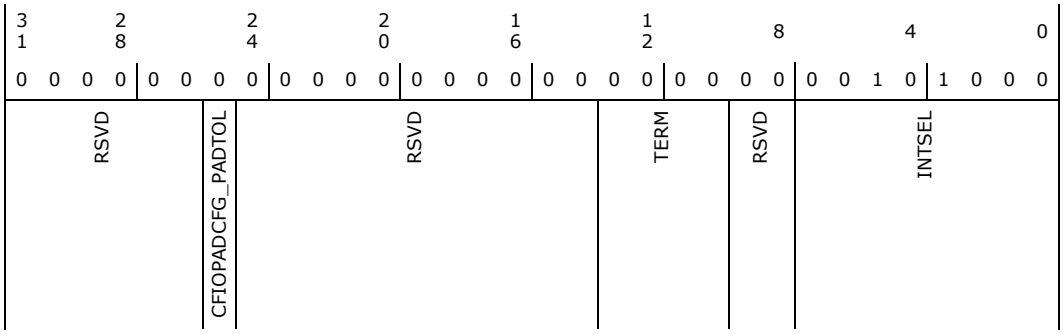
18.7.81 Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_16)—Offset 484h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 28h



Bit Range	Default & Access	Field Name (ID): Description
31:26	0h RO	Reserved
25	0h RW	**Pad Voltage Tolerance (CFIOPADCFG_PADTOL):** 0 = pad is 3.3V tolerance (VccIO is 3.3V) 1 = pad is 1.8V tolerance (VccIO is 3.3V).
24:14	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
13:10	0h RO	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	28h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

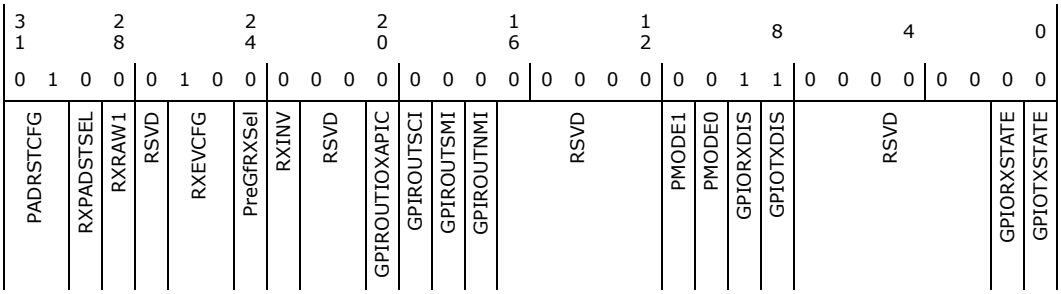
18.7.82 Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_17)—Offset 488h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

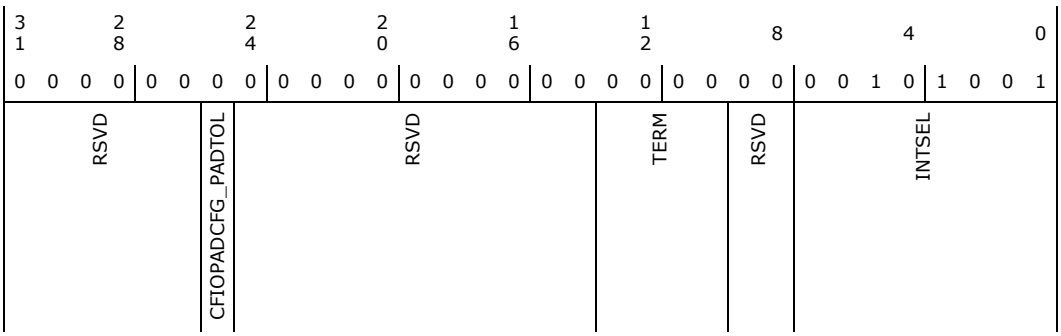
18.7.83 Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_17)—Offset 48Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 29h



Bit Range	Default & Access	Field Name (ID): Description
31:26	0h RO	Reserved
25	0h RW	**Pad Voltage Tolerance (CFIOPADCFG_PADTOL):** 0 = pad is 3.3V tolerance (VccIO is 3.3V) 1 = pad is 1.8V tolerance (VccIO is 3.3V).
24:14	0h RO	Reserved
13:10	0h RO	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	29h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.7.84 Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_18)—Offset 490h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG		PreGfRXSel	RXINV	RSVD		GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD						PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

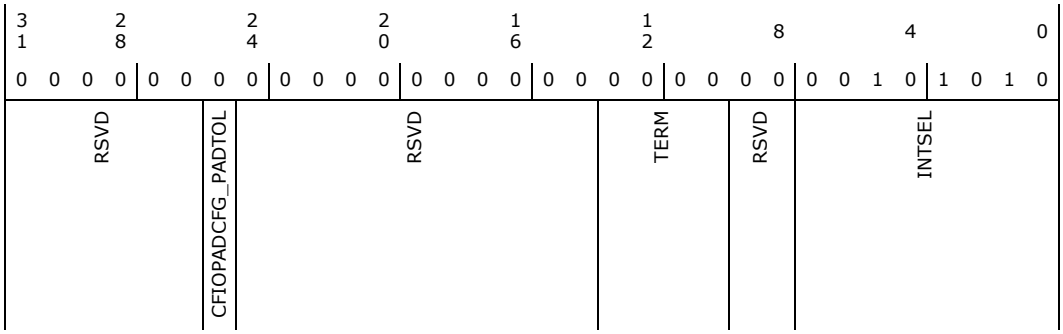
18.7.85 Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_18)—Offset 494h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 2Ah



Bit Range	Default & Access	Field Name (ID): Description
31:26	0h RO	Reserved
25	0h RW	**Pad Voltage Tolerance (CFIOPADCFG_PADTOL):** 0 = pad is 3.3V tolerance (VccIO is 3.3V) 1 = pad is 1.8V tolerance (VccIO is 3.3V).
24:14	0h RO	Reserved
13:10	0h RO	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	2Ah RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.7.86 Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_19)—Offset 498h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD		RXEVCFG	PreGfRXSel	RXINV	RSVD		GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD				GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.7.87 Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_19)—Offset 49Ch

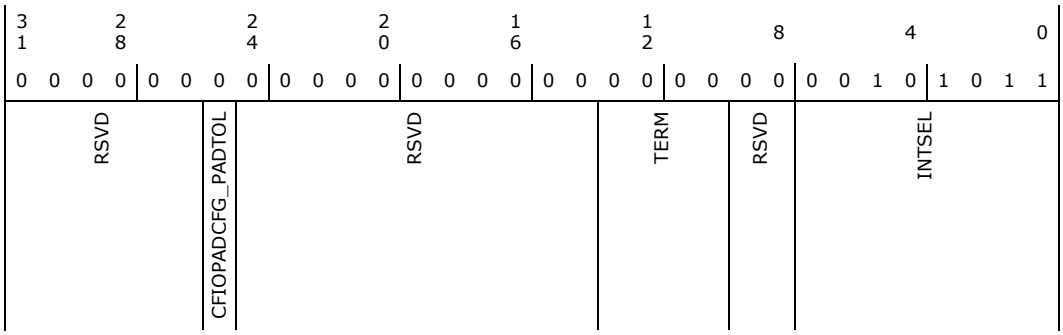
Same description as PAD_CFG_DW1_GPP_C_16 Exception: The default value of the INTSEL bit field in this register is 2Bh.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 2Bh



Bit Range	Default & Access	Field Name (ID): Description
31:26	0h RO	Reserved
25	0h RW	**Pad Voltage Tolerance (CFIOPADCFG_PADTOL):** 0 = pad is 3.3V tolerance (VccIO is 3.3V) 1 = pad is 1.8V tolerance (VccIO is 3.3V).
24:14	0h RO	Reserved
13:10	0h RO	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	2Bh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

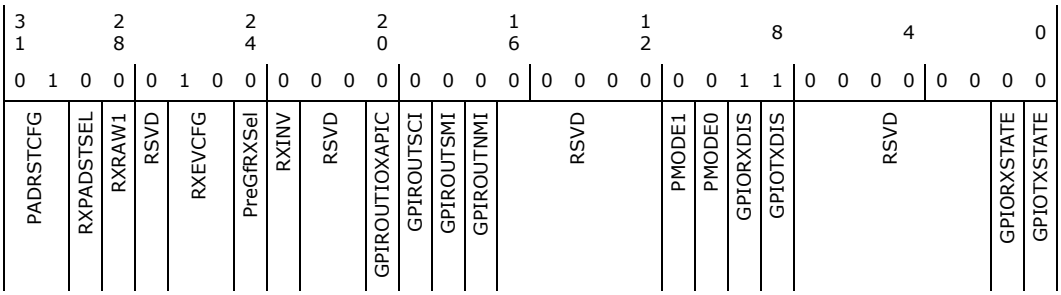
18.7.88 Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_20)—Offset 4A0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

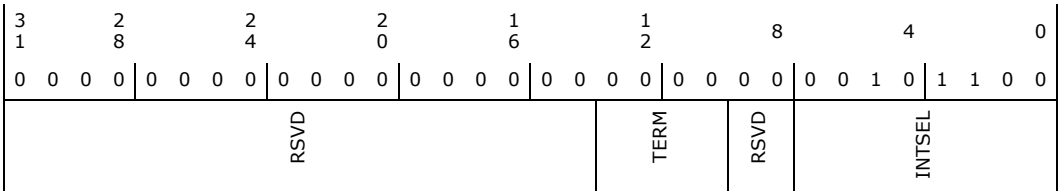
18.7.89 Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_20)—Offset 4A4h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 2Ch



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	2Ch RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

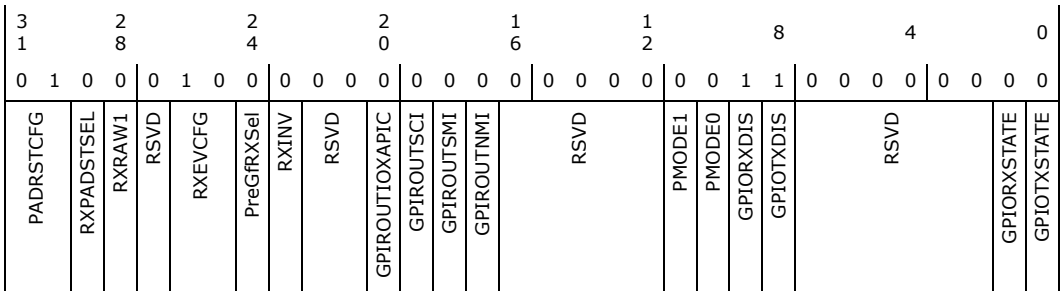
18.7.90 Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_21)—Offset 4A8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

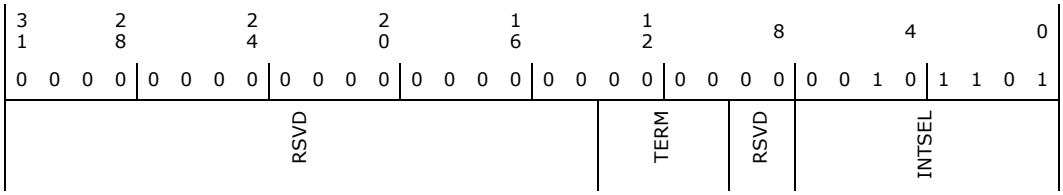
18.7.91 Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_21)—Offset 4ACh

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 2Dh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	2Dh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

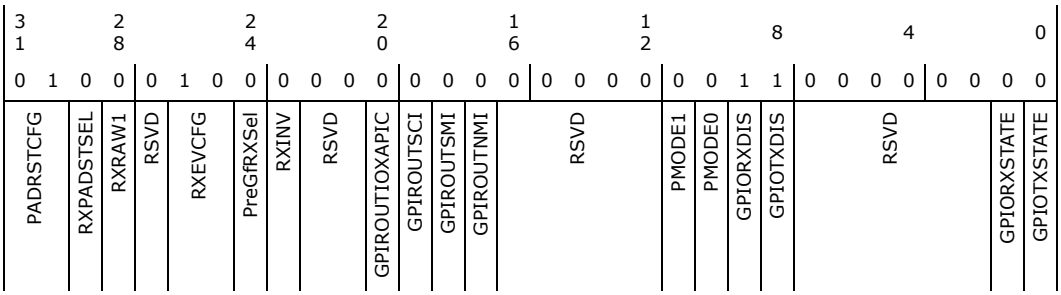
18.7.92 Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_22)—Offset 4B0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

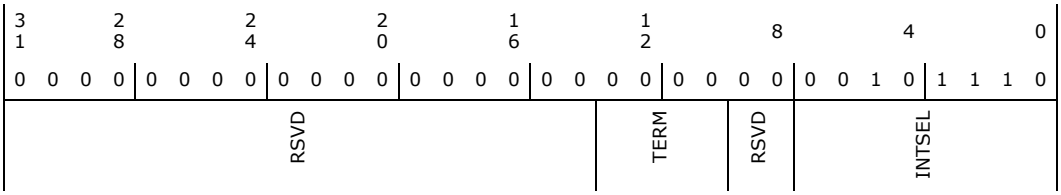
18.7.93 Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_22)—Offset 4B4h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 2Eh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	2Eh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

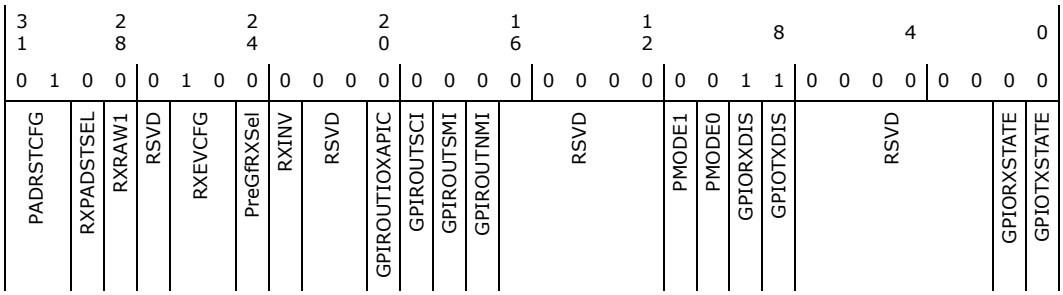
18.7.94 Pad Configuration DW0 (PAD_CFG_DW0_GPP_C_23)—Offset 4B8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

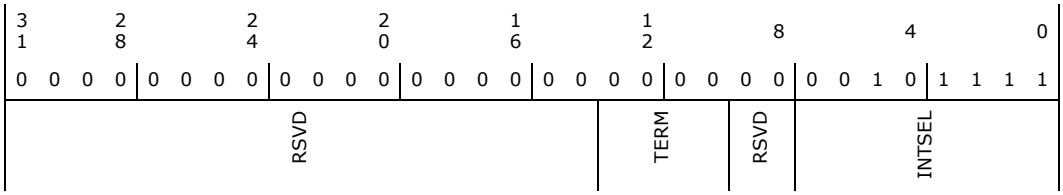
18.7.95 Pad Configuration DW1 (PAD_CFG_DW1_GPP_C_23)—Offset 4BCh

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 2Fh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	2Fh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

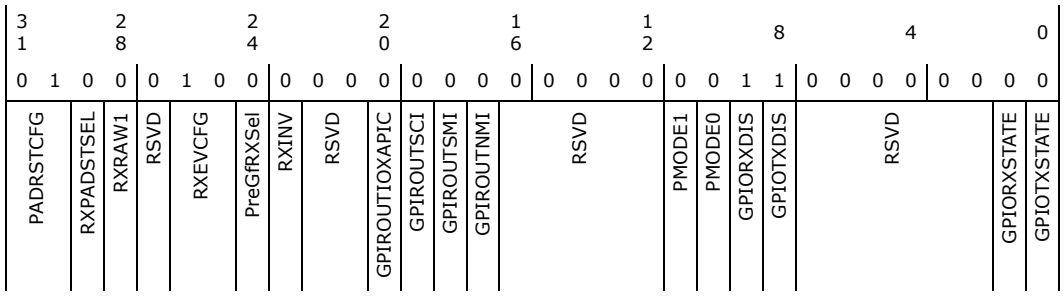
18.7.96 Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_0)—Offset 4C0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

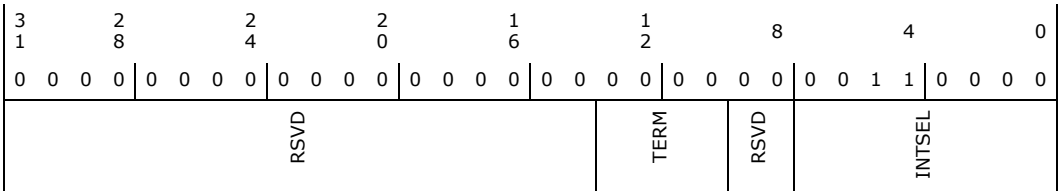
18.7.97 Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_0)—Offset 4C4h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 30h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	30h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

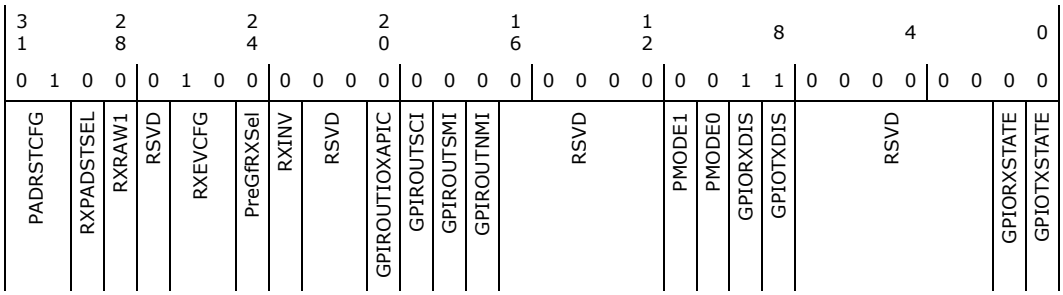
18.7.98 Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_1)—Offset 4C8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

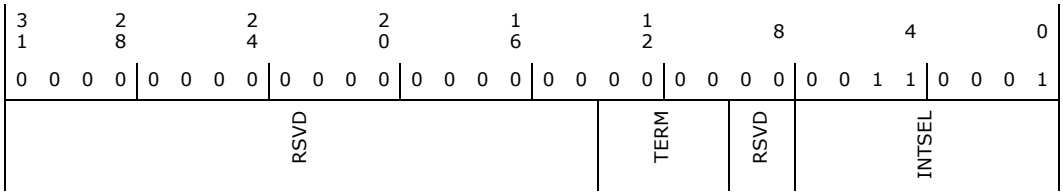
18.7.99 Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_1)—Offset 4CCh

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 31h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	31h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

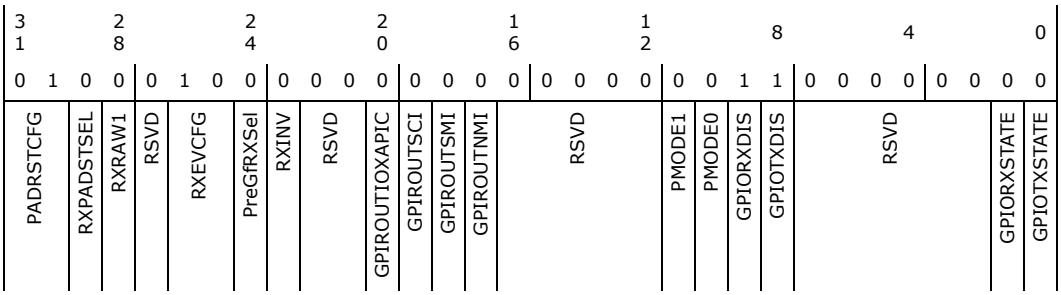
18.7.100 Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_2)—Offset 4D0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

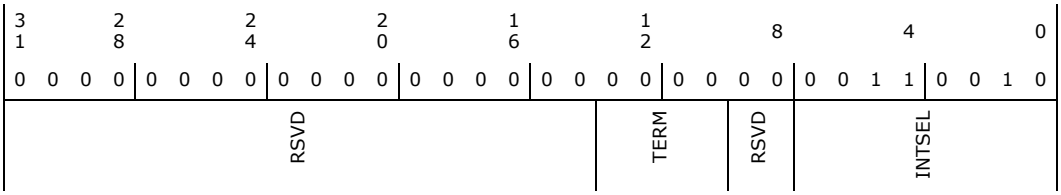
18.7.101 Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_2)—Offset 4D4h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 32h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	32h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

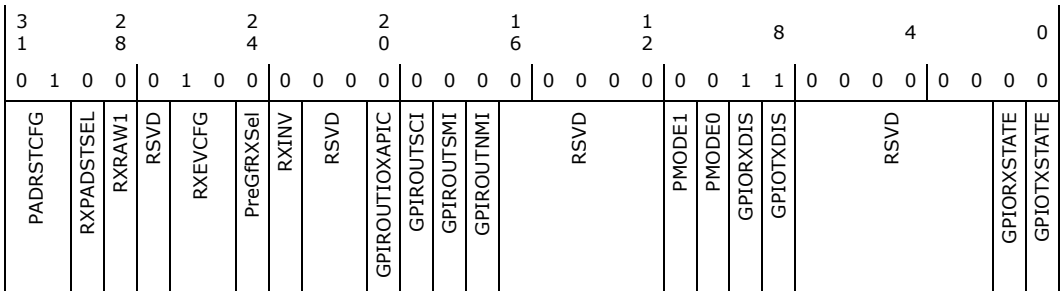
18.7.102 Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_3)—Offset 4D8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

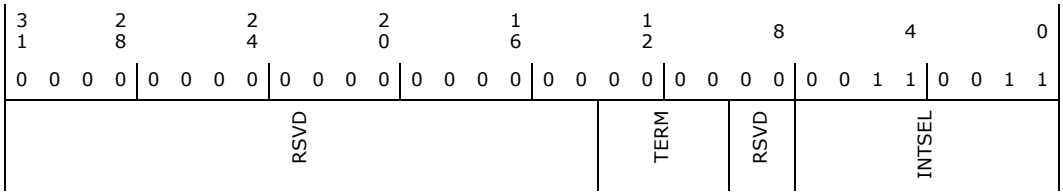
18.7.103 Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_3)—Offset 4DCh

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 33h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	33h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

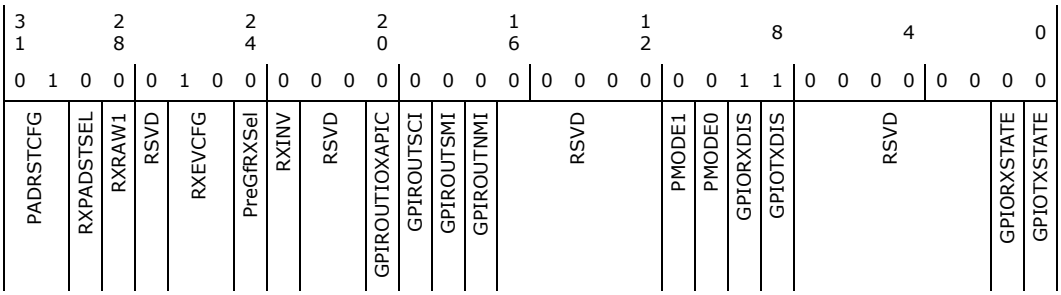
18.7.104 Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_4)—Offset 4E0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.7.105 Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_4)—Offset 4E4h

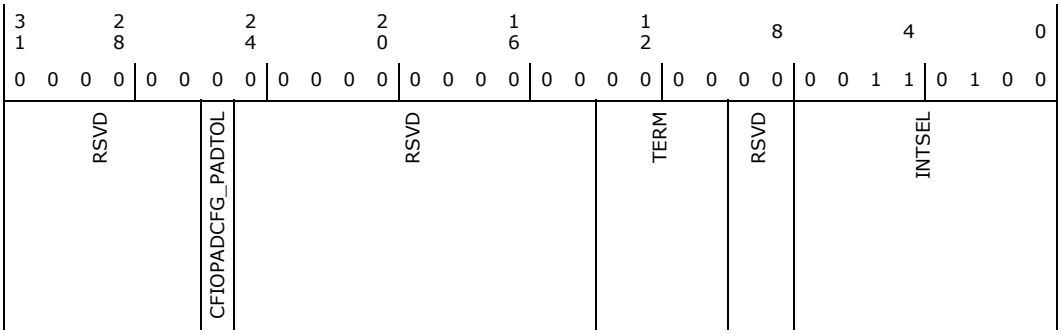
Same description as PAD_CFG_DW1_GPP_C_16 Exception: The default value of the INTSEL bit field in this register is 34h.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 34h



Bit Range	Default & Access	Field Name (ID): Description
31:26	0h RO	Reserved
25	0h RW	**Pad Voltage Tolerance (CFIOPADCFG_PADTOL):** 0 = pad is 3.3V tolerance (VccIO is 3.3V) 1 = pad is 1.8V tolerance (VccIO is 3.3V).
24:14	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
13:10	0h RO	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	34h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

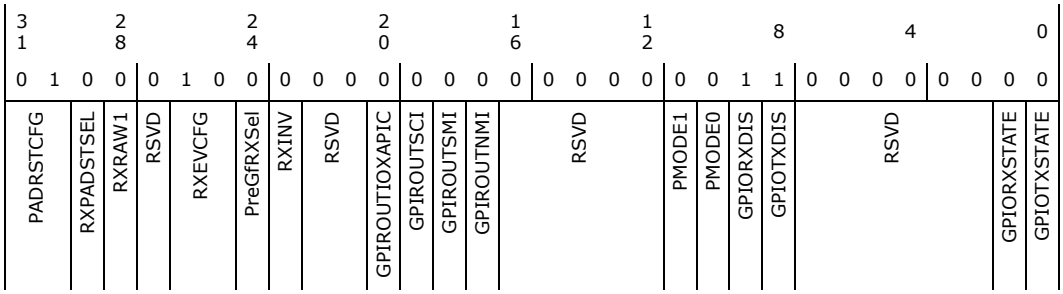
18.7.106 Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_5)—Offset 4E8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

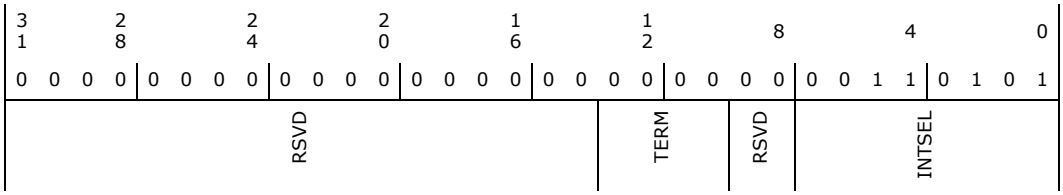
18.7.107 Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_5)—Offset 4ECh

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 35h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	35h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.7.108 Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_6)—Offset 4F0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD		RXEVCFG		PreGfRXSel	RXINV	RSVD		GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD							PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD					GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

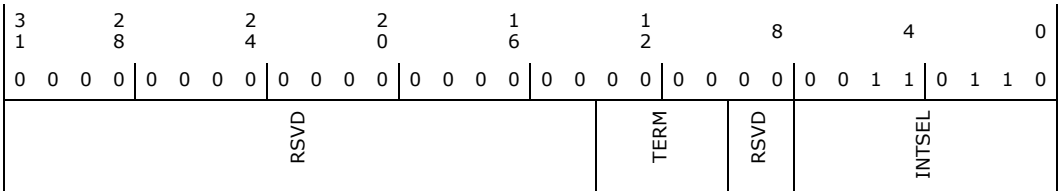
18.7.109 Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_6)—Offset 4F4h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 36h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	36h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.7.110 Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_7)—Offset 4F8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG		PreGfRXSel	RXINV	RSVD		GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD						PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

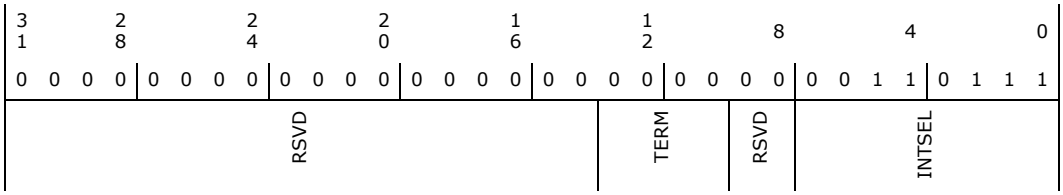
18.7.111 Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_7)—Offset 4FCh

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 37h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	37h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.7.112 Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_8)—Offset 500h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

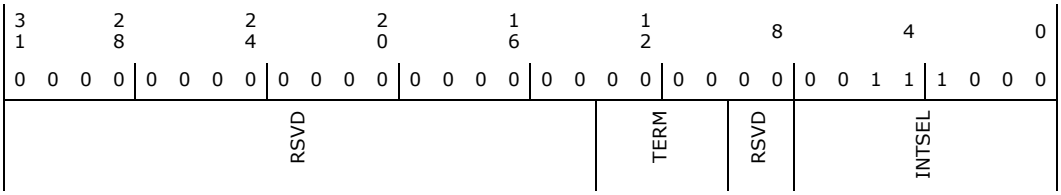
18.7.113 Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_8)—Offset 504h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 38h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	38h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.7.114 Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_9)—Offset 508h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG		PreGfRXSel	RXINV	RSVD		GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD						PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

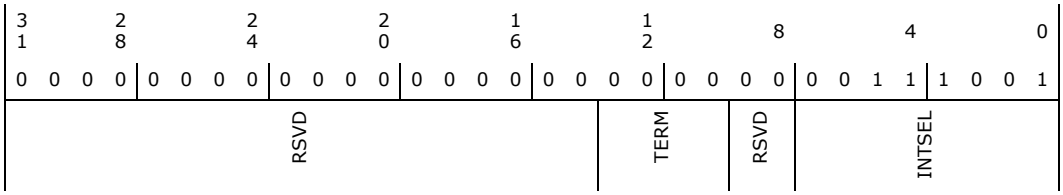
18.7.115 Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_9)—Offset 50Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 39h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	39h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.7.116 Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_10)—Offset 510h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).

Bit Range	Default & Access	Field Name (ID): Description
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

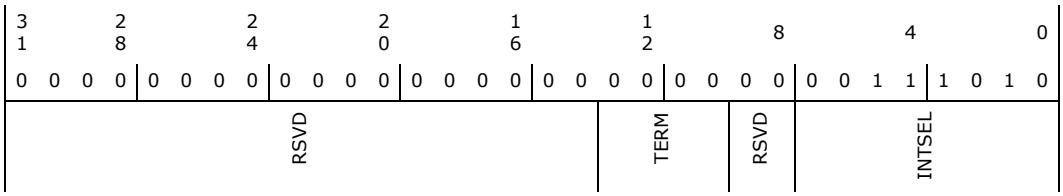
18.7.117 Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_10)—Offset 514h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 3Ah



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	3Ah RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

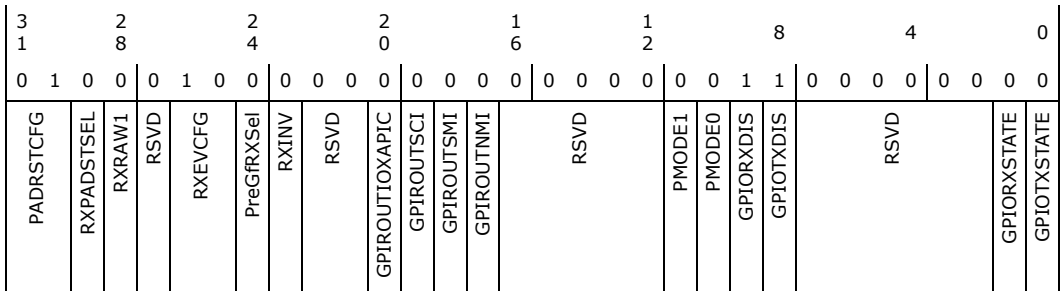
18.7.118 Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_11)—Offset 518h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

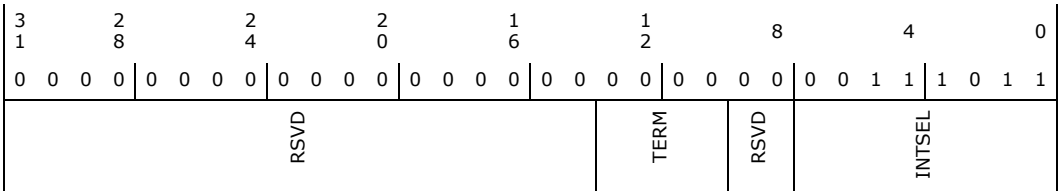
18.7.119 Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_11)—Offset 51Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 3Bh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	3Bh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

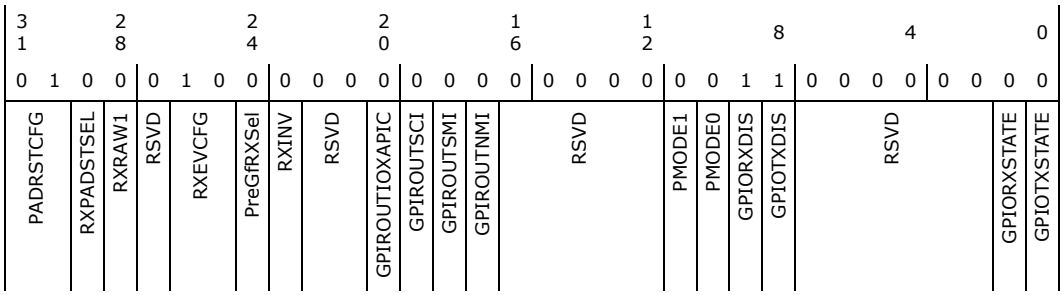
18.7.120 Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_12)—Offset 520h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

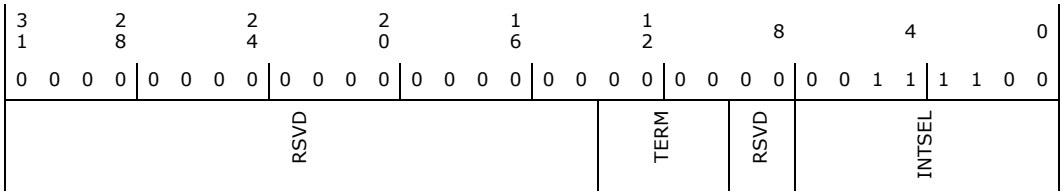
18.7.121 Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_12)—Offset 524h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 3Ch



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	3Ch RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

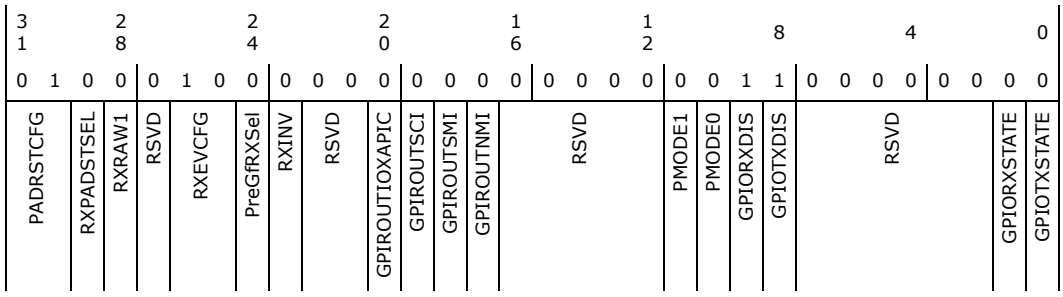
18.7.122 Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_13)—Offset 528h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

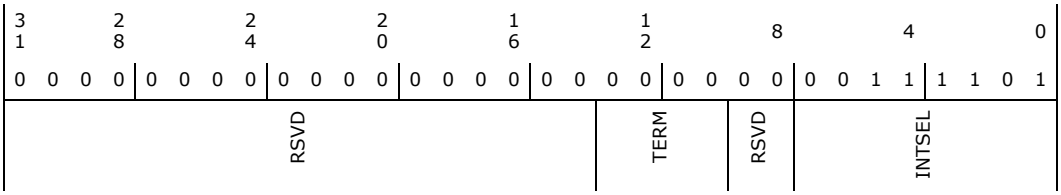
18.7.123 Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_13)—Offset 52Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 3Dh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	3Dh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

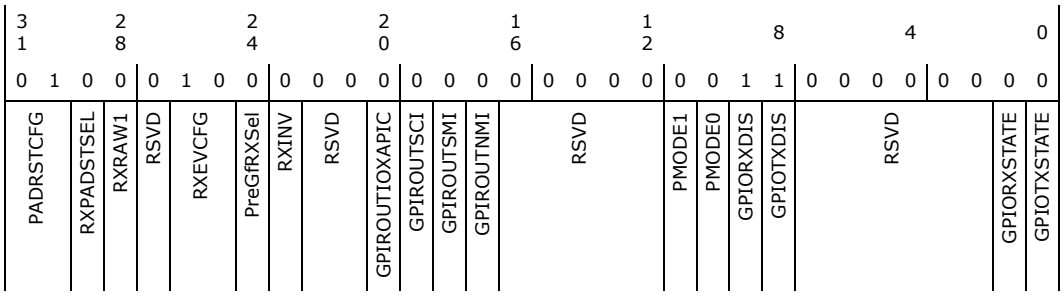
18.7.124 Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_14)—Offset 530h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

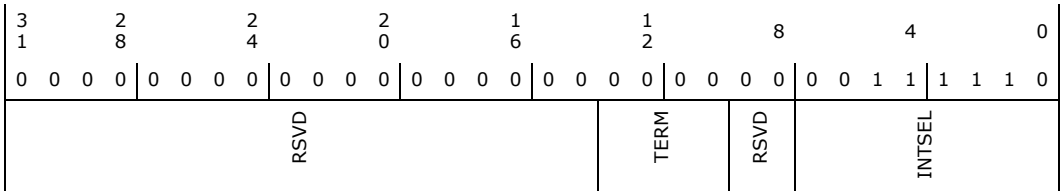
18.7.125 Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_14)—Offset 534h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 3Eh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	3Eh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

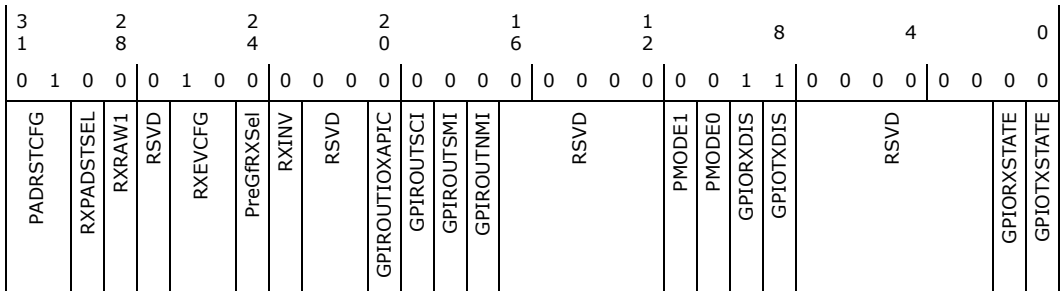
18.7.126 Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_15)—Offset 538h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

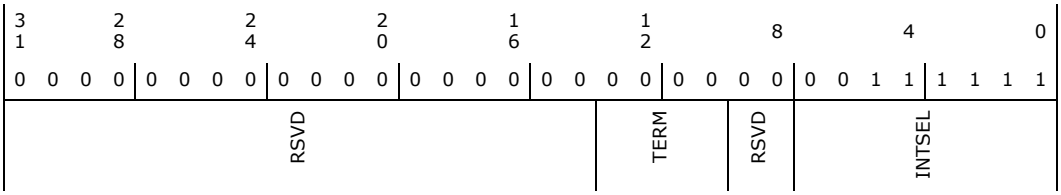
18.7.127 Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_15)—Offset 53Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 3Fh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	3Fh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

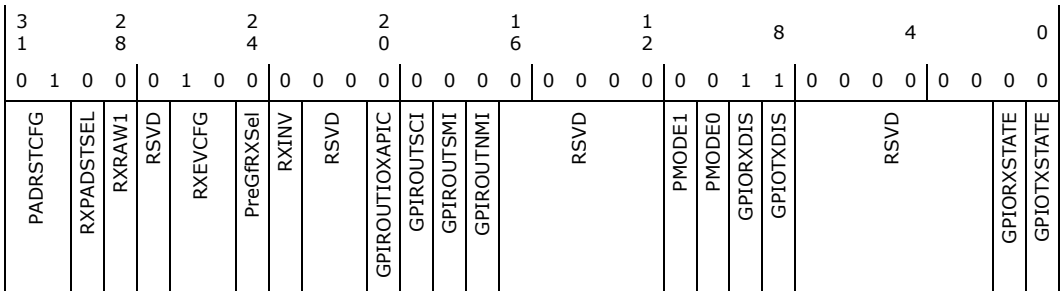
18.7.128 Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_16)—Offset 540h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0’s in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

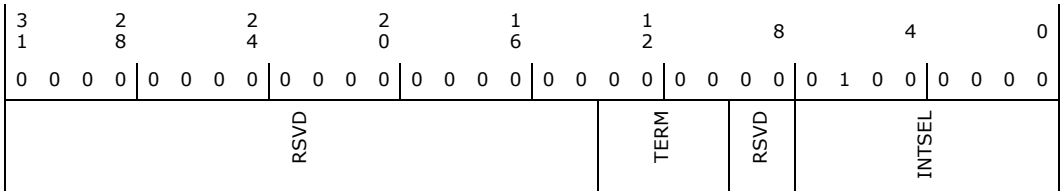
18.7.129 Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_16)—Offset 544h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 40h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	40h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

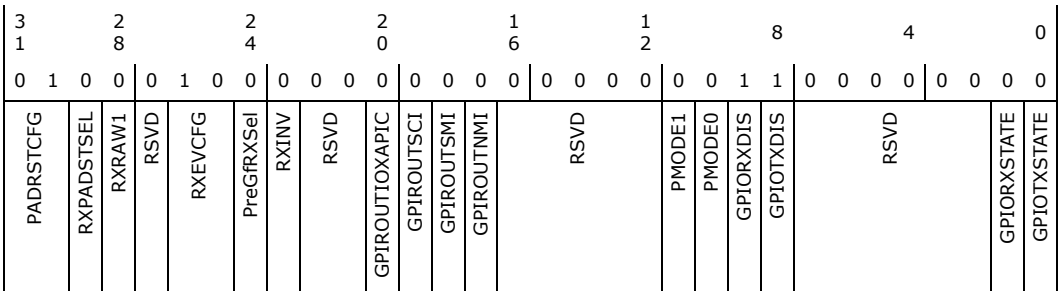
18.7.130 Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_17)—Offset 548h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad;i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

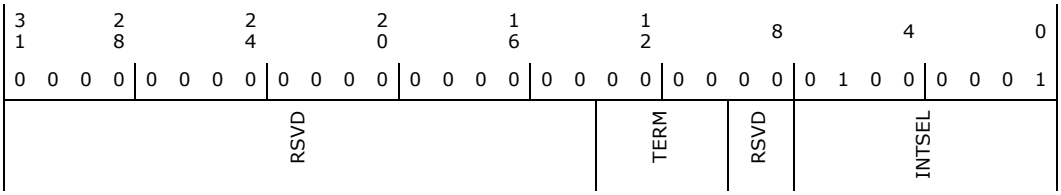
18.7.131 Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_17)—Offset 54Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 41h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	41h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

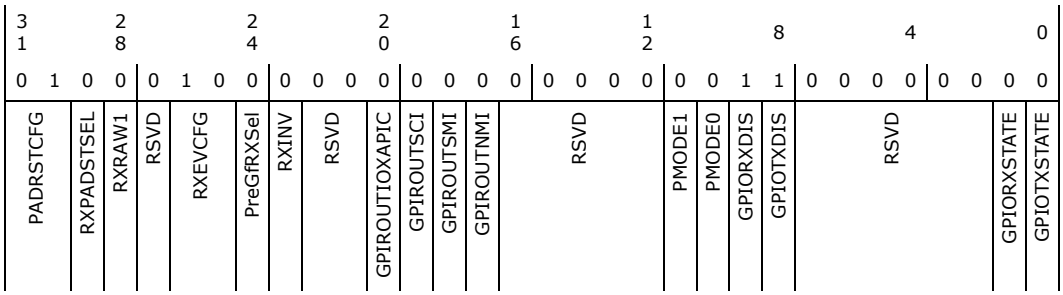
18.7.132 Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_18)—Offset 550h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

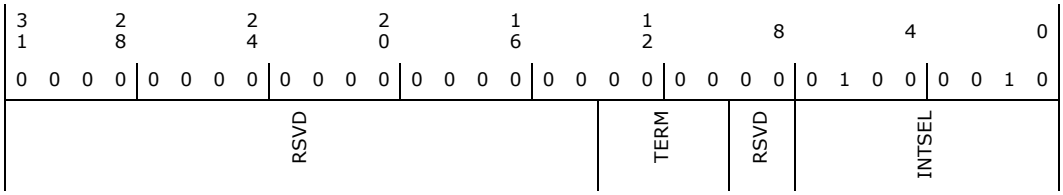
18.7.133 Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_18)—Offset 554h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 42h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	42h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

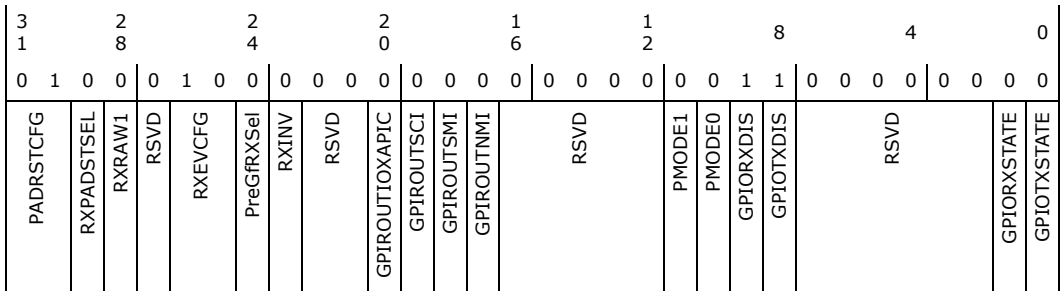
18.7.134 Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_19)—Offset 558h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

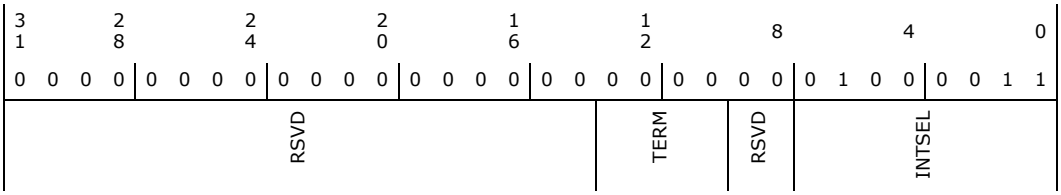
18.7.135 Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_19)—Offset 55Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 43h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	43h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

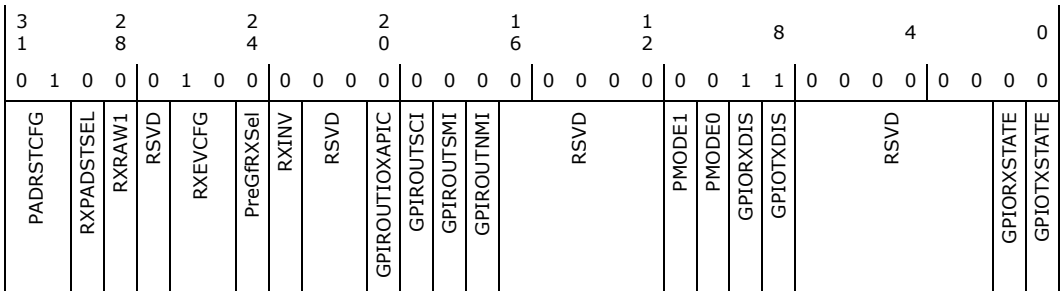
18.7.136 Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_20)—Offset 560h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

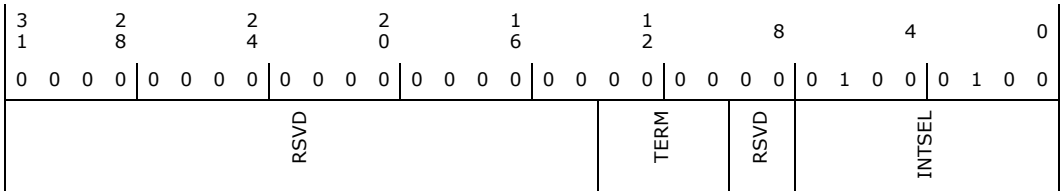
18.7.137 Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_20)—Offset 564h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	44h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

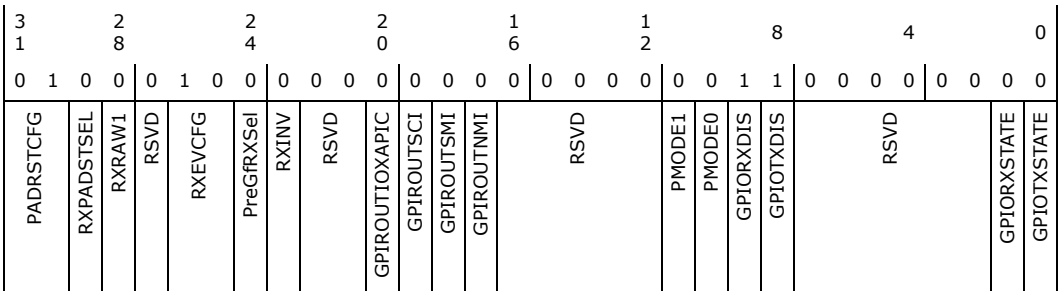
18.7.138 Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_21)—Offset 568h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

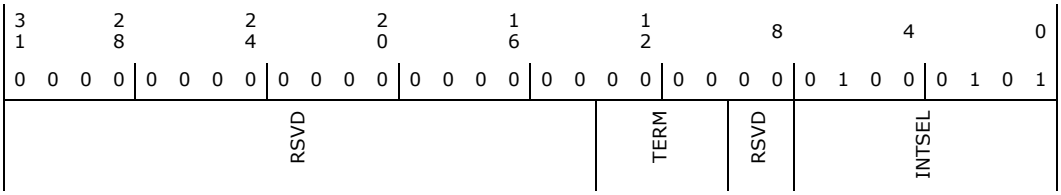
18.7.139 Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_21)—Offset 56Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 45h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	45h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

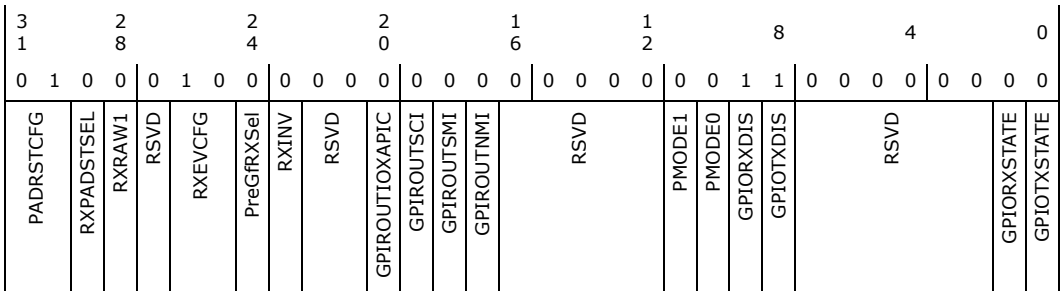
18.7.140 Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_22)—Offset 570h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

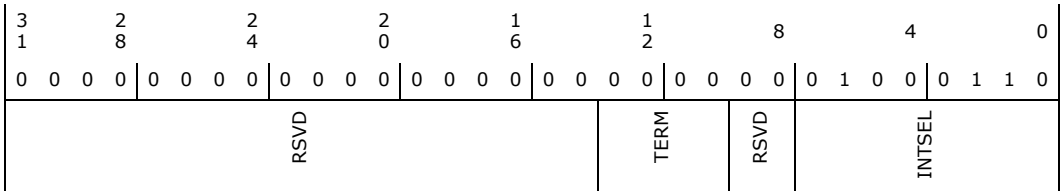
18.7.141 Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_22)—Offset 574h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 46h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	46h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

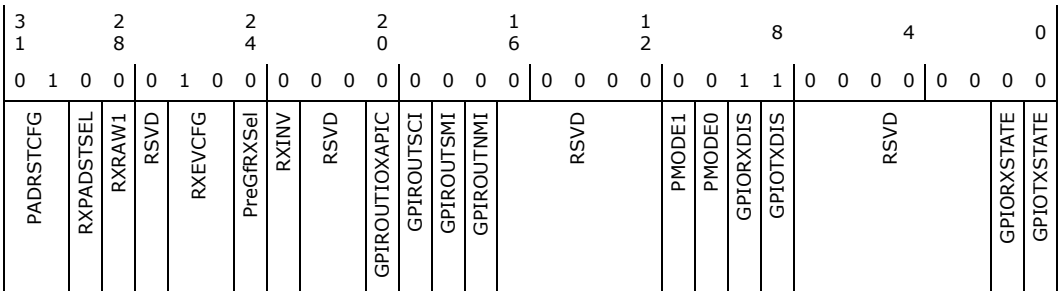
18.7.142 Pad Configuration DW0 (PAD_CFG_DW0_GPP_D_23)—Offset 578h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

18.7.143 Pad Configuration DW1 (PAD_CFG_DW1_GPP_D_23)—Offset 57Ch

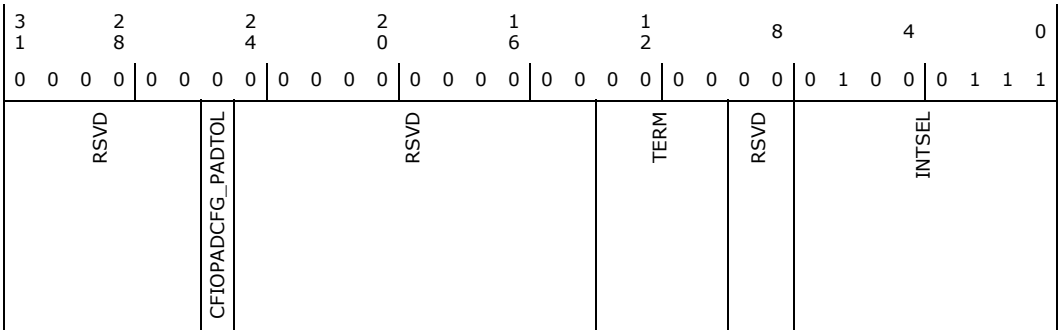
Same description as PAD_CFG_DW1_GPP_C_16 Exception: The default value of the INTSEL bit field in this register is 47h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 47h



Bit Range	Default & Access	Field Name (ID): Description
31:26	0h RO	Reserved
25	0h RW	**Pad Voltage Tolerance (CFIOPADCFG_PADTOL):** 0 = pad is 3.3V tolerance (VccIO is 3.3V) 1 = pad is 1.8V tolerance (VccIO is 3.3V).
24:14	0h RO	Reserved.
13:10	0h RO	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.

Bit Range	Default & Access	Field Name (ID): Description
9:8	0h RO	Reserved
7:0	47h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

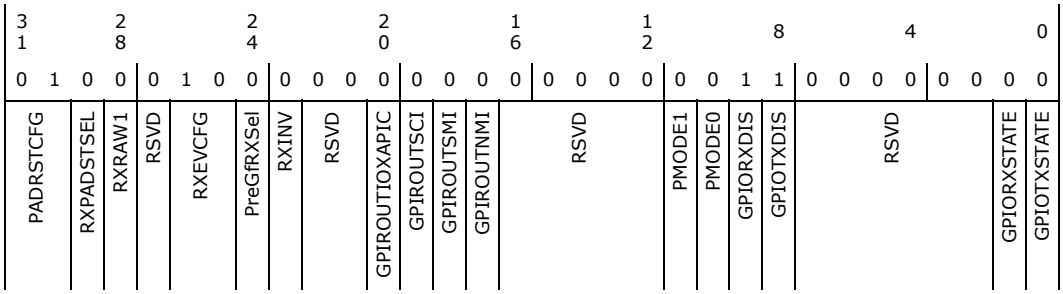
18.7.144 Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_0)—Offset 580h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.

Bit Range	Default & Access	Field Name (ID): Description
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

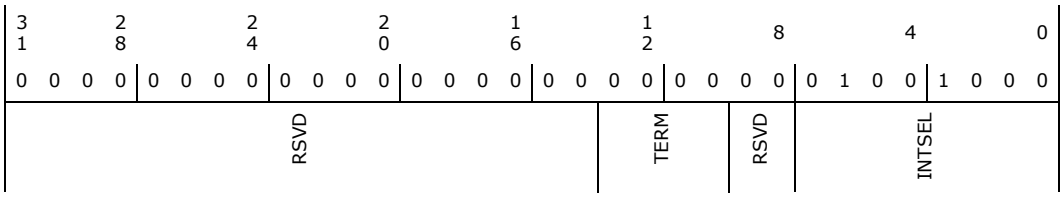
18.7.145 Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_0)—Offset 584h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 48h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	48h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

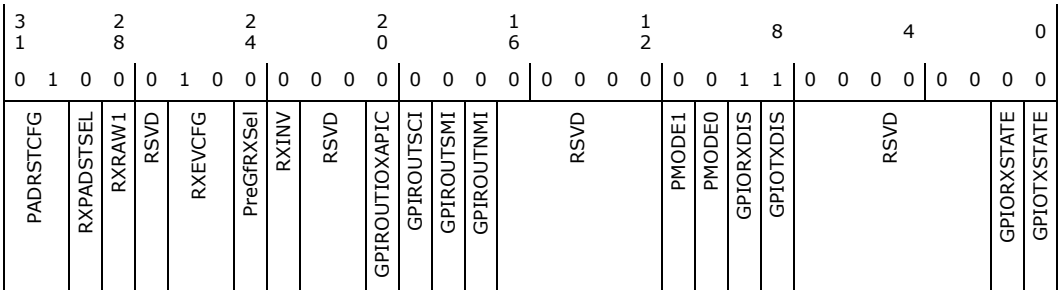
18.7.146 Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_1)—Offset 588h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

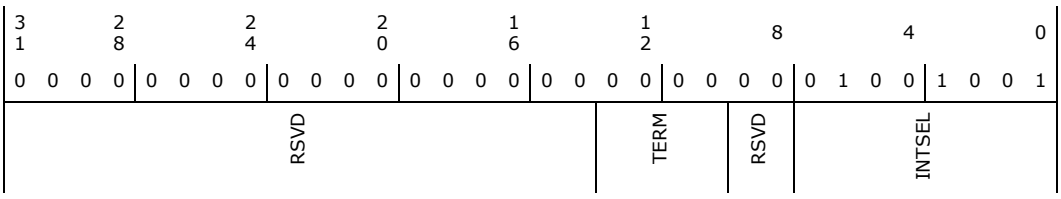
18.7.147 Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_1)—Offset 58Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 49h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	49h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.7.148 Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_2)—Offset 590h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG		RXPADSTSEL		RXRAW1		RSVD		RXEVCFG		PreGfRXSel		RXINV		RSVD		GPIROUTIOXAPIC		GPIROUTSCI		GPIROUTSMI		GPIROUTNMI		RSVD		PMODE1		PMODE0		GPIORXDIS		GPIOTXDIS		RSVD		GPIORXSTATE		GPIOTXSTATE	

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

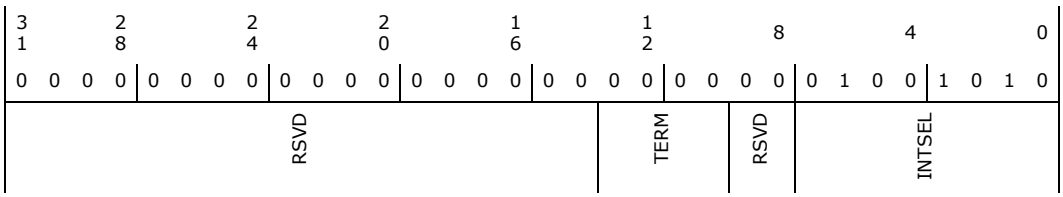
18.7.149 Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_2)—Offset 594h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 4Ah



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	4Ah RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

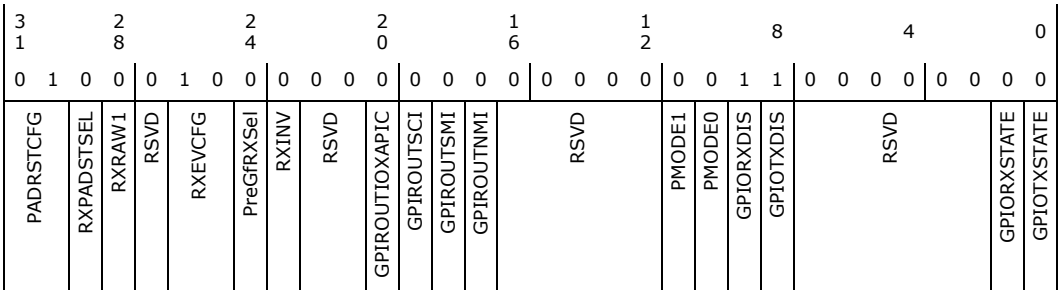
18.7.150 Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_3)—Offset 598h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

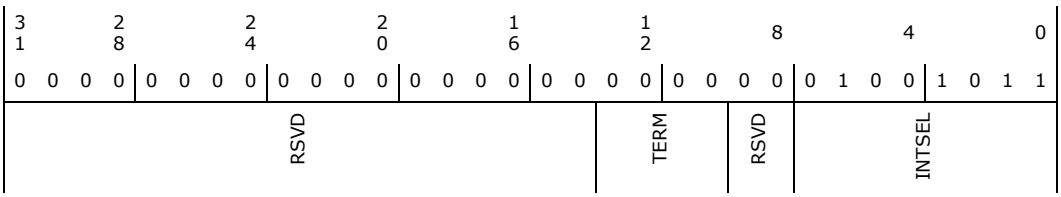
18.7.151 Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_3)—Offset 59Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 4Bh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	4Bh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.7.152 Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_4)—Offset 5A0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

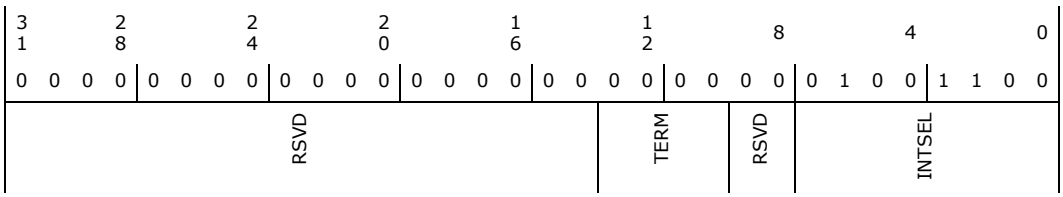
18.7.153 Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_4)—Offset 5A4h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 4Ch



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	4Ch RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

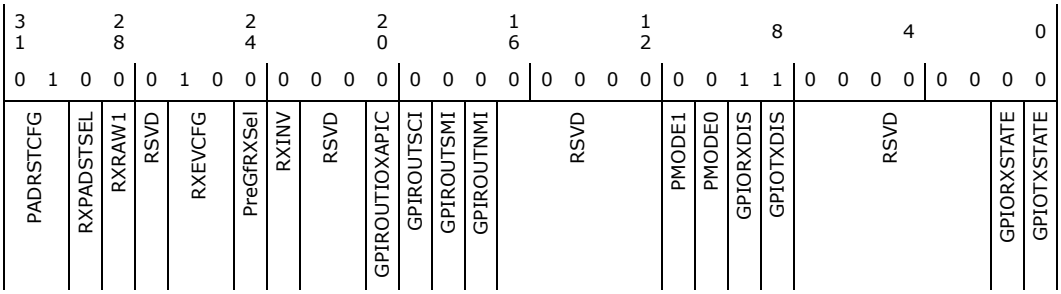
18.7.154 Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_5)—Offset 5A8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

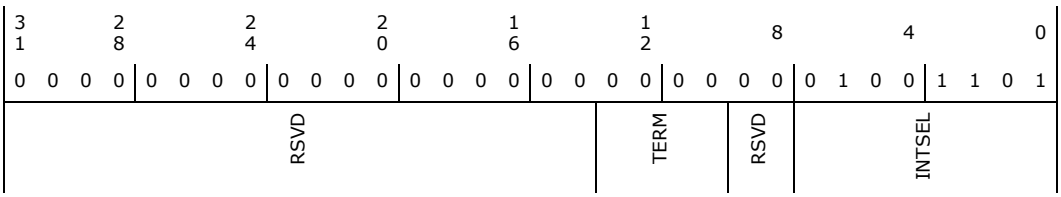
18.7.155 Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_5)—Offset 5ACh

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 4Dh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	4Dh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.7.156 Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_6)—Offset 5B0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

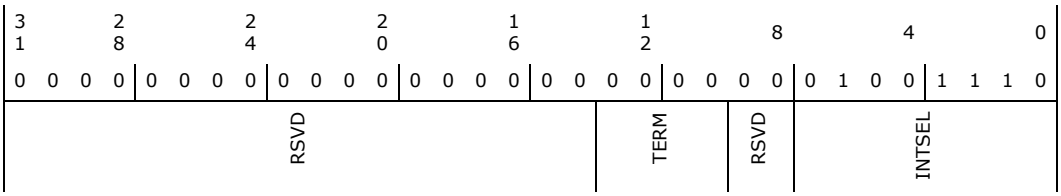
18.7.157 Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_6)—Offset 5B4h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 4Eh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	4Eh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

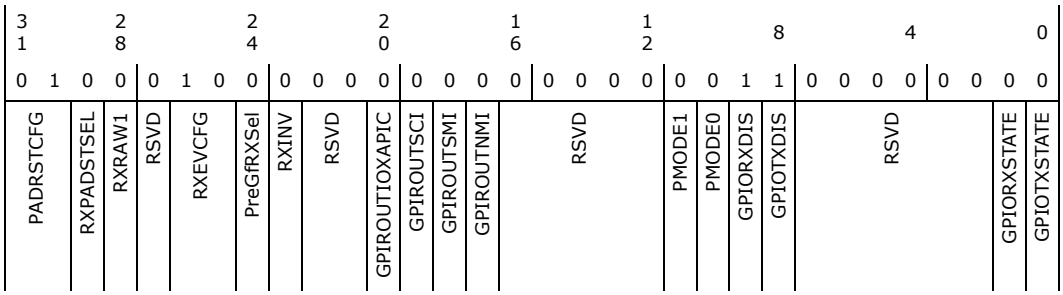
18.7.158 Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_7)—Offset 5B8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

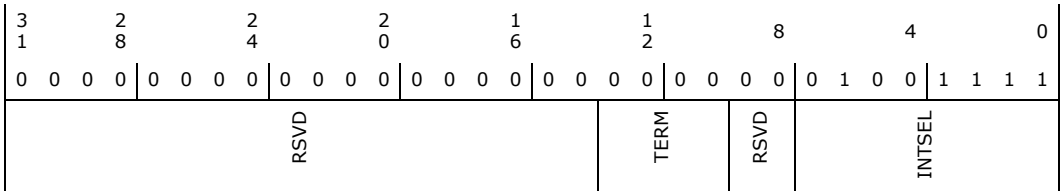
18.7.159 Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_7)—Offset 5BCh

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 4Fh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	4Fh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

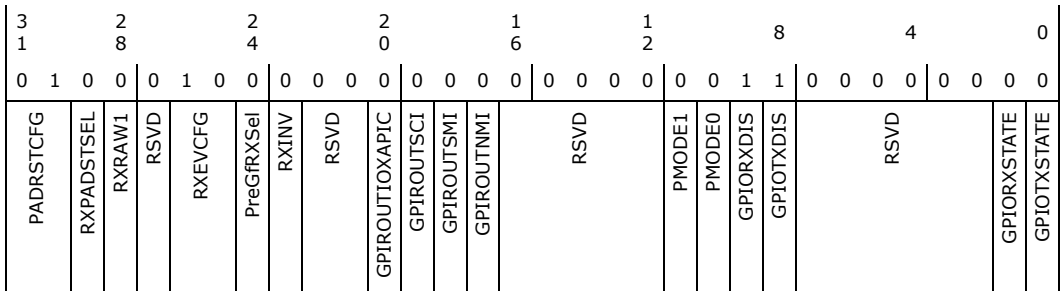
18.7.160 Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_8)—Offset 5C0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

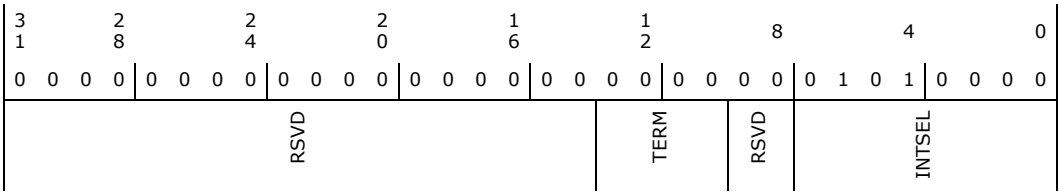
18.7.161 Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_8)—Offset 5C4h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 50h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	50h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

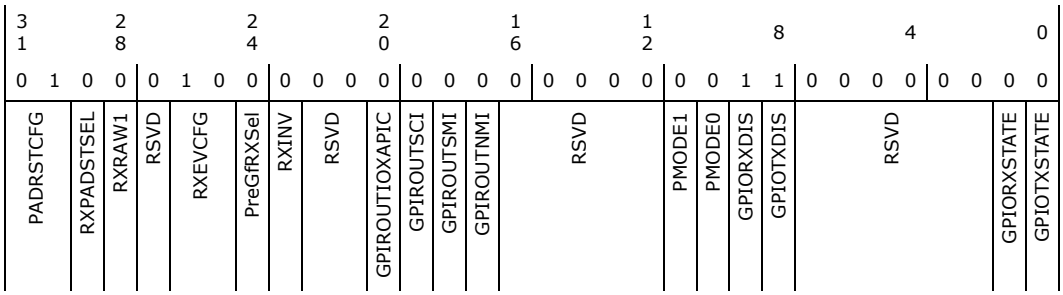
18.7.162 Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_9)—Offset 5C8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

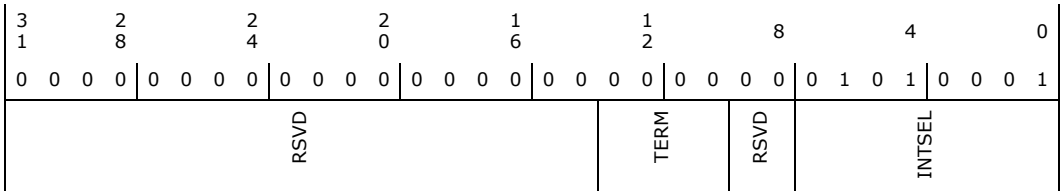
18.7.163 Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_9)—Offset 5CCh

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 51h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	51h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

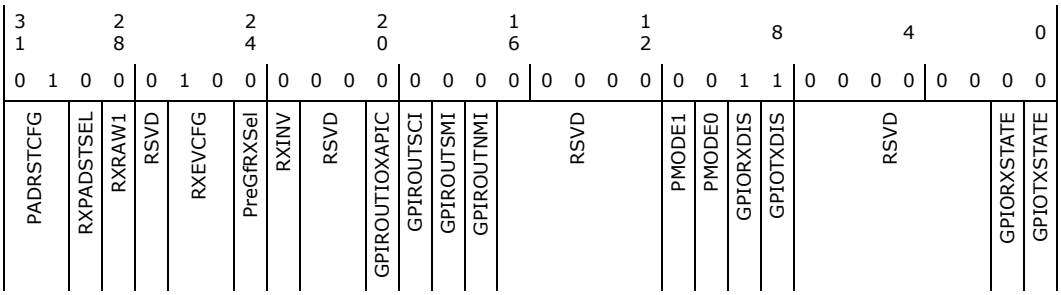
18.7.164 Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_10)—Offset 5D0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

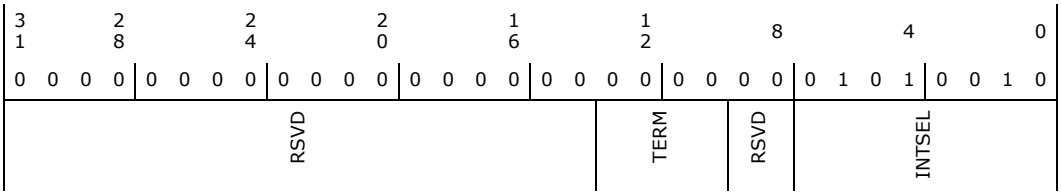
18.7.165 Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_10)—Offset 5D4h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 52h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	52h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

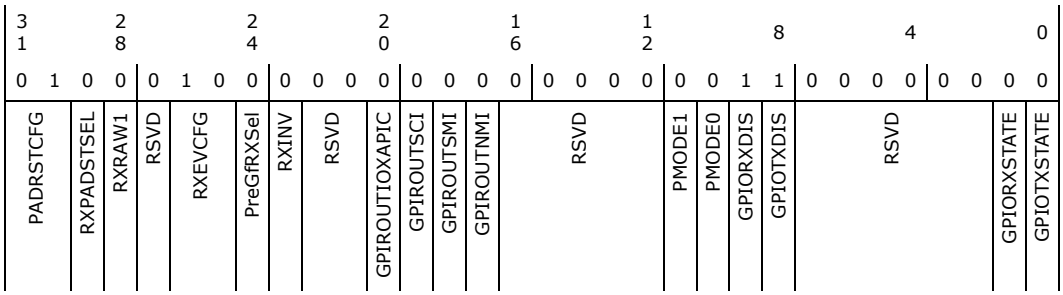
18.7.166 Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_11)—Offset 5D8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0’s in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

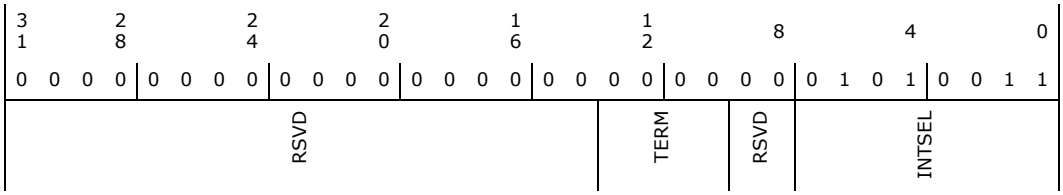
18.7.167 Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_11)—Offset 5DCh

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 53h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	53h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

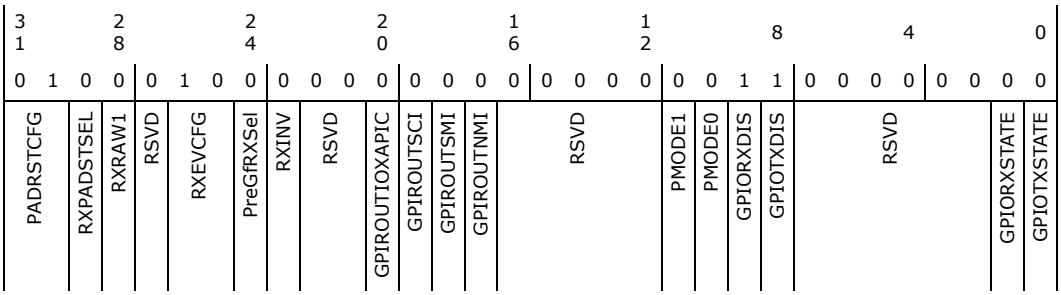
18.7.168 Pad Configuration DW0 (PAD_CFG_DW0_GPP_E_12)—Offset 5E0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad.

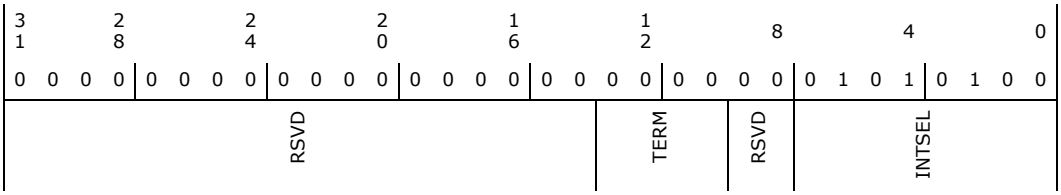
18.7.169 Pad Configuration DW1 (PAD_CFG_DW1_GPP_E_12)—Offset 5E4h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 54h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	54h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.8 GPIO Community 2 Registers Summary

Community 2 Registers are for GPP_DSW group.

Table 18-5. Summary of GPIO Community 2 Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
Ch	Fh	Pad Base Address (PADBAR)—Offset Ch	00000400h
10h	13h	Miscellaneous Configuration (MISCCFG)—Offset 10h	00043900h
20h	23h	Pad Ownership (PAD_OWN_GPD_0)—Offset 20h	00000000h
24h	27h	Pad Ownership (PAD_OWN_GPD_1)—Offset 24h	00000000h
60h	63h	Pad Configuration Lock (PADCFGLOCK_GPD_0)—Offset 60h	00000000h
64h	67h	Pad Configuration Lock (PADCFGLOCKTX_GPD_0)—Offset 64h	00000000h
80h	83h	Host Software Pad Ownership (HOSTSW_OWN_GPD_0)—Offset D0h	00000000h
100h	103h	GPI Interrupt Status (GPI_IS_GPD_0)—Offset 100h	00000000h
110h	113h	GPI Interrupt Enable (GPI_IE_GPD_0)—Offset 110h	00000000h
120h	123h	GPI General Purpose Events Status (GPI_GPE_STS_GPD_0)—Offset 120h	00000000h
130h	133h	GPI General Purpose Events Enable (GPI_GPE_EN_GPD_0)—Offset 130h	00000000h
400h	403h	Pad Configuration DW0 (PAD_CFG_DW0_GPD_0)—Offset 400h	4000700h
404h	407h	Pad Configuration DW1 (PAD_CFG_DW1_GPD_0)—Offset 404h	00000018h
408h	40Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPD_1)—Offset 408h	4000700h
40Ch	40Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPD_1)—Offset 40Ch	00003C19h
410h	413h	Pad Configuration DW0 (PAD_CFG_DW0_GPD_2)—Offset 410h	4000700h
414h	417h	Pad Configuration DW1 (PAD_CFG_DW1_GPD_2)—Offset 414h	00003C1Ah
418h	41Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPD_3)—Offset 418h	4000700h
41Ch	41Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPD_3)—Offset 41Ch	00003C1Bh
420h	423h	Pad Configuration DW0 (PAD_CFG_DW0_GPD_4)—Offset 420h	4000600h
424h	427h	Pad Configuration DW1 (PAD_CFG_DW1_GPD_4)—Offset 424h	0000001Ch
428h	42Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPD_5)—Offset 428h	4000600h
42Ch	42Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPD_5)—Offset 42Ch	0000001Dh
430h	433h	Pad Configuration DW0 (PAD_CFG_DW0_GPD_6)—Offset 430h	4000600h
434h	437h	Pad Configuration DW1 (PAD_CFG_DW1_GPD_6)—Offset 434h	0000001Eh
438h	43Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPD_7)—Offset 438h	4000601h
43Ch	43Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPD_7)—Offset 43Ch	0000001Fh
440h	443h	Pad Configuration DW0 (PAD_CFG_DW0_GPD_8)—Offset 440h	4000700h
444h	447h	Pad Configuration DW1 (PAD_CFG_DW1_GPD_8)—Offset 444h	00000020h
448h	44Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPD_9)—Offset 448h	4000600h
44Ch	44Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPD_9)—Offset 44Ch	00000021h
450h	453h	Pad Configuration DW0 (PAD_CFG_DW0_GPD_10)—Offset 450h	4000600h
454h	457h	Pad Configuration DW1 (PAD_CFG_DW1_GPD_10)—Offset 454h	00000022h
458h	45Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPD_11)—Offset 458h	4000600h
45Ch	45Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPD_11)—Offset 45Ch	00000023h

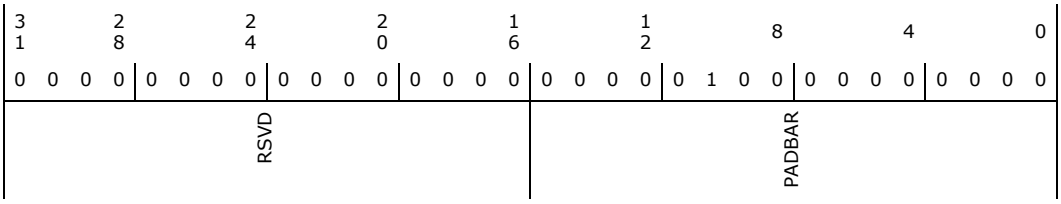
18.8.1 Pad Base Address (PADBAR)—Offset Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 400h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0h RO	Reserved
15:0	400h RO	**Pad Base Address (PADBAR):** This field provides the starting byte-align address of Pad0 register sets. It is meant for software to discover from where the very first Pad register (i.e., Pad0 register) starts to compute the next Pad address offsets.

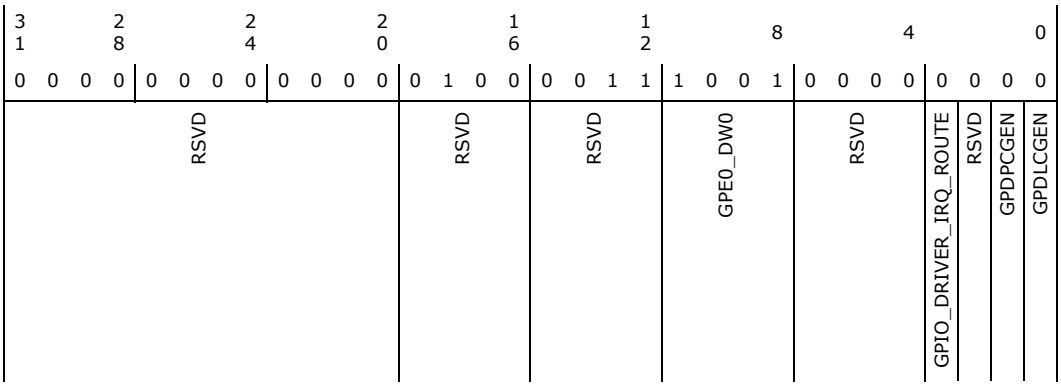
18.8.2 Miscellaneous Configuration (MISCCFG)—Offset 10h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00043900h



Bit Range	Default & Access	Field Name (ID): Description
31:20	0h RO	Reserved
19:16	4h RO	Reserved
15:12	3h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
11:8	9h RO	**GPIO Group to GPE_DW0 assignment encoding (GPE0_DW0):** This register assigns a specific GPIO Group to the ACPI GPE0[31:0]. 0h = GPP_A[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 1h = GPP_B[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 2h = GPP_C[23:0] mapped to GPE[23:0] and GPE[31:24] is not used] 3h = GPP_D[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 4h = GPP_E[12:0] mapped to GPE[12:0] and GPE[31:13] is not used 5h = GPP_F[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 6h = GPP_G[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 7h = GPP_H[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 8h = GPP_I[10:0] mapped to GPE[10:0] and GPE[31:11] is not used 9h = GPD[11:0] mapped to GPE[11:0] and GPE[31:12] is not used Ah= GPP_J[23:0] mapped to GPE[23:0] and GPE[31:24] not used Bh= GPP_K[11:0]mapped to GPE[11:0] and GPE[31:12] is not use
7:4	0h RO	Reserved
3	0h RW	**GPIO Driver IRQ Route (GPIO_DRIVER_IRQ_ROUTE):** GPIO Driver IRQ_ROUTE[1:0]: Specifies the APIC IRQ globally for all pads within the current community (GPI_IS with corresponding GPI_IE enable). 0 = IRQ14 1 = IRQ15
2	0h RO	Reserved
1	0h RW	**GPIO Dynamic Partition Clock Gating Enable (GPDPCGEN):** Specifies whether the GPIO Community should take part in partition clock gating 0 = Disable participation in dynamic partition clock gating 1 = Enable participation in dynamic partition clock gating.
0	0h RW	**GPIO Dynamic Local Clock Gating Enable (GPDLCGEN):** Specifies whether the GPIO Community should perform local clock gating. 0 = Disable dynamic local clock gating 1 = Enable dynamic local clock gating

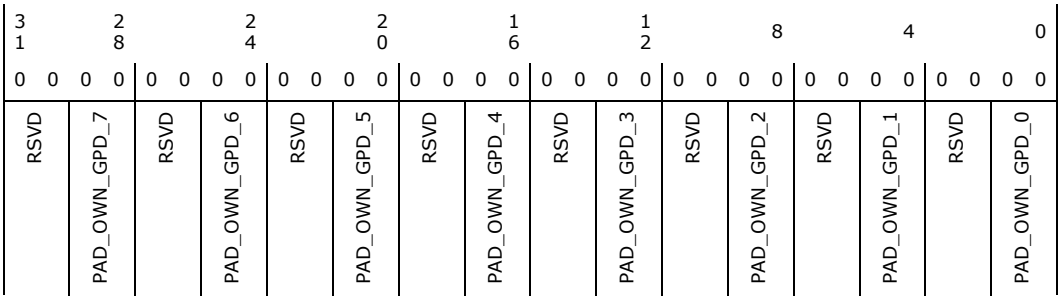
18.8.3 Pad Ownership (PAD_OWN_GPD_0)—Offset 20h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RO	**Pad Ownership (PAD_OWN_GPD_7):** Same description as bit 0, except that the bit field applies to GPD_7.
27:26	0h RO	Reserved
25:24	0h RO	**Pad Ownership (PAD_OWN_GPD_6):** Same description as bit 0, except that the bit field applies to GPD_6.
23:22	0h RO	Reserved
21:20	0h RO	**Pad Ownership (PAD_OWN_GPD_5):** Same description as bit 0, except that the bit field applies to GPD_5.
19:18	0h RO	Reserved
17:16	0h RO	**Pad Ownership (PAD_OWN_GPD_4):** Same description as bit 0, except that the bit field applies to GPD_4.
15:14	0h RO	Reserved
13:12	0h RO	**Pad Ownership (PAD_OWN_GPD_3):** Same description as bit 0, except that the bit field applies to GPD_3.
11:10	0h RO	Reserved
9:8	0h RO	**Pad Ownership (PAD_OWN_GPD_2):** Same description as bit 0, except that the bit field applies to GPD_2.
7:6	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
5:4	0h RO	**Pad Ownership (PAD_OWN_GPD_1):** Same description as bit 0, except that the bit field applies to GPD_1.
3:2	0h RO	Reserved
1:0	0h RO	**Pad Ownership (PAD_OWN_GPD_0):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = ME GPIO Mode. ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad

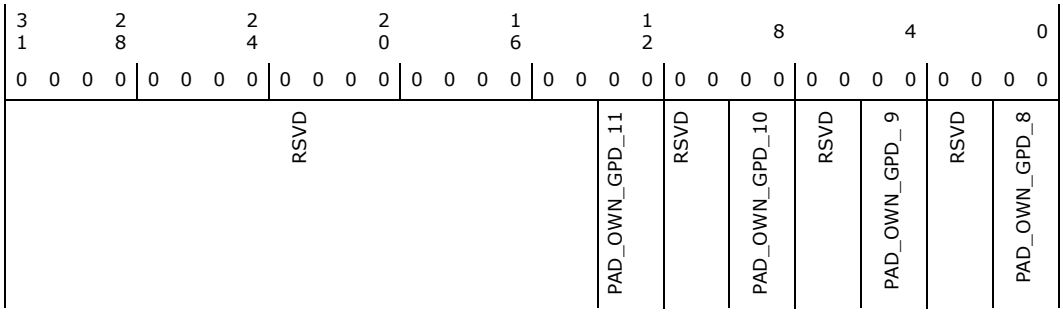
18.8.4 Pad Ownership (PAD_OWN_GPD_1)—Offset 24h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0000h RO	Reserved
13:12	0h RO	**Pad Ownership (PAD_OWN_GPD_11):** Same description as bit 0, except that the bit field applies to GPD_3.
11:10	0h RO	Reserved
9:8	0h RO	**Pad Ownership (PAD_OWN_GPD_10):** Same description as bit 0, except that the bit field applies to GPD_2.
7:6	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
5:4	0h RO	**Pad Ownership (PAD_OWN_GPD_9):** Same description as bit 0, except that the bit field applies to GPD_1.
3:2	0h RO	Reserved
1:0	0h RO	**Pad Ownership (PAD_OWN_GPD_8):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = ME GPIO Mode. ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad

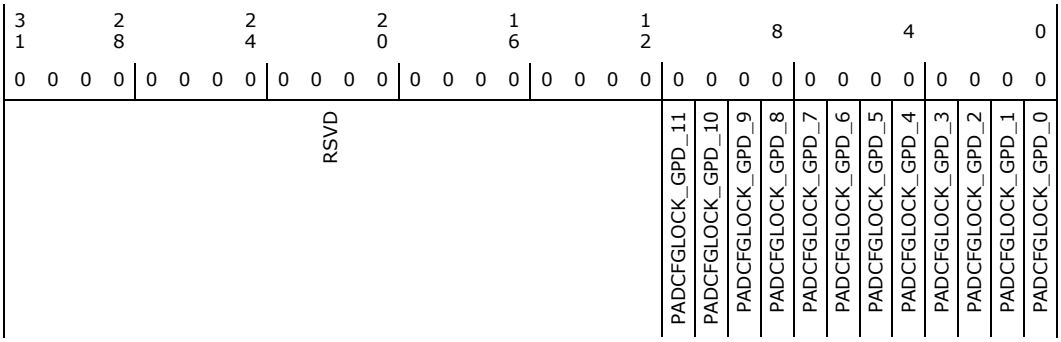
18.8.5 Pad Configuration Lock (PADCFGLOCK_GPD_0)—Offset 60h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:12	0h RO	Reserved
11	0h RW	**Pad Config Lock (PADCFGLOCK_GPD_11):** Applied to GPD_11. Same description as bit 0.
10	0h RW	**Pad Config Lock (PADCFGLOCK_GPD_10):** Applied to GPD_2. Same description as bit 0.
9	0h RW	**Pad Config Lock (PADCFGLOCK_GPD_9):** Applied to GPD_2. Same description as bit 0.
8	0h RW	**Pad Config Lock (PADCFGLOCK_GPD_8):** Applied to GPD_8. Same description as bit 0.
7	0h RW	**Pad Config Lock (PADCFGLOCK_GPD_7):** Applied to GPD_7. Same description as bit 0.
6	0h RW	**Pad Config Lock (PADCFGLOCK_GPD_6):** Applied to GPD_6. Same description as bit 0.
5	0h RW	**Pad Config Lock (PADCFGLOCK_GPD_5):** Applied to GPD_5. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
4	0h RW	**Pad Config Lock (PADCFGLOCK_GPD_4):** Applied to GPD_4. Same description as bit 0.
3	0h RW	**Pad Config Lock (PADCFGLOCK_GPD_3):** Applied to GPD_3. Same description as bit 0.
2	0h RW	**Pad Config Lock (PADCFGLOCK_GPD_2):** Applied to GPD_2. Same description as bit 0.
1	0h RW	**Pad Config Lock (PADCFGLOCK_GPD_1):** Applied to GPD_2. Same description as bit 0.
0	0h RW	**Pad Config Lock (PADCFGLOCK_GPD_0):** Pad Configuration Lock locks specific register fields in the GPP specific registers from being configured. The registers affected become Read-Only and software writes to these registers have no effect. 0 = Unlock 1 = Lock the following register fields as read-only (RO): - Pad Configuration registers (exclude GPIOTXState) - GPI_NMI_EN Register (if implemented) - GPI_SMI_EN Register (if implemented) - GPI_GPE_EN Register (if implemented) When PadCfgLock is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

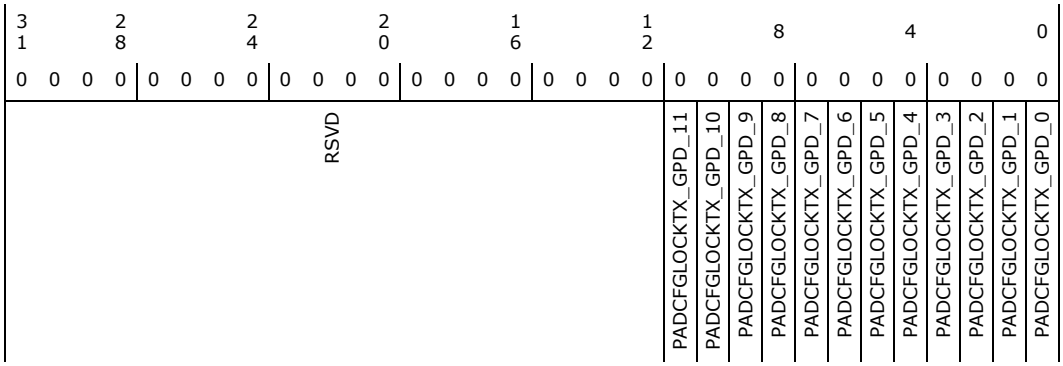
18.8.6 Pad Configuration Lock (PADCFGLOCKTX_GPD_0)—Offset 64h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:12	0h RO	Reserved
11	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPD_11):** Applied to GPD_11. Same description as bit 0.
10	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPD_10):** Applied to GPD_10. Same description as bit 0.
9	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPD_9):** Applied to GPD_9. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
8	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPD_8):** Applied to GPD_8. Same description as bit 0.
7	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPD_7):** Applied to GPD_7. Same description as bit 0.
6	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPD_6):** Applied to GPD_6. Same description as bit 0.
5	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPD_5):** Applied to GPD_5. Same description as bit 0.
4	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPD_4):** Applied to GPD_4. Same description as bit 0.
3	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPD_3):** Applied to GPD_3. Same description as bit 0.
2	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPD_2):** Applied to GPD_2. Same description as bit 0.
1	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPD_1):** Applied to GPD_1. Same description as bit 0.
0	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPD_0):** PadCfgLockTx locks the GPIOTxState bit from being configured. The GPIOTxState register becomes Read-Only and software writes to the register have no effect. 0 = Unlock 1 = Locks the Pad Configuration GPIOTXState field as read-only (RO) When PadCfgLockTx is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

18.8.7 Host Software Pad Ownership (HOSTSW_OWN_GPD_0)—Offset 80h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																				HOSTSW_OWN_GPD_11	HOSTSW_OWN_GPD_10	HOSTSW_OWN_GPD_9	HOSTSW_OWN_GPD_8	HOSTSW_OWN_GPD_7	HOSTSW_OWN_GPD_6	HOSTSW_OWN_GPD_5	HOSTSW_OWN_GPD_4	HOSTSW_OWN_GPD_3	HOSTSW_OWN_GPD_2	HOSTSW_OWN_GPD_1	HOSTSW_OWN_GPD_0

Bit Range	Default & Access	Field Name (ID): Description
31:12	0h RO	Reserved
11	0h RW	**HostSW_Own (HOSTSW_OWN_GPD_11):** Applied to GPD_11. Same description as bit 0.
10	0h RW	**HostSW_Own (HOSTSW_OWN_GPD_10):** Applied to GPD_10. Same description as bit 0.
9	0h RW	**HostSW_Own (HOSTSW_OWN_GPD_9):** Applied to GPD_9. Same description as bit 0.
8	0h RW	**HostSW_Own (HOSTSW_OWN_GPD_8):** Applied to GPD_8. Same description as bit 0.
7	0h RW	**HostSW_Own (HOSTSW_OWN_GPD_7):** Applied to GPD_7. Same description as bit 0.
6	0h RW	**HostSW_Own (HOSTSW_OWN_GPD_6):** Applied to GPD_6. Same description as bit 0.
5	0h RW	**HostSW_Own (HOSTSW_OWN_GPD_5):** Applied to GPD_5. Same description as bit 0.
4	0h RW	**HostSW_Own (HOSTSW_OWN_GPD_4):** Applied to GPD_4. Same description as bit 0.
3	0h RW	**HostSW_Own (HOSTSW_OWN_GPD_3):** Applied to GPD_3. Same description as bit 0.
2	0h RW	**HostSW_Own (HOSTSW_OWN_GPD_2):** Applied to GPD_2. Same description as bit 0.
1	0h RW	**HostSW_Own (HOSTSW_OWN_GPD_1):** Applied to GPD_1. Same description as bit 0.
0	0h RW	**HostSW_Own (HOSTSW_OWN_GPD_0):** This register determines the appropriate host status bit update when a pad is under host ownership. 0 = ACPI Mode. GPIO input event updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. GPI_STS update is masked. 1 = GPIO Driver Mode. GPIO input event updates are limited to GPI_STS. GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS updates are masked.

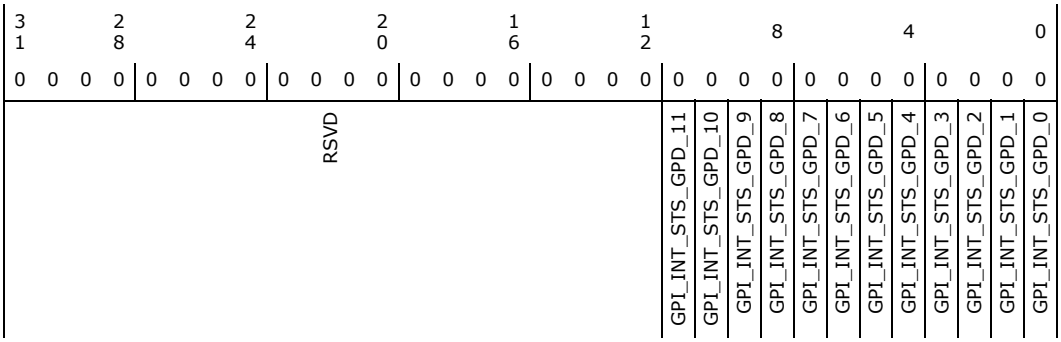
18.8.8 GPI Interrupt Status (GPI_IS_GPD_0)—Offset 100h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:12	0h RO	Reserved
11	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPD_11):** Applied to GPD_11. Same description as bit 0.
10	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPD_10):** Applied to GPD_10. Same description as bit 0.
9	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPD_9):** Applied to GPD_9. Same description as bit 0.
8	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPD_8):** Applied to GPD_8. Same description as bit 0.
7	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPD_7):** Applied to GPD_7. Same description as bit 0.
6	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPD_6):** Applied to GPD_6. Same description as bit 0.
5	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPD_5):** Applied to GPD_5. Same description as bit 0.
4	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPD_4):** Applied to GPD_4. Same description as bit 0.
3	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPD_3):** Applied to GPD_3. Same description as bit 0.
2	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPD_2):** Applied to GPD_2. Same description as bit 0.
1	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPD_1):** Applied to GPD_1. Same description as bit 0.
0	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPD_0):** GPI Interrupt Status (GPI_INT_STS) This bit is set to '1' by hardware when either an edge or a level event is detected (See RxEdCfg RxInv) on pad and all the following conditions are true: - The corresponding pad is used in GPIO input mode - HOSTSW_OWN = 1 (i.e. Host GPIO Driver Mode). Writing a value of 1 will clear the bit while writing a value of 0 has no effect. 0 = No interrupt 1 = Interrupt asserts The state of GPI_INT_EN does not prevent the setting of GPI_INT_STS.

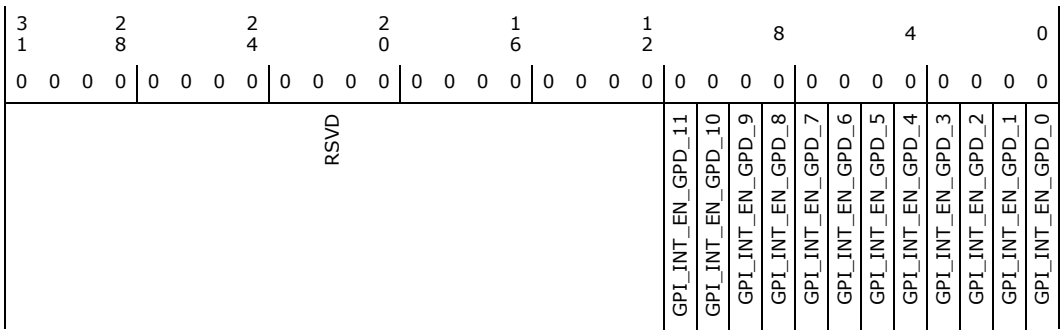
18.8.9 GPI Interrupt Enable (GPI_IE_GPD_0)—Offset 110h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:12	0h RO	Reserved
11	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPD_11):** Applied to GPD_11. Same description as bit 0.
10	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPD_10):** Applied to GPD_10. Same description as bit 0.
9	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPD_9):** Applied to GPD_9. Same description as bit 0.
8	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPD_8):** Applied to GPD_8. Same description as bit 0.
7	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPD_7):** Applied to GPD_7. Same description as bit 0.
6	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPD_6):** Applied to GPD_6. Same description as bit 0.
5	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPD_5):** Applied to GPD_5. Same description as bit 0.
4	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPD_4):** Applied to GPD_4. Same description as bit 0.
3	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPD_3):** Applied to GPD_3. Same description as bit 0.
2	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPD_2):** Applied to GPD_2. Same description as bit 0.
1	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPD_1):** Applied to GPD_1. Same description as bit 0.
0	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPD_0):** This bit is used to enable/disable the generation of APIC interrupt when the corresponding GPI_INT_STS bit is set. 0 = disable interrupt generation 1 = enable interrupt generation Refer to GPI_IRQ_ROUTE for host GPIO Driver Mode interrupt routing.

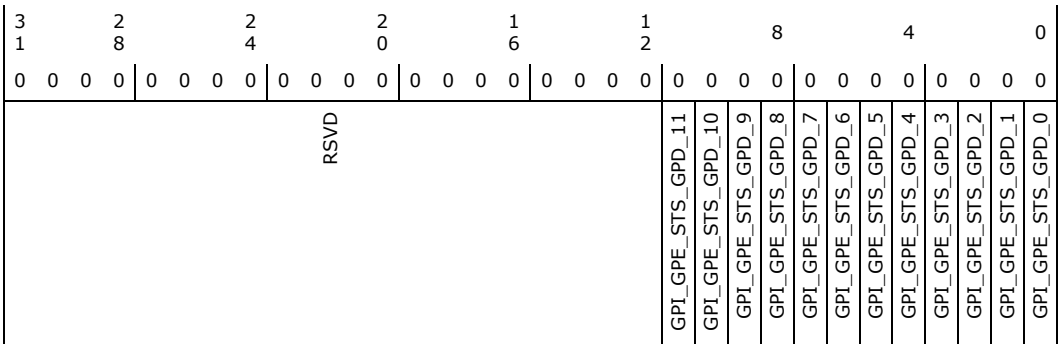
18.8.10 GPI General Purpose Events Status (GPI_GPE_STS_GPD_0)—Offset 120h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:12	0h RO	Reserved
11	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPD_11):** Applied to GPD_11. Same description as bit 0.
10	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPD_10):** Applied to GPD_10. Same description as bit 0.
9	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPD_9):** Applied to GPD_9. Same description as bit 0.
8	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPD_8):** Applied to GPD_8. Same description as bit 0.
7	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPD_7):** Applied to GPD_7. Same description as bit 0.
6	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPD_6):** Applied to GPD_6. Same description as bit 0.
5	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPD_5):** Applied to GPD_5. Same description as bit 0.
4	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPD_4):** Applied to GPD_4. Same description as bit 0.
3	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPD_3):** Applied to GPD_3. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
2	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPD_2):** Applied to GPD_2. Same description as bit 0.
1	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPD_1):** Applied to GPD_1. Same description as bit 0.
0	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPD_0):** These bits are set any time the corresponding GPIO pad is set up as an input, under host ownership and the corresponding GPIO signal is high (or low if the corresponding RXINV bit is set). If the corresponding enable bit is set in the GPI_GPE_EN register, then when the GPI_GPE_STS bit is set: - If the system is in an S3-S5 state, the event will also wake the system. - If the system is in an S0 state (or upon waking back to an S0 state), an SCI will be caused, depending on the GPIRoutSCI bit for the corresponding pad. These bits are sticky bits and are cleared by writing a 1 back to this bit position. The state of GPI_GPE_EN does not prevent the setting of GPI_GPE_STS.

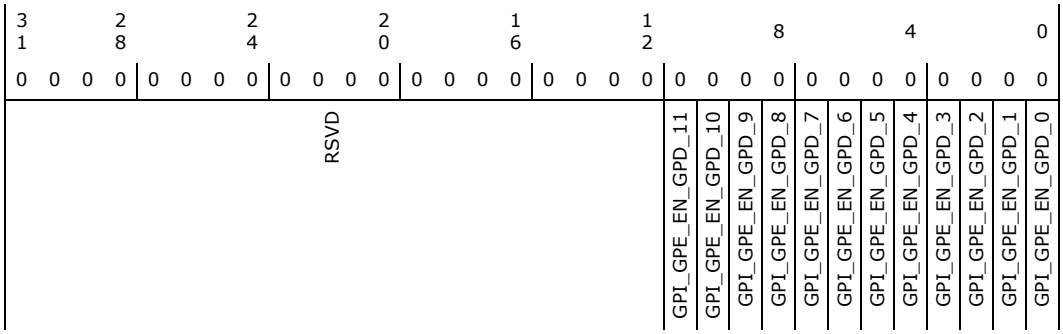
18.8.11 GPI General Purpose Events Enable (GPI_GPE_EN_GPD_0)—Offset 130h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:12	0h RO	Reserved
11	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPD_11):** Applied to GPD_11. Same description as bit 0.
10	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPD_10):** Applied to GPD_10. Same description as bit 0.
9	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPD_9):** Applied to GPD_9. Same description as bit 0.
8	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPD_8):** Applied to GPD_8. Same description as bit 0.
7	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPD_7):** Applied to GPD_7. Same description as bit 0.
6	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPD_6):** Applied to GPD_6. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
5	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPD_5):** Applied to GPD_5. Same description as bit 0.
4	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPD_4):** Applied to GPD_4. Same description as bit 0.
3	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPD_3):** Applied to GPD_3. Same description as bit 0.
2	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPD_2):** Applied to GPD_2. Same description as bit 0.
1	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPD_1):** Applied to GPD_1. Same description as bit 0.
0	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPD_0):** This bit is used to enable/disable the generation of GPE to cause SCI and/or wake when the corresponding GPI_GPE_STS bit is set. 0 = disable GPE generation 1 = enable GPE generation ***Note:*** The pad must also be routed for GPE functionality in order for GPE to be generated, i.e. the corresponding GPIRoutSCI must be set to '1'.

18.8.12 Pad Configuration DW0 (PAD_CFG_DW0_GPD_0)—Offset 400h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 4000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RO	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See description of Pad Mode bit 0.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

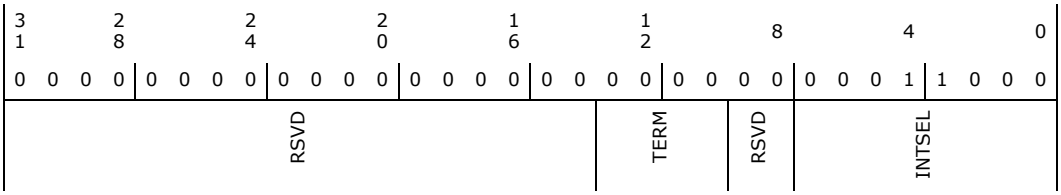
18.8.13 Pad Configuration DW1 (PAD_CFG_DW1_GPD_0)—Offset 404h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 18h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	18h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

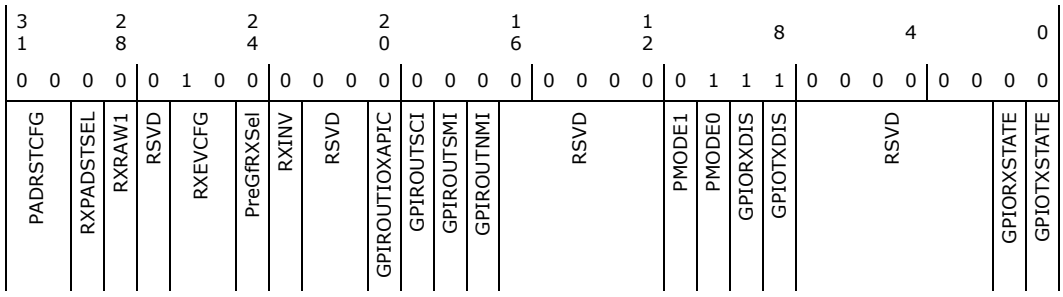
18.8.14 Pad Configuration DW0 (PAD_CFG_DW0_GPD_1)—Offset 408h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 4000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RO	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See description of Pad Mode bit 0.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

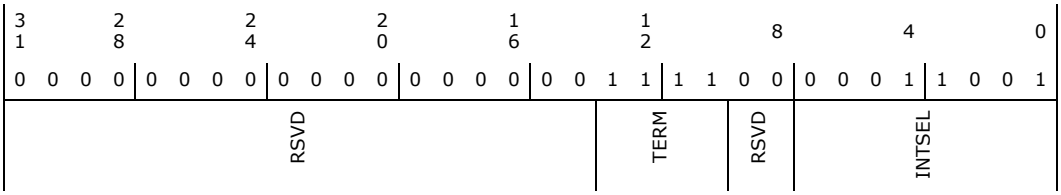
18.8.15 Pad Configuration DW1 (PAD_CFG_DW1_GPD_1)—Offset 40Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 19h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	1111b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu 1111: Native controller selected by pad mode controls termination All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	19h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

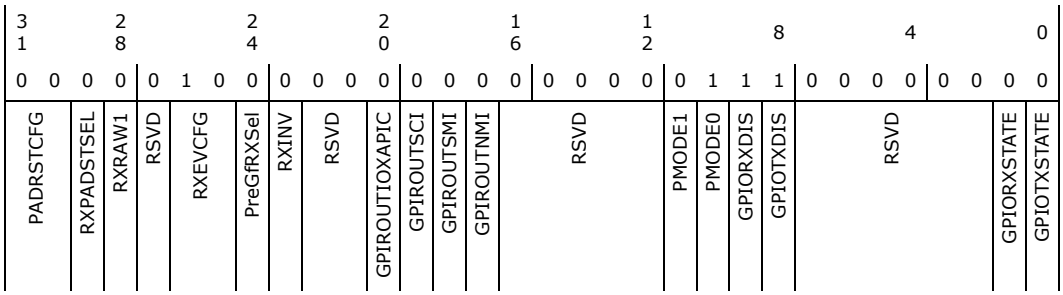
18.8.16 Pad Configuration DW0 (PAD_CFG_DW0_GPD_2)—Offset 410h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 04000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RO	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See description of Pad Mode bit 0.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

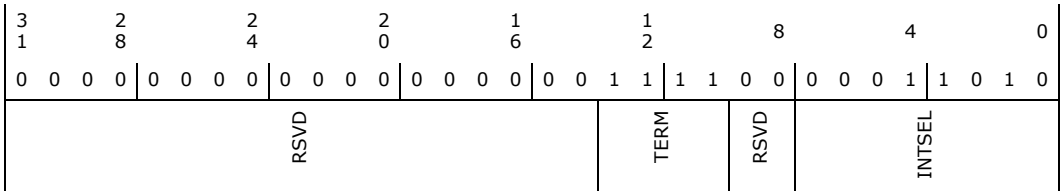
18.8.17 Pad Configuration DW1 (PAD_CFG_DW1_GPD_2)—Offset 414h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003C1Ah



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu 1111: Native controller selected by Pad Mode controls the termination. All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Ah RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

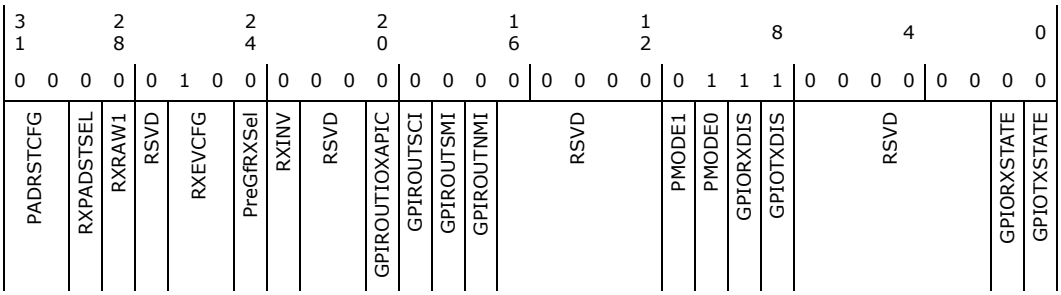
18.8.18 Pad Configuration DW0 (PAD_CFG_DW0_GPD_3)—Offset 418h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 04000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RO	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See description of Pad Mode bit 0.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

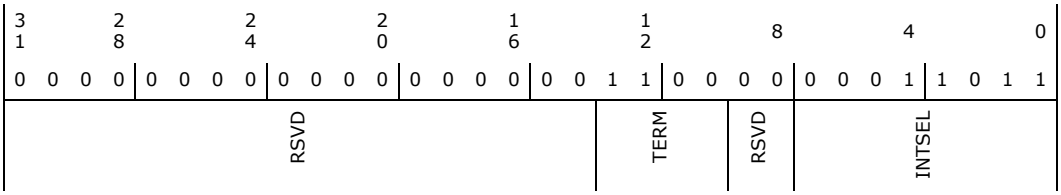
18.8.19 Pad Configuration DW1 (PAD_CFG_DW1_GPD_3)—Offset 41Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0000301Bh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	1100b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Bh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

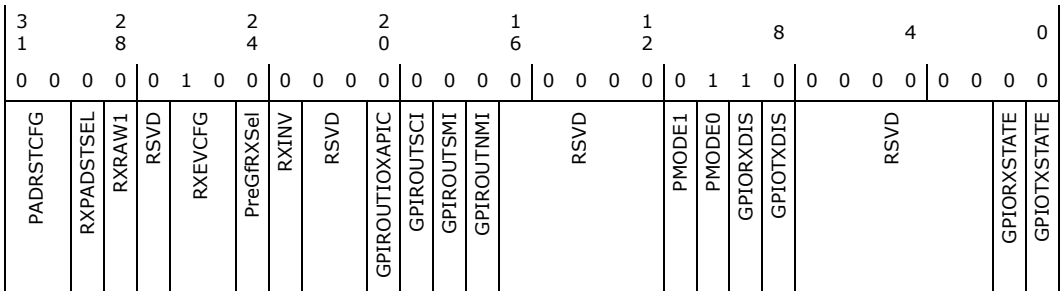
18.8.20 Pad Configuration DW0 (PAD_CFG_DW0_GPD_4)—Offset 420h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 4000600h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RO	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See description of Pad Mode bit 0.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0’s in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

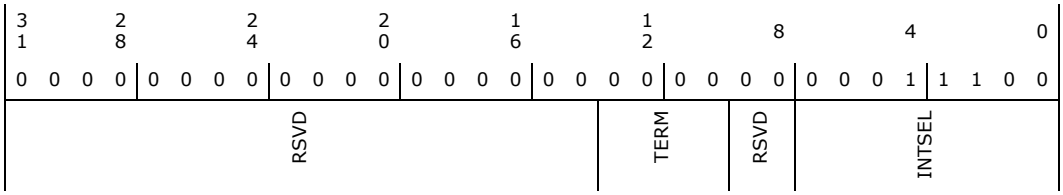
18.8.21 Pad Configuration DW1 (PAD_CFG_DW1_GPD_4)—Offset 424h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Ch



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Ch RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

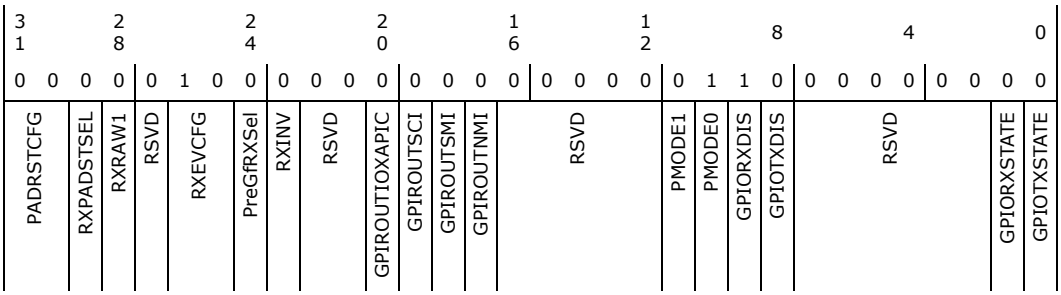
18.8.22 Pad Configuration DW0 (PAD_CFG_DW0_GPD_5)—Offset 428h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 4000600h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RO	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. ***Note:*** This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See description of Pad Mode bit 0.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

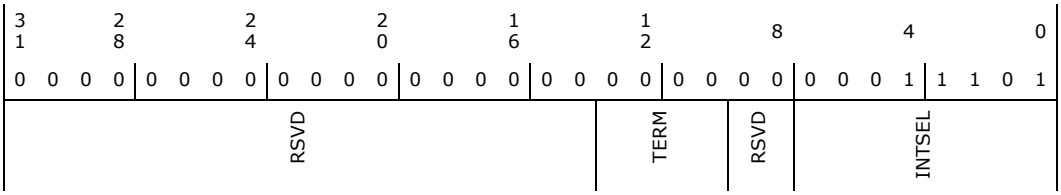
18.8.23 Pad Configuration DW1 (PAD_CFG_DW1_GPD_5)—Offset 42Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Dh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Dh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

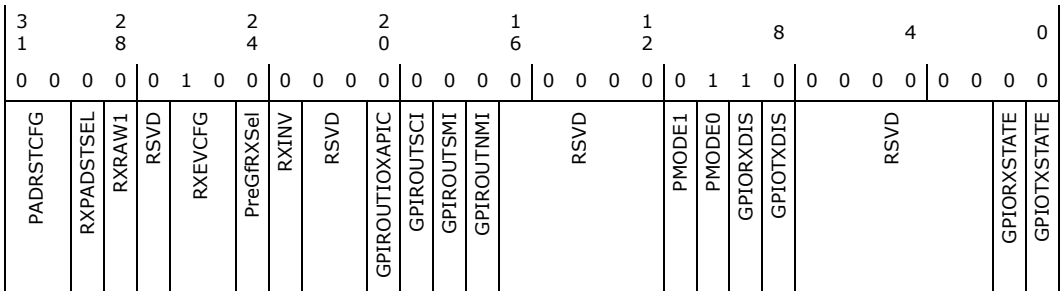
18.8.24 Pad Configuration DW0 (PAD_CFG_DW0_GPD_6)—Offset 430h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 4000600h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RO	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See description of Pad Mode bit 0.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

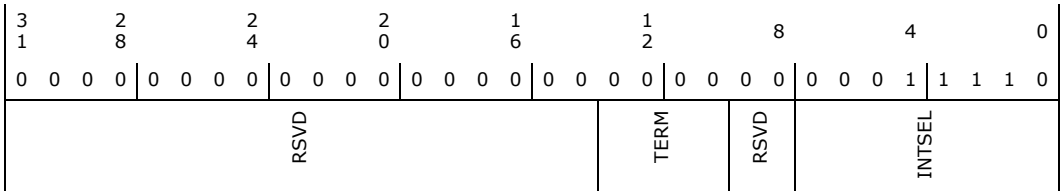
18.8.25 Pad Configuration DW1 (PAD_CFG_DW1_GPD_6)—Offset 434h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Eh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Eh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

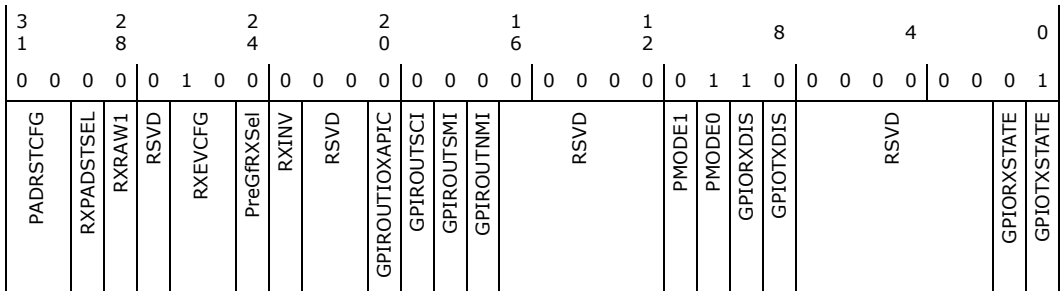
18.8.26 Pad Configuration DW0 (PAD_CFG_DW0_GPD_7)—Offset 438h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 040006001



Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RO	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See description of Pad Mode bit 0.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	1b RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

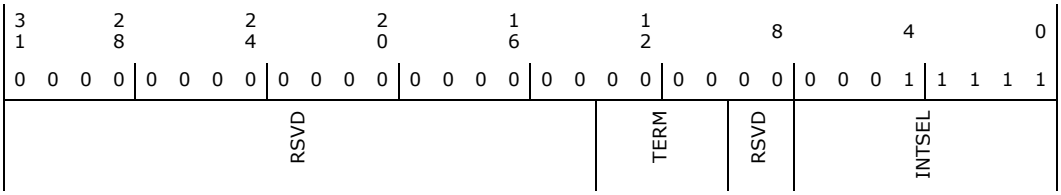
18.8.27 Pad Configuration DW1 (PAD_CFG_DW1_GPD_7)—Offset 43Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Fh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Fh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

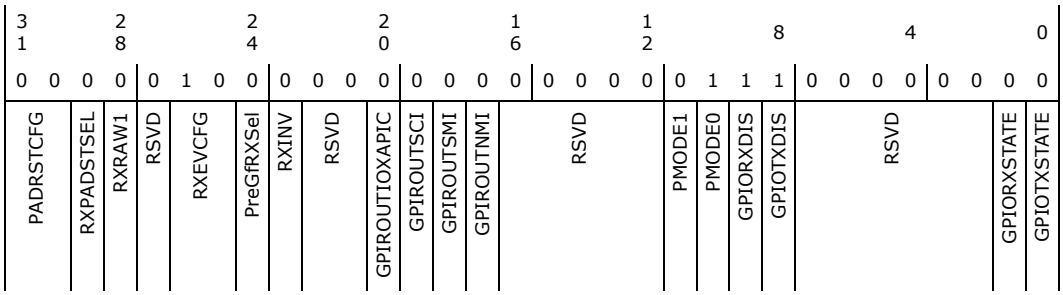
18.8.28 Pad Configuration DW0 (PAD_CFG_DW0_GPD_8)—Offset 440h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 4000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RO	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See description of Pad Mode bit 0.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

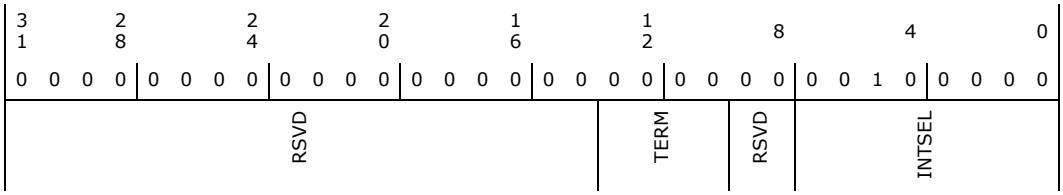
18.8.29 Pad Configuration DW1 (PAD_CFG_DW1_GPD_8)—Offset 444h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 20h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	20h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

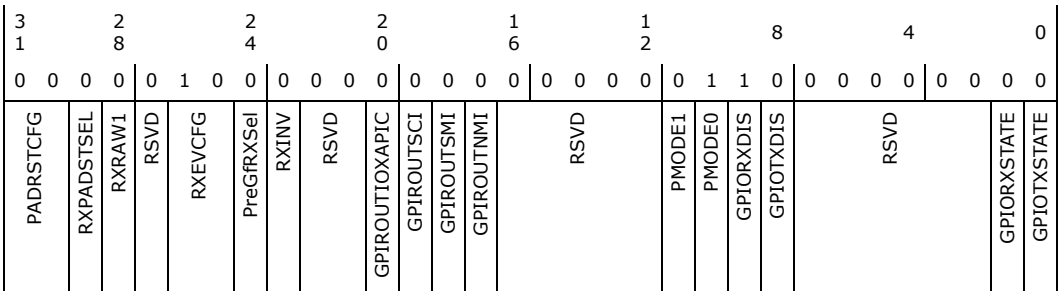
18.8.30 Pad Configuration DW0 (PAD_CFG_DW0_GPD_9)—Offset 448h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 4000600h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RO	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See description of Pad Mode bit 0.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

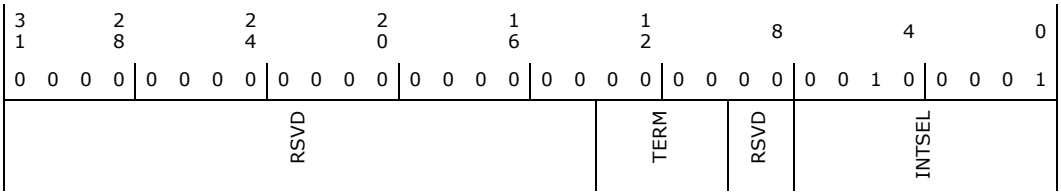
18.8.31 Pad Configuration DW1 (PAD_CFG_DW1_GPD_9)—Offset 44Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 21h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	21h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

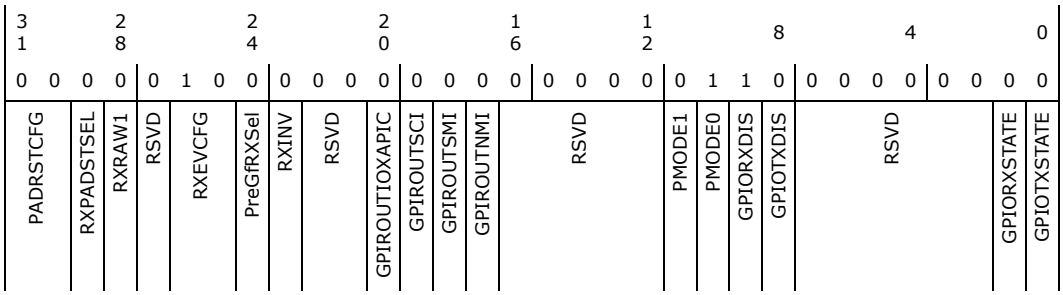
18.8.32 Pad Configuration DW0 (PAD_CFG_DW0_GPD_10)—Offset 450h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 4000600h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RO	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See description of Pad Mode bit 0.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

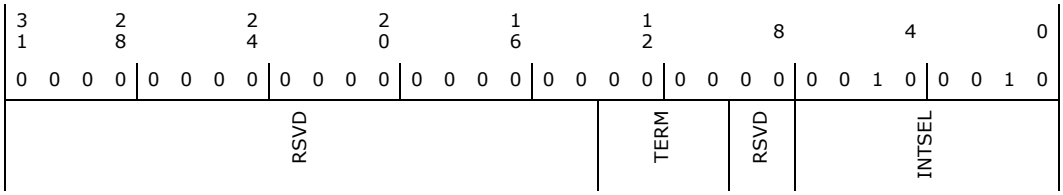
18.8.33 Pad Configuration DW1 (PAD_CFG_DW1_GPD_10)—Offset 454h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 22h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	22h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

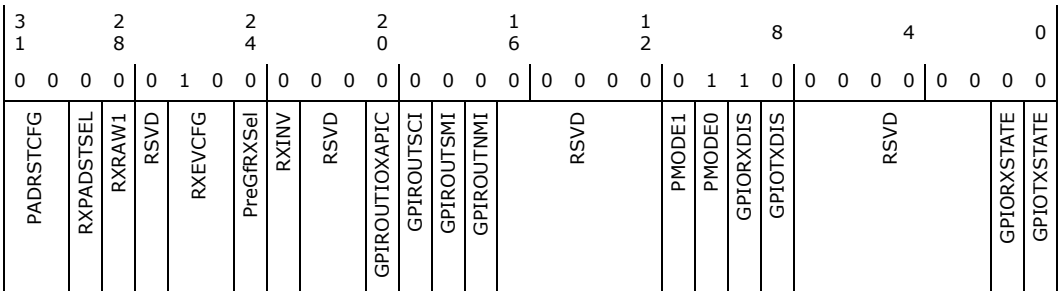
18.8.34 Pad Configuration DW0 (PAD_CFG_DW0_GPD_11)—Offset 458h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 04000600h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RO	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See description of Pad Mode bit 0.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

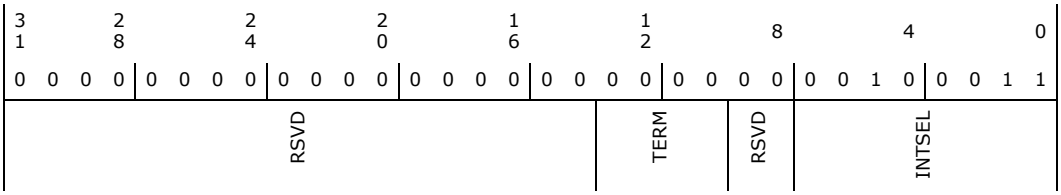
18.8.35 Pad Configuration DW1 (PAD_CFG_DW1_GPD_11)—Offset 45Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 23h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	23h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.9 GPIO Community 3 Registers Summary

Community 3 Registers are for GPP_I group.

Table 18-6. Summary of GPIO Community 3 Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
Ch	Fh	Pad Base Address (PADBAR)—Offset Ch	00000400h
10h	13h	Miscellaneous Configuration (MISCCFG)—Offset 10h	00043800h
20h	23h	Pad Ownership (PAD_OWN_GPP_I_0)—Offset 20h	00000000h
24h	27h	Pad Ownership (PAD_OWN_GPP_I_1)—Offset 24h	00000000h
60h	63h	Pad Configuration Lock (PADCFGLOCK_GPP_I_0)—Offset 60h	00000000h
64h	67h	Pad Configuration Lock (PADCFGLOCKTX_GPP_I_0)—Offset 64h	00000000h
80h	83h	Host Software Pad Ownership (HOSTSW_OWN_GPP_I_0)—Offset 80h	00000000h
100h	103h	GPI Interrupt Status (GPI_IS_GPP_I_0)—Offset 100h	00000000h
110h	113h	GPI Interrupt Enable (GPI_IE_GPP_I_0)—Offset 110h	00000000h
120h	123h	GPI General Purpose Events Status (GPI_GPE_STS_GPP_I_0)—Offset 120h	00000000h
130h	133h	GPI General Purpose Events Enable (GPI_GPE_EN_GPP_I_0)—Offset 130h	00000000h
140h	143h	SMI Status (GPI_SMI_STS_GPP_I_0)—Offset 140h	00000000h
150h	153h	SMI Enable (GPI_SMI_EN_GPP_I_0)—Offset 150h	00000000h
160h	163h	NMI Status (GPI_NMI_STS_GPP_I_0)—Offset 160h	00000000h
170h	173h	NMI Enable (GPI_NMI_EN_GPP_I_0)—Offset 170h	00000000h
400h	403h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_I_0)—Offset 400h	44000B00h
404h	407h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_I_0)—Offset 404h	00000018h
408h	40Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_I_1)—Offset 408h	44000B00h
40Ch	40Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_I_1)—Offset 40Ch	00000019h
410h	413h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_I_2)—Offset 410h	44000B00h
414h	417h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_I_2)—Offset 414h	0000001Ah
418h	41Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_I_3)—Offset 418h	44000B00h
41Ch	41Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_I_3)—Offset 41Ch	0000001Bh
420h	423h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_I_4)—Offset 420h	44000300h
424h	427h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_I_4)—Offset 424h	0000001Ch
428h	42Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_I_5)—Offset 428h	44000300h
42Ch	42Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_I_5)—Offset 42Ch	0000001Dh
430h	433h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_I_6)—Offset 430h	44000300h
434h	437h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_I_6)—Offset 434h	0000001Eh
438h	43Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_I_7)—Offset 438h	44000B00h
43Ch	43Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_I_7)—Offset 43Ch	0000001Fh
440h	443h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_I_8)—Offset 440h	44000B00h
444h	447h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_I_8)—Offset 444h	00003C20h
448h	44Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_I_9)—Offset 448h	44000B00h
44Ch	44Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_I_9)—Offset 44Ch	00003C21h
450h	453h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_I_10)—Offset 450h	44000300h
454h	457h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_I_10)—Offset 454h	00000022h

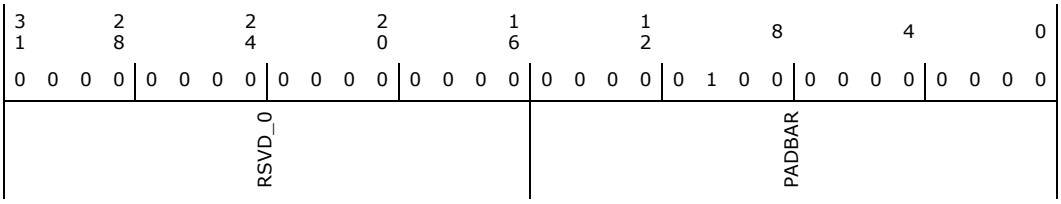
18.9.1 Pad Base Address (PADBAR)—Offset Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 400h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0h RO	**Reserved (RSVD_0)**
15:0	400h RO	**Pad Base Address (PADBAR):** This field provides the starting byte-align address of Pad0 register sets. It is meant for software to discover from where the very first Pad register (i.e. Pad0 register) starts to compute the next Pad address offsets.

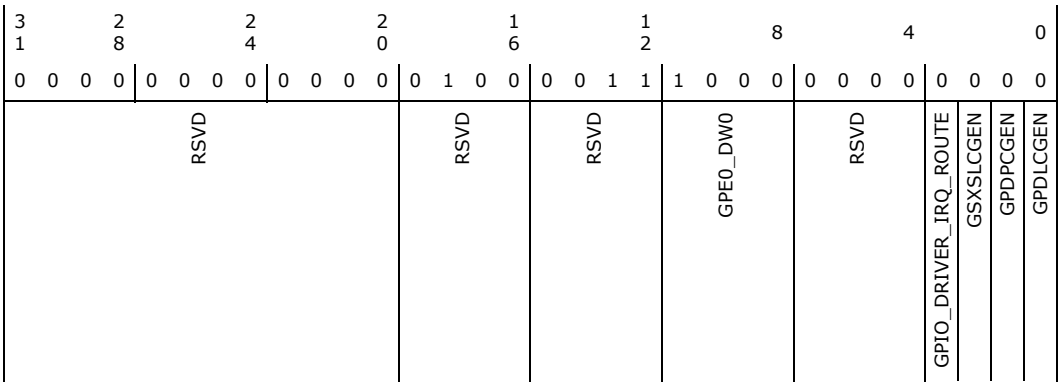
18.9.2 Miscellaneous Configuration (MISCCFG)—Offset 10h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00043800h



Bit Range	Default & Access	Field Name (ID): Description
31:20	0h RO	Reserved.
19:16	4h RO	Reserved
15:12	3h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
11:8	8h RW	**GPIO Group to GPE_DW0 assignment encoding (GPE0_DW0):** This register assigns a specific GPIO Group to the ACPI GPE0[31:0]. 0h = GPP_A[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 1h = GPP_B[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 2h = GPP_C[23:0] mapped to GPE[23:0] and GPE[31:24] is not used] 3h = GPP_D[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 4h = GPP_E[12:0] mapped to GPE[12:0] and GPE[31:13] is not used 5h = GPP_F[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 6h = GPP_G[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 7h = GPP_H[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 8h = GPP_I[10:0] mapped to GPE[10:0] and GPE[31:11] is not used 9h = GPD[11:0] mapped to GPE[11:0] and GPE[31:12] is not used Ah= GPP_J[23:0] mapped to GPE[23:0] and GPE[31:24] not used Bh= GPP_K[11:0]mapped to GPE[11:0] and GPE[31:12] is not used
7:4	0h RO	Reserved
3	0h RW	**GPIO Driver IRQ Route (GPIO_DRIVER_IRQ_ROUTE):** GPIO Driver IRQ_ROUTE[1:0]: Specifies the APIC IRQ globally for all pads within the current community (GPI_IS with corresponding GPI_IE enable). 0 = IRQ14 1 = IRQ15
2	0h RW	**GSX Static Local Clock Gating (GSXSLCGEN):** Specifies whether the GSX controller should be statically clock gated for power saving if it is not enabled (even though the capability is available). 0 = Disable dynamic local clock gating 1 = Enable dynamic local clock gating.
1	0h RW	**GPIO Dynamic Partition Clock Gating Enable (GPDPCGEN):** Specifies whether the GPIO Community should take part in partition clock gating 0 = Disable participation in dynamic partition clock gating 1 = Enable participation in dynamic partition clock gating.
0	0h RW	**GPIO Dynamic Local Clock Gating Enable (GPDLCGEN):** Specifies whether the GPIO Community should perform local clock gating. 0 = Disable dynamic local clock gating 1 = Enable dynamic local clock gating.

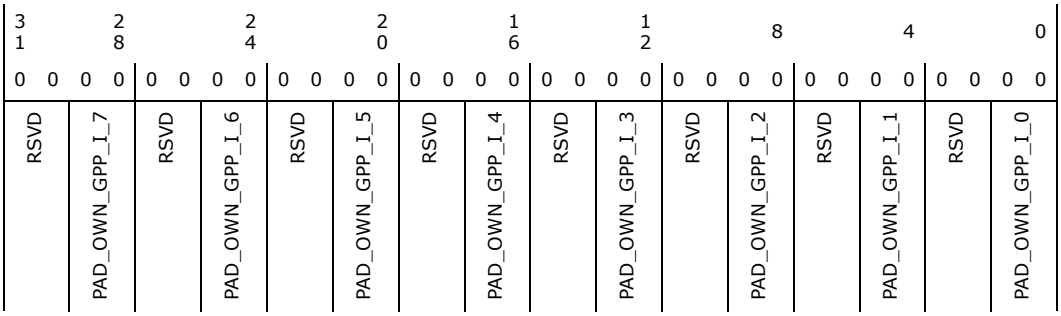
18.9.3 Pad Ownership (PAD_OWN_GPP_I_0)—Offset 20h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RW	**Pad Ownership (PAD_OWN_GPP_I_7):** Same description as bit 0, except that the bit field applies to GPP_I7.
27:26	0h RO	Reserved
25:24	0h RW	**Pad Ownership (PAD_OWN_GPP_I_6):** Same description as bit 0, except that the bit field applies to GPP_I6.
23:22	0h RO	Reserved
21:20	0h RW	**Pad Ownership (PAD_OWN_GPP_I_5):** Same description as bit 0, except that the bit field applies to GPP_I5.
19:18	0h RO	Reserved
17:16	0h RW	**Pad Ownership (PAD_OWN_GPP_I_4):** Same description as bit 0, except that the bit field applies to GPP_I4.
15:14	0h RO	Reserved
13:12	0h RW	**Pad Ownership (PAD_OWN_GPP_I_3):** Same description as bit 0, except that the bit field applies to GPP_I3.
11:10	0h RO	Reserved
9:8	0h RW	**Pad Ownership (PAD_OWN_GPP_I_2):** Same description as bit 0, except that the bit field applies to GPP_I2.
7:6	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
5:4	0h RW	**Pad Ownership (PAD_OWN_GPP_I_1):** Same description as bit 0, except that the bit field applies to GPP_I1.
3:2	0h RO	Reserved
1:0	0h RO	**Pad Ownership (PAD_OWN_GPP_I_0):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = ME GPIO Mode. ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. ME has ownership of the pad

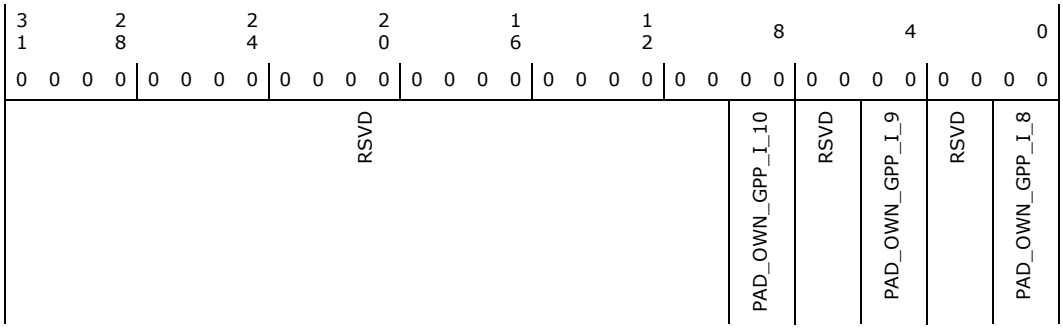
18.9.4 Pad Ownership (PAD_OWN_GPP_I_1)—Offset 24h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:10	000000h RO	Reserved
9:8	0h RW	**Pad Ownership (PAD_OWN_GPP_I_10):** Same description as bit 0, except that the bit field applies to GPP_I_8
7:6	0h RO	Reserved
5:4	0h RW	**Pad Ownership (PAD_OWN_GPP_I_9):** Same description as bit 0, except that the bit field applies to GPP_I_8.
3:2	0h RO	Reserved
1:0	0h RW	**Pad Ownership (PAD_OWN_GPP_I_8):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = ME GPIO Mode. ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. ME has ownership of the pad

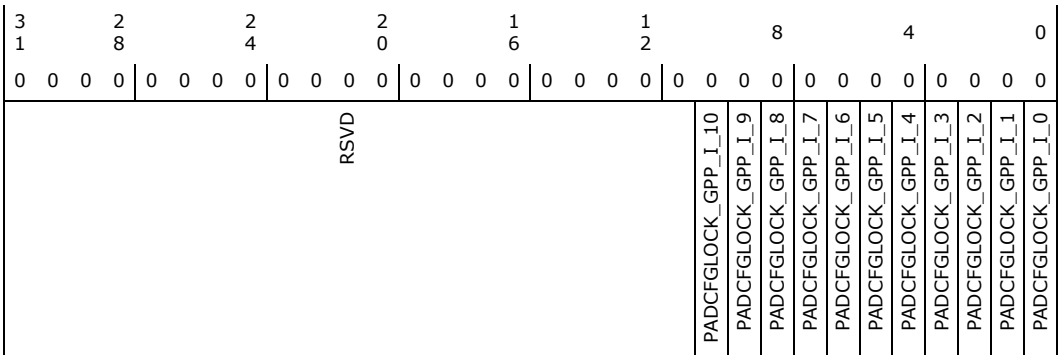
18.9.5 Pad Configuration Lock (PADCFGLOCK_GPP_I_0)—Offset 60h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:11	0h RO	Reserved
10	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_I_10):** Applied to GPP_I10. Same description as bit 0.
9	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_I_9):** Applied to GPP_I9. Same description as bit 0.
8	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_I_8):** Applied to GPP_I8. Same description as bit 0.
7	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_I_7):** Applied to GPP_I7. Same description as bit 0.
6	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_I_6):** Applied to GPP_I6. Same description as bit 0.
5	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_I_5):** Applied to GPP_I5. Same description as bit 0.
4	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_I_4):** Applied to GPP_I4. Same description as bit 0.
3	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_I_3):** Applied to GPP_I3. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
2	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_I_2):** Applied to GPP_I2. Same description as bit 0.
1	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_I_1):** Applied to GPP_I1. Same description as bit 0.
0	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_I_0):** Pad Configuration Lock locks specific register fields in the GPP specific registers from being configured. The registers affected become Read-Only and software writes to these registers have no effect. 0 = Unlock 1 = Lock the following register fields as read-only (RO): - Pad Configuration registers (exclude GPIOTXState) - GPI_NMI_EN Register (if implemented) - GPI_SMI_EN Register (if implemented) - GPI_GPE_EN Register (if implemented) When PadCfgLock is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

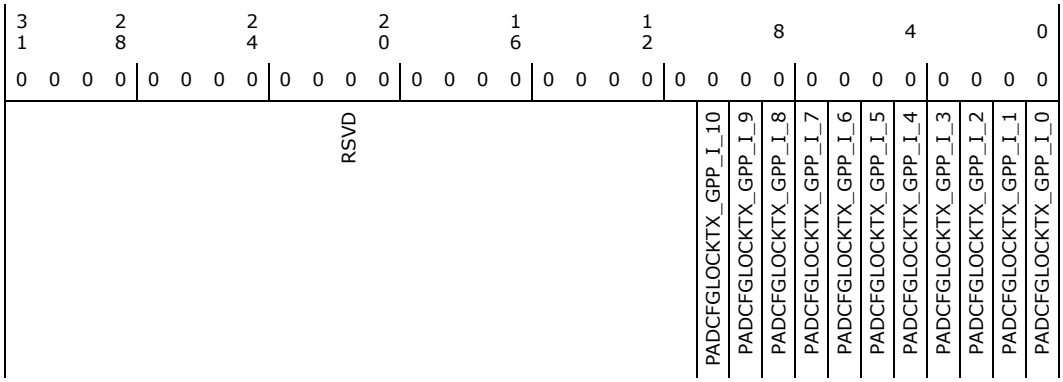
18.9.6 Pad Configuration Lock (PADCFGLOCKTX_GPP_I_0)—Offset 64h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:11	0h RO	Reserved
10	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_I_10):** Applied to GPP_I10. Same description as bit 0.
9	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_I_9):** Applied to GPP_I9. Same description as bit 0.
8	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_I_8):** Applied to GPP_I8. Same description as bit 0.
7	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_I_7):** Applied to GPP_I7. Same description as bit 0.
6	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_I_6):** Applied to GPP_I6. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
5	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_I_5):** Applied to GPP_I5. Same description as bit 0.
4	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_I_4):** Applied to GPP_I4. Same description as bit 0.
3	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_I_3):** Applied to GPP_I3. Same description as bit 0.
2	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_I_2):** Applied to GPP_I2. Same description as bit 0.
1	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_I_1):** Applied to GPP_I1. Same description as bit 0.
0	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_I_0):** PadCfgLockTx locks the GPIOTxState bit from being configured. The GPIOTxState register becomes Read-Only and software writes to the register have no effect. 0 = Unlock 1 = Locks the Pad Configuration GPIOTXState field as read-only (RO) When PadCfgLockTx is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

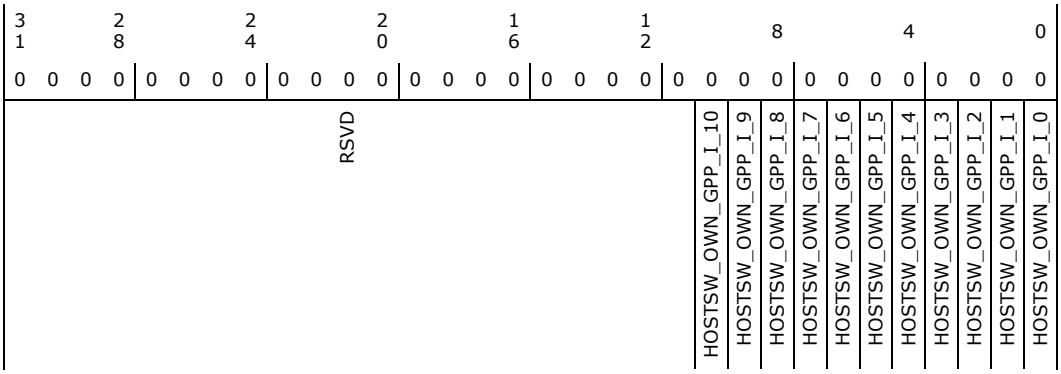
18.9.7 Host Software Pad Ownership (HOSTSW_OWN_GPP_I_0)—Offset 80h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:11	0h RO	Reserved
10	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_I_10):** Applied to GPP_I10. Same description as bit 0.
9	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_I_9):** Applied to GPP_I9. Same description as bit 0.
8	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_I_8):** Applied to GPP_I8. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
7	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_I_7):** Applied to GPP_I7. Same description as bit 0.
6	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_I_6):** Applied to GPP_I6. Same description as bit 0.
5	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_I_5):** Applied to GPP_I5. Same description as bit 0.
4	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_I_4):** Applied to GPP_I4. Same description as bit 0.
3	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_I_3):** Applied to GPP_I3. Same description as bit 0.
2	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_I_2):** Applied to GPP_I2. Same description as bit 0.
1	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_I_1):** Applied to GPP_I1. Same description as bit 0.
0	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_I_0):** This register determines the appropriate host status bit update when a pad is under host ownership. 0 = ACPI Mode. GPIO input event updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. GPI_STS update is masked. 1 = GPIO Driver Mode. GPIO input event updates are limited to GPI_STS. GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS updates are masked.

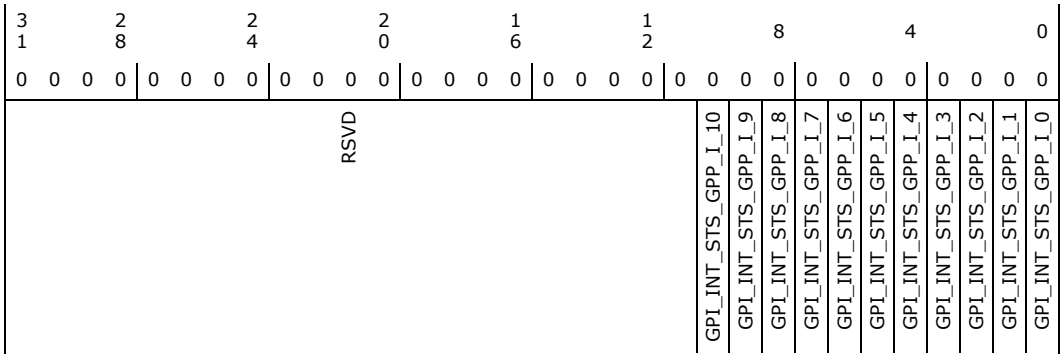
18.9.8 GPI Interrupt Status (GPI_IS_GPP_I_0)—Offset 100h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:11	0h RO	Reserved
10	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_I_10):** Applied to GPP_I10. Same description as bit 0.
9	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_I_9):** Applied to GPP_I9. Same description as bit 0.
8	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_I_8):** Applied to GPP_I8. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
7	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_I_7):** Applied to GPP_I7. Same description as bit 0.
6	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_I_6):** Applied to GPP_I6. Same description as bit 0.
5	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_I_5):** Applied to GPP_I5. Same description as bit 0.
4	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_I_4):** Applied to GPP_I4. Same description as bit 0.
3	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_I_3):** Applied to GPP_I3. Same description as bit 0.
2	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_I_2):** Applied to GPP_I2. Same description as bit 0.
1	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_I_1):** Applied to GPP_I1. Same description as bit 0.
0	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_I_0):** T 0: No interrupt 1: Interrupt asserted This bit is set to "1" by hardware when either an edge or a level event is detected on pad and all the following conditions are met: 1) The pad is used in GPIO Input mode 2) the corresponding PAD_OWN bit setting is 00b 3: The corresponding HOSTSW_OWN bit setting is 1b

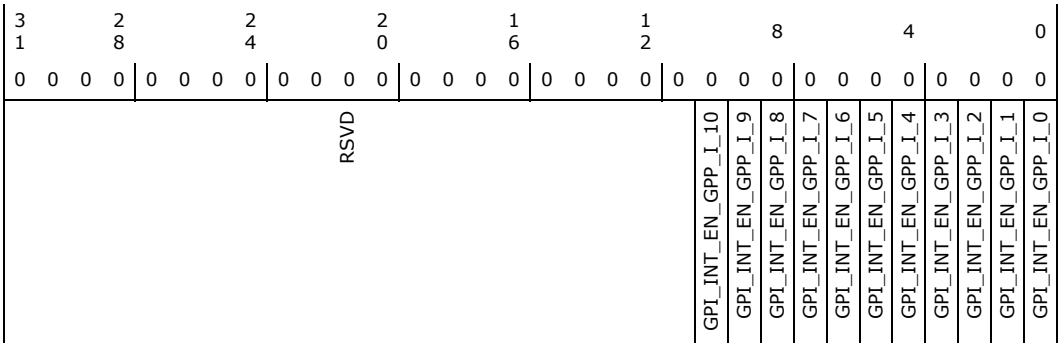
18.9.9 GPI Interrupt Enable (GPI_IE_GPP_I_0)—Offset 110h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:11	0h RO	Reserved
10	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_I_10):** Applied to GPP_I10. Same description as bit 0.
9	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_I_9):** Applied to GPP_I9. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
8	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_I_8):** Applied to GPP_I8. Same description as bit 0.
7	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_I_7):** Applied to GPP_I7. Same description as bit 0.
6	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_I_6):** Applied to GPP_I6. Same description as bit 0.
5	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_I_5):** Applied to GPP_I5. Same description as bit 0.
4	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_I_4):** Applied to GPP_I4. Same description as bit 0.
3	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_I_3):** Applied to GPP_I3. Same description as bit 0.
2	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_I_2):** Applied to GPP_I2. Same description as bit 0.
1	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_I_1):** Applied to GPP_I1. Same description as bit 0.
0	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_I_0):** This bit is used to enable/ disable the generation of APIC interrupt when the corresponding GPI_INT_STS bit is set. 0 = disable interrupt generation 1 = enable interrupt generation Refer to GPI_IRQ_ROUTE for host GPIO Driver Mode interrupt routing.

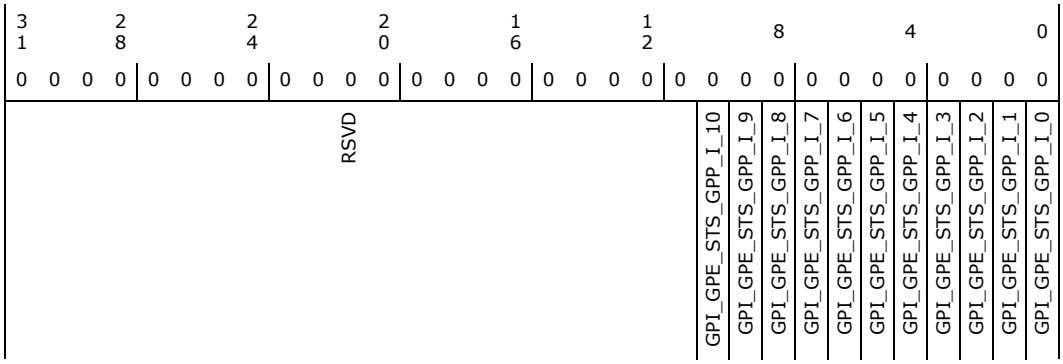
18.9.10 GPI General Purpose Events Status (GPI_GPE_STS_GPP_I_0)—Offset 120h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:11	0h RO	Reserved
10	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_I_10):** Applied to GPP_I10. Same description as bit 0.
9	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_I_9):** Applied to GPP_I9. Same description as bit 0.
8	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_I_8):** Applied to GPP_I8. Same description as bit 0.
7	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_I_7):** Applied to GPP_I7. Same description as bit 0.
6	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_I_6):** Applied to GPP_I6. Same description as bit 0.
5	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_I_5):** Applied to GPP_I5. Same description as bit 0.
4	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_I_4):** Applied to GPP_I4. Same description as bit 0.
3	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_I_3):** Applied to GPP_I3. Same description as bit 0.
2	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_I_2):** Applied to GPP_I2. Same description as bit 0.
1	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_I_1):** Applied to GPP_I1. Same description as bit 0.
0	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_I_0):** These bits are set any time the corresponding GPIO pad is set up as an input, under host ownership and the corresponding GPIO signal is high (or low if the corresponding RXINV bit is set). If the corresponding enable bit is set in the GPI_GPE_EN register, then when the GPI_GPE_STS bit is set: - If the system is in an S3-S5 state, the event will also wake the system. - If the system is in an S0 state (or upon waking back to an S0 state), an SCI will be caused, depending on the GPIRoutSCI bit for the corresponding pad. These bits are sticky bits and are cleared by writing a 1 back to this bit position. The state of GPI_GPE_EN does not prevent the setting of GPI_GPE_STS.

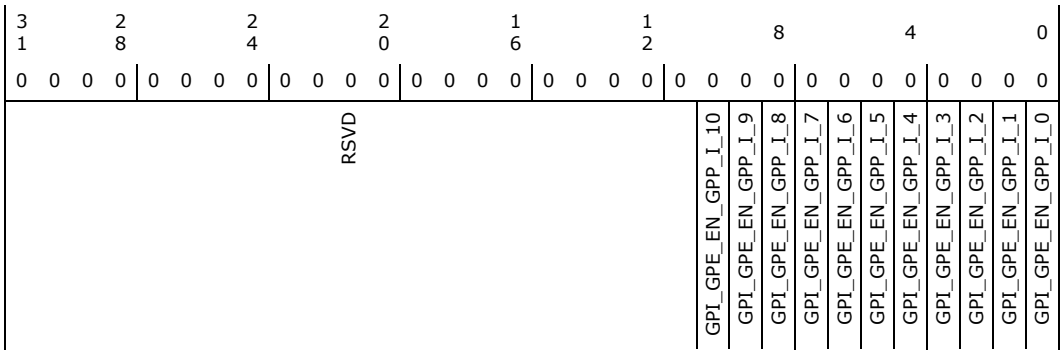
18.9.11 GPI General Purpose Events Enable (GPI_GPE_EN_GPP_I_0)—Offset 130h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:11	0h RO	Reserved
10	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_I_10):** Applied to GPP_I10. Same description as bit 0.
9	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_I_9):** Applied to GPP_I9. Same description as bit 0.
8	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_I_8):** Applied to GPP_I8. Same description as bit 0.
7	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_I_7):** Applied to GPP_I7. Same description as bit 0.
6	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_I_6):** Applied to GPP_I6. Same description as bit 0.
5	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_I_5):** Applied to GPP_I5. Same description as bit 0.
4	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_I_4):** Applied to GPP_I4. Same description as bit 0.
3	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_I_3):** Applied to GPP_I3. Same description as bit 0.
2	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_I_2):** Applied to GPP_I2. Same description as bit 0.
1	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_I_1):** Applied to GPP_I1. Same description as bit 0.
0	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_I_0):** This bit is used to enable/disable the generation of GPE to cause SCI and/or wake when the corresponding GPI_GPE_STS bit is set. 0 = disable GPE generation 1 = enable GPE generation ***Note:*** The pad must also be routed for GPE functionality in order for GPE to be generated, i.e. the corresponding GPIRoutSCI must be set to '1'.

18.9.12 SMI Status (GPI_SMI_STS_GPP_I_0)—Offset 140h

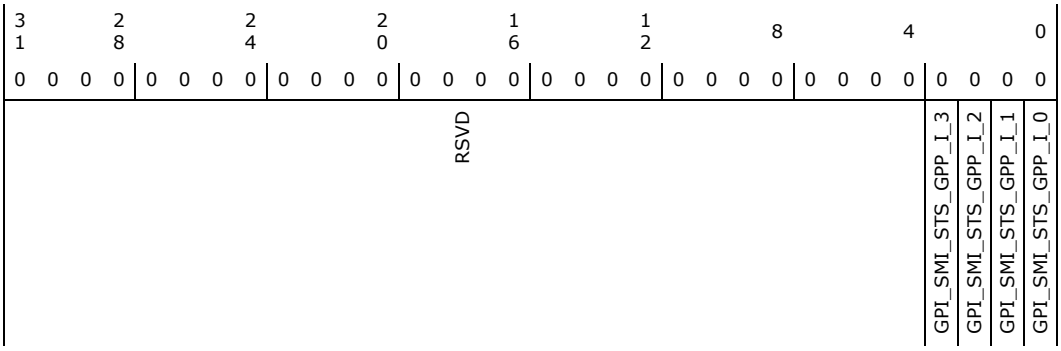
Register bits in this register are implemented for GPP_I signals that have SMI capability only. Other bits are reserved and RO.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:4	0h RO	Reserved
3	0h RW1C	**GPI SMI Status (GPI_SMI_STS_GPP_I_3):** Applied to GPP_I3. Same description as bit 0.
2	0h RW1C	**GPI SMI Status (GPI_SMI_STS_GPP_I_2):** Applied to GPP_I2. Same description as bit 0.
1	0h RW1C	**GPI SMI Status (GPI_SMI_STS_GPP_I_1):** Applied to GPP_I1. Same description as bit 0.
0	0h RW1C	**GPI SMI Status (GPI_SMI_STS_GPP_I_0):** This bit is set to 1 by hardware when a level event (See RxEdCfg,RxInv) is detected, and all the following conditions are true: - The corresponding pad is used in GPIO input mode - The corresponding PAD_OWN[2:0] is '000' (i.e. ACPI GPIO Mode). If the following conditions are true, then an SMI will be generated if the GPI_SMI_STS bit is set: 1. The corresponding bit in the GPI_SMI_EN register is set 2. The corresponding pad's GPIROUTSMI is set Writing a value of 1 will clear the bit while writing a value of 0 has no effect. 0 = There is no SMI event 1 = There is an SMI event The state of GPI_SMI_EN does not prevent the setting of GPI_SMI_STS. Defaults for these bits are dependent on the state of the GPI pads.

18.9.13 SMI Enable (GPI_SMI_EN_GPP_I_0)—Offset 150h

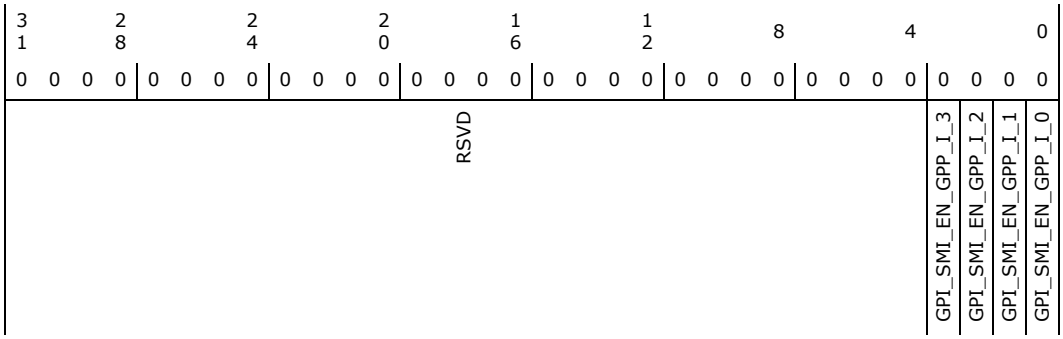
Register bits in this register are implemented for GPP_I signals that have SMI capability only. Other bits are reserved and RO.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:4	0h RO	Reserved
3	0h RW	**GPI SMI Enable (GPI_SMI_EN_GPP_I_3):** Applied to GPP_I3. Same description as bit 0.
2	0h RW	**GPI SMI Enable (GPI_SMI_EN_GPP_I_2):** Applied to GPP_I2. Same description as bit 0.
1	0h RW	**GPI SMI Enable (GPI_SMI_EN_GPP_I_1):** Applied to GPP_I1. Same description as bit 0.
0	0h RW	**GPI SMI Enable (GPI_SMI_EN_GPP_I_0):** This bit is used to enable/disable the generation of SMI when the corresponding GPI_SMI_STS bit is set. The pad must also be routed for SMI functionality in order for SMI to be generated, i.e. the corresponding GPIROUTSMI must be set to '1'. 0 = disable SMI generation 1 = enable SMI generation ***Note:*** Each of the bit is lock-able dependent on the associated Pad Configuration register PadCfgLock setting. E.g. if the PadCfgLock of pad0 is '1', bit0 of this bit is locked down to read-only.

18.9.14 NMI Status (GPI_NMI_STS_GPP_I_0)—Offset 160h

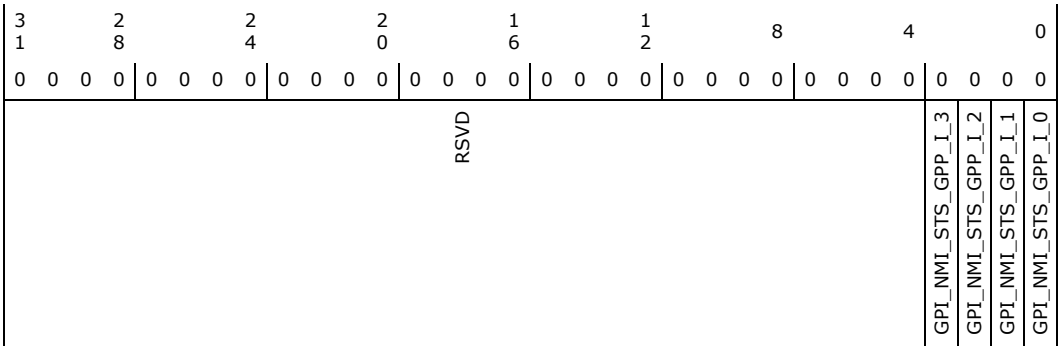
Register bits in this register are implemented for GPP_I signals that have NMI capability only. Other bits are reserved and RO.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:4	0h RO	Reserved
3	0h RW1C	**GPI NMI Status (GPI_NMI_STS_GPP_I_3):** Applied to GPP_I3. Same description as bit 0.
2	0h RW1C	**GPI NMI Status (GPI_NMI_STS_GPP_I_2):** Applied to GPP_I2. Same description as bit 0.
1	0h RW1C	**GPI NMI Status (GPI_NMI_STS_GPP_I_1):** Applied to GPP_I1. Same description as bit 0.
0	0h RW1C	**GPI NMI Status (GPI_NMI_STS_GPP_I_0):** This bit is set to 1 by hardware when an edge event is detected (See RxEdCfg, RxInv) on pad and all the following conditions are true: - The corresponding pad is used in GPIO input mode (PMode) - The corresponding GPIONMIRout is set to 1, i.e. programmed to route as NMI - The corresponding GPIOOwn[2:0] is '000' (i.e. ACPI GPIO Mode). - The corresponding GPI_NMI_EN is set Writing a value of 1 will clear the bit while writing a value of 0 has no effect. 0 = There is no NMI event 1 = There is an NMI event

18.9.15 NMI Enable (GPI_NMI_EN_GPP_I_0)—Offset 170h

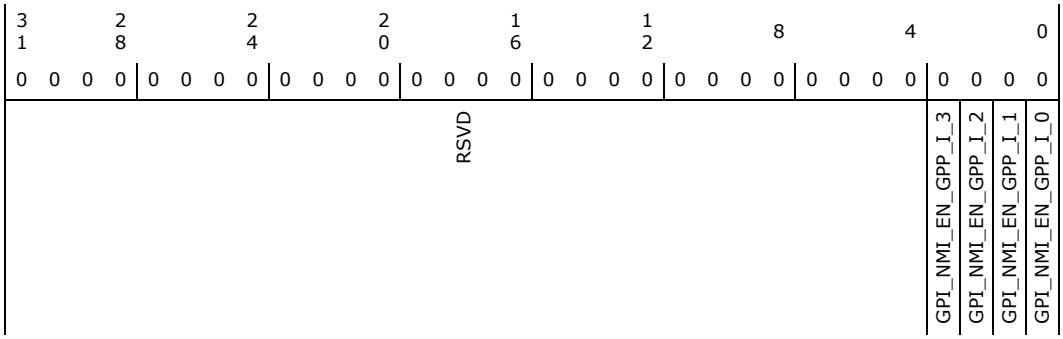
Register bits in this register are implemented for GPP_I signals that have NMI capability only. Other bits are reserved and RO.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:4	0h RO	Reserved
3	0h RW	**GPI NMI Enable (GPI_NMI_EN_GPP_I_3):** Applied to GPP_I3. Same description as bit 0.
2	0h RW	**GPI NMI Enable (GPI_NMI_EN_GPP_I_2):** Applied to GPP_I2. Same description as bit 0.
1	0h RW	**GPI NMI Enable (GPI_NMI_EN_GPP_I_1):** Applied to GPP_I1. Same description as bit 0.
0	0h RW	**GPI NMI Enable (GPI_NMI_EN_GPP_I_0):** This bit is used to enable/disable the generation of NMI when the corresponding GPI_NMI_STS bit is set and its GPIROUTNMI is set. 0 = disable NMI generation 1 = enable NMI generation Each bit is lock-able dependent on the associated Pad Configuration register PadCfgLock setting. E.g. if the PadCfgLock of pad0 is '1', bit0 of this bit is locked down to read-only.

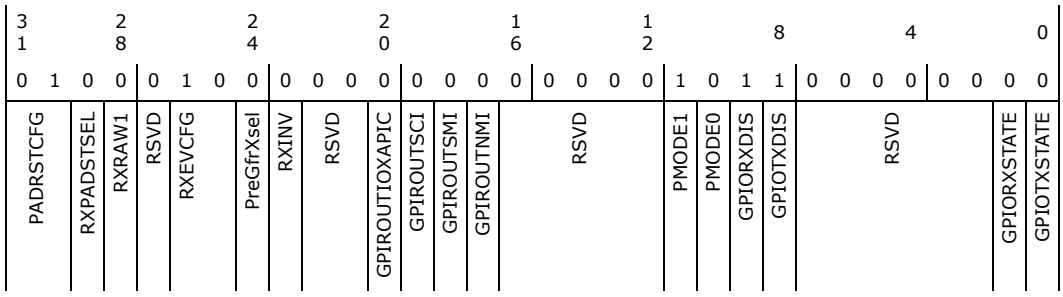
18.9.16 Pad Configuration DW0 (PAD_CFG_DW0_GPP_I_0)—Offset 400h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000B00h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	1b RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

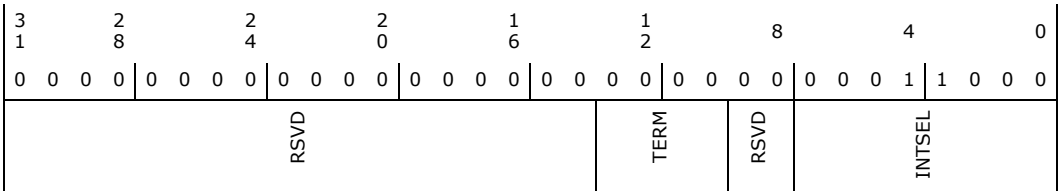
18.9.17 Pad Configuration DW1 (PAD_CFG_DW1_GPP_I_0)—Offset 404h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 18h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	18h RO	**Interrupt Select (INTSEL):** The Interrupt Select indicates which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

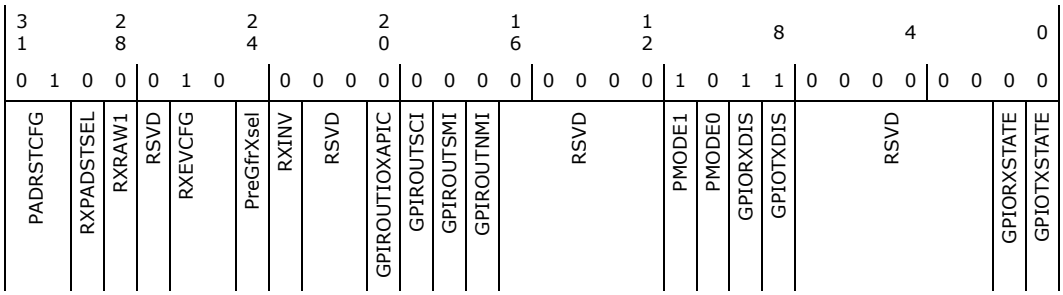
18.9.18 Pad Configuration DW0 (PAD_CFG_DW0_GPP_I_1)—Offset 408h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000B00h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	1b RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

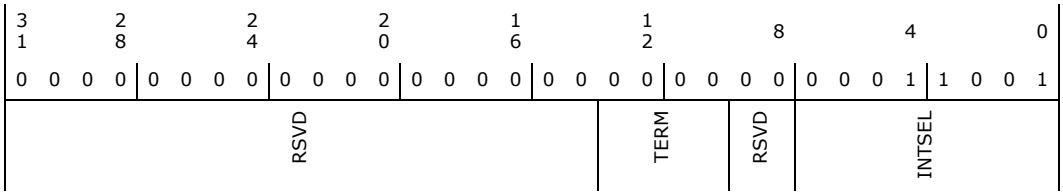
18.9.19 Pad Configuration DW1 (PAD_CFG_DW1_GPP_I_1)—Offset 40Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 19h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	19h RO	**Interrupt Select (INTSEL):** The Interrupt Select indicates which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

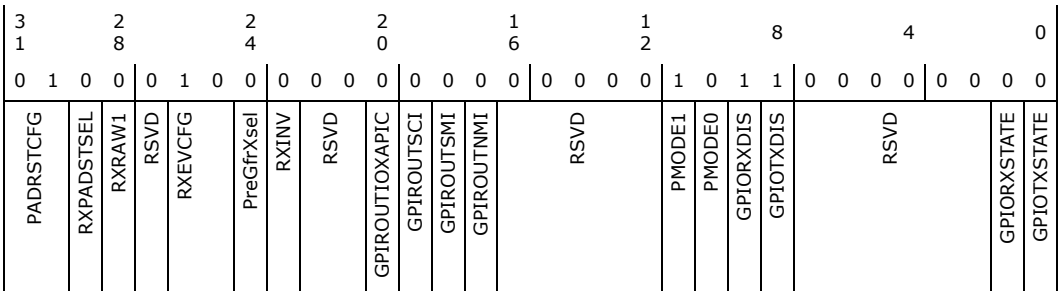
18.9.20 Pad Configuration DW0 (PAD_CFG_DW0_GPP_I_2)—Offset 410h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000B00h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	1b RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

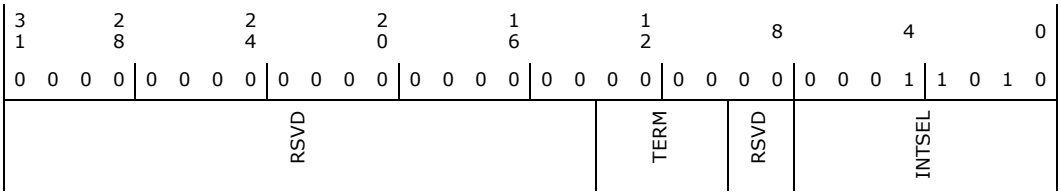
18.9.21 Pad Configuration DW1 (PAD_CFG_DW1_GPP_I_2)—Offset 414h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Ah



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Ah RO	**Interrupt Select (INTSEL):** The Interrupt Select indicates which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

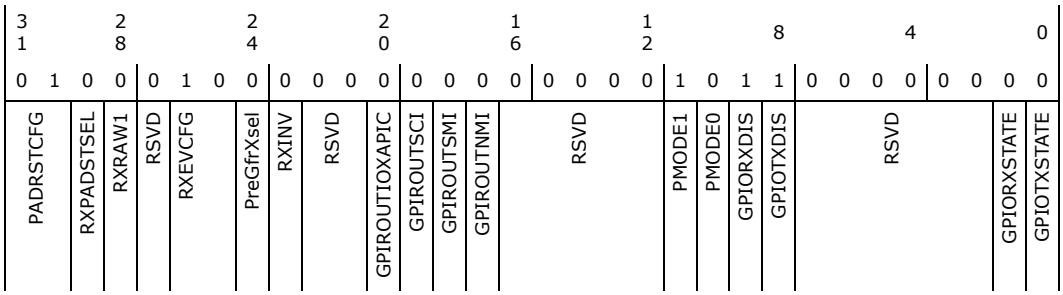
18.9.22 Pad Configuration DW0 (PAD_CFG_DW0_GPP_I_3)—Offset 418h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000B00h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	1b RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

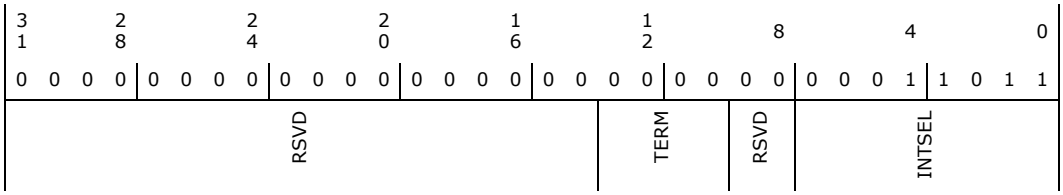
18.9.23 Pad Configuration DW1 (PAD_CFG_DW1_GPP_I_3)—Offset 41Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Bh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Bh RO	**Interrupt Select (INTSEL):** The Interrupt Select indicates which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

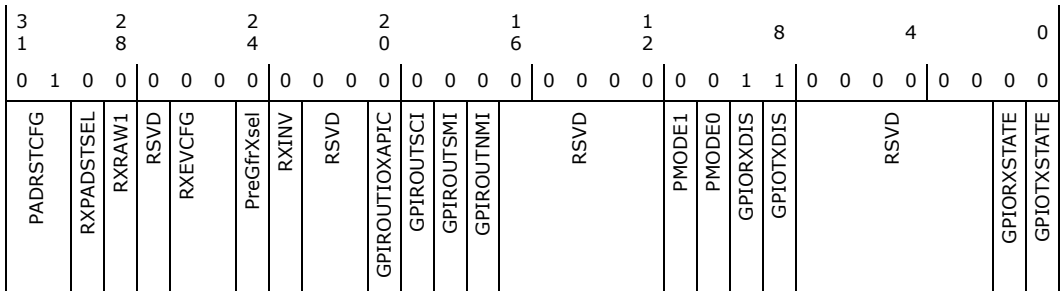
18.9.24 Pad Configuration DW0 (PAD_CFG_DW0_GPP_I_4)—Offset 420h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s) This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

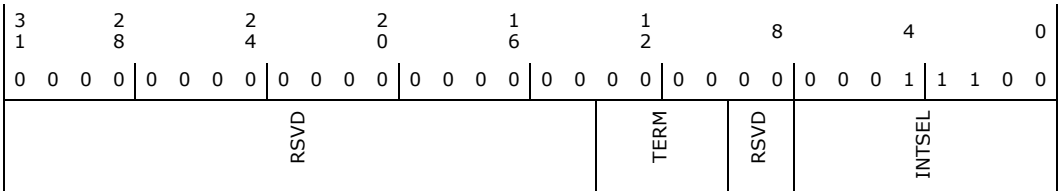
18.9.25 Pad Configuration DW1 (PAD_CFG_DW1_GPP_I_4)—Offset 424h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Ch



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Ch RO	**Interrupt Select (INTSEL):** The Interrupt Select indicates which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

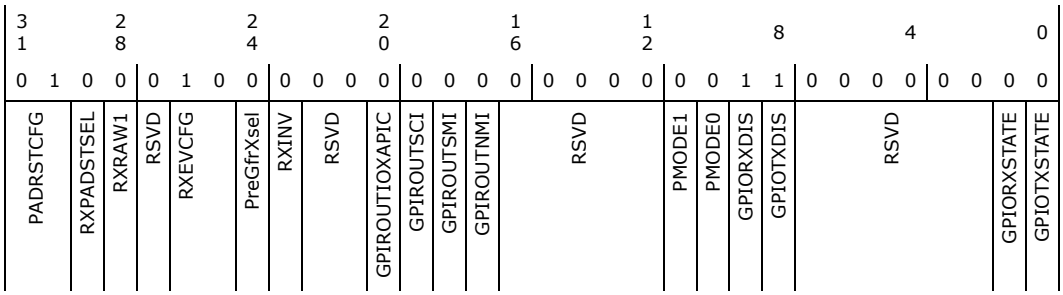
18.9.26 Pad Configuration DW0 (PAD_CFG_DW0_GPP_I_5)—Offset 428h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

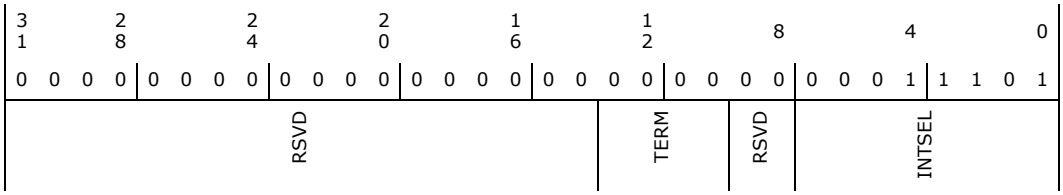
18.9.27 Pad Configuration DW1 (PAD_CFG_DW1_GPP_I_5)—Offset 42Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Dh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Dh RO	**Interrupt Select (INTSEL):** The Interrupt Select indicates which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

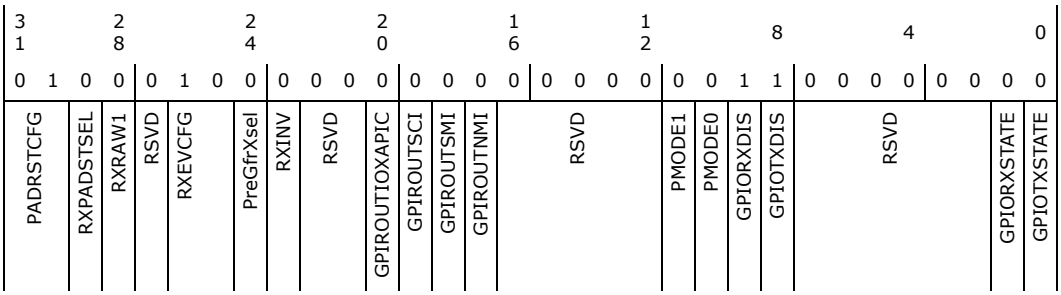
18.9.28 Pad Configuration DW0 (PAD_CFG_DW0_GPP_I_6)—Offset 430h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

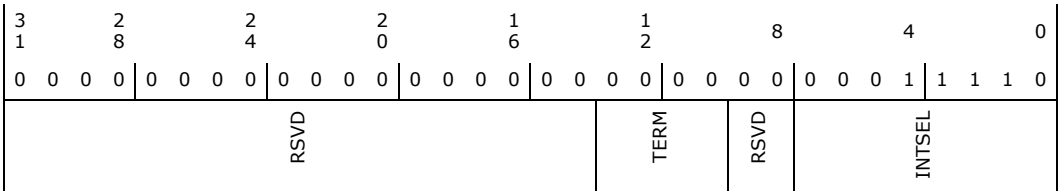
18.9.29 Pad Configuration DW1 (PAD_CFG_DW1_GPP_I_6)—Offset 434h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Eh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Eh RO	**Interrupt Select (INTSEL):** The Interrupt Select indicates which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

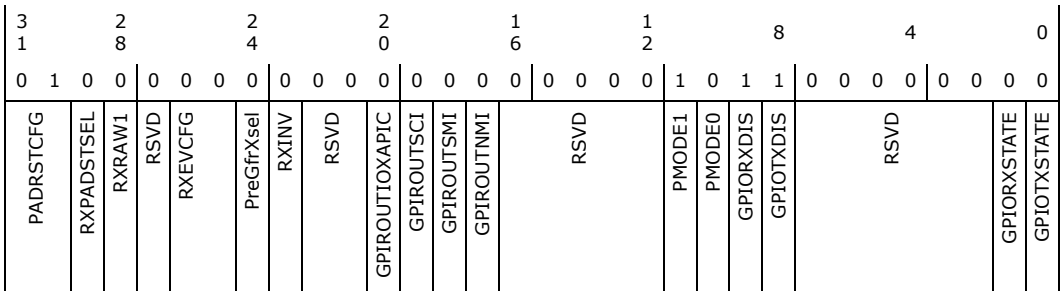
18.9.30 Pad Configuration DW0 (PAD_CFG_DW0_GPP_I_7)—Offset 438h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000B00h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	1b RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

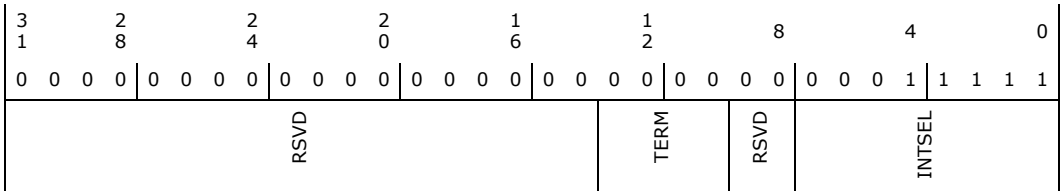
18.9.31 Pad Configuration DW1 (PAD_CFG_DW1_GPP_I_7)—Offset 43Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Fh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Fh RO	**Interrupt Select (INTSEL):** The Interrupt Select indicates which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

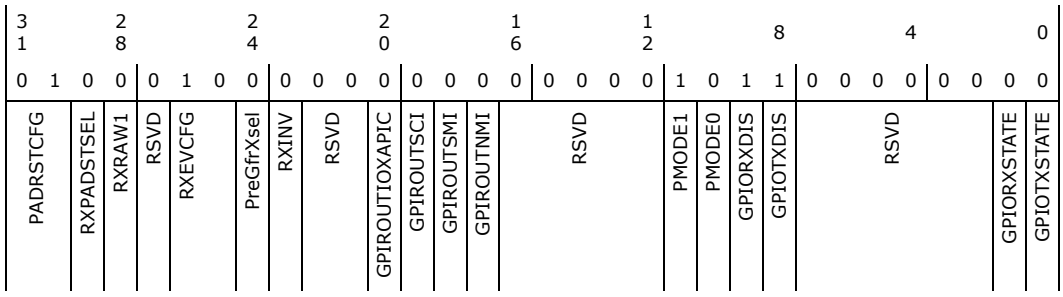
18.9.32 Pad Configuration DW0 (PAD_CFG_DW0_GPP_I_8)—Offset 440h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000B00h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s) This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	1b RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

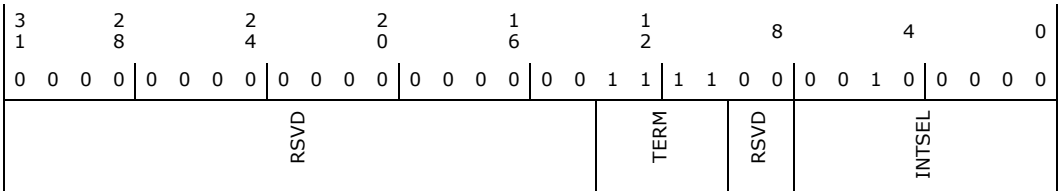
18.9.33 Pad Configuration DW1 (PAD_CFG_DW1_GPP_I_8)—Offset 444h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003C20h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	1111b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu 1111: Native controller selected by Pad Mode controls the Termination All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	20h RO	**Interrupt Select (INTSEL):** The Interrupt Select indicates which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

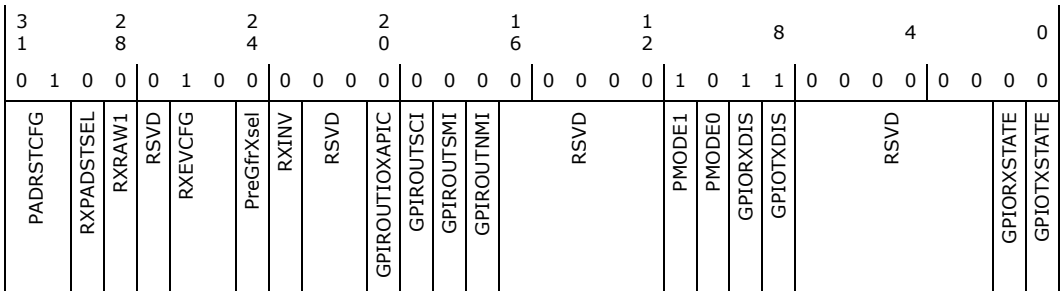
18.9.34 Pad Configuration DW0 (PAD_CFG_DW0_GPP_I_9)—Offset 448h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000B00h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	1b RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

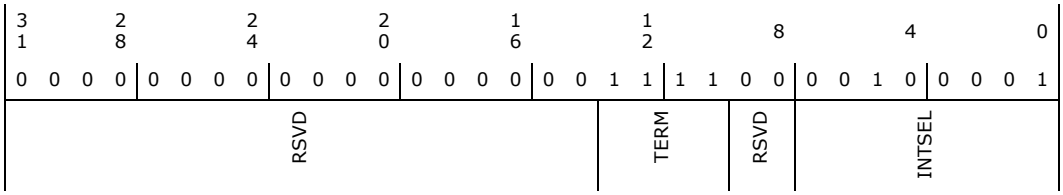
18.9.35 Pad Configuration DW1 (PAD_CFG_DW1_GPP_I_9)—Offset 44Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003C21h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	1111b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu 1111: Native controller selected by Pad Mode controls the Termination All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	21h RO	**Interrupt Select (INTSEL):** The Interrupt Select indicates which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

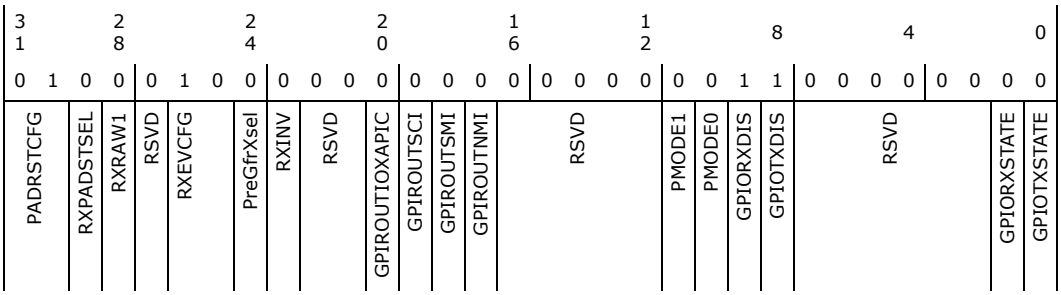
18.9.36 Pad Configuration DW0 (PAD_CFG_DW0_GPP_I_10)—Offset 450h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.

Bit Range	Default & Access	Field Name (ID): Description
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect. Default value can be found based on the Default functionality of the pad.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.

Bit Range	Default & Access	Field Name (ID): Description
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

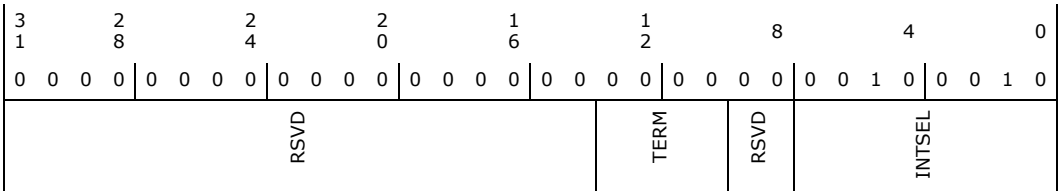
18.9.37 Pad Configuration DW1 (PAD_CFG_DW1_GPP_I_10)—Offset 454h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 22h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	22h RO	**Interrupt Select (INTSEL):** The Interrupt Select indicates which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.10 GPIO Community 4 Registers Summary

Community 0 Registers are for GPP_J and GPP_K groups.

Table 18-7. Summary of GPIO Community 4 Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
Ch	Fh	Pad Base Address (PADBAR)—Offset Ch	00000400h
10h	13h	Miscellaneous Configuration (MISCCFG)—Offset 10h	0004BA00h
20h	23h	Pad Ownership (PAD_OWN_GPP_J_0)—Offset 20h	00000000h
24h	27h	Pad Ownership (PAD_OWN_GPP_J_1)—Offset 24h	00000000h
28h	2Bh	Pad Ownership (PAD_OWN_GPP_J_2)—Offset 28h	00000000h
2C	2F	Pad Ownership (PAD_OWN_GPP_K_0)—Offset 2Ch	00000000h
30	33	Pad Ownership (PAD_OWN_GPP_K_1)—Offset 30h	00000000h
60h	63	Pad Configuration Lock (PADCFGLOCK_GPP_J)—Offset 60h	00000000h
64	67	Pad Configuration Lock (PADCFGLOCKTX_GPP_J)—Offset 64h	00000000h
68	6B	Pad Configuration Lock (PADCFGLOCK_GPP_K)—Offset 68h	00000000h
6C	6F	Pad Configuration Lock (PADCFGLOCKTX_GPP_K)—Offset 6Ch	00000000h
80h	83h	Host Software Pad Ownership (HOSTSW_OWN_GPP_J)—Offset 80h	00000000h
84h	87h	Host Software Pad Ownership (HOSTSW_OWN_GPP_K)—Offset 84h	00000000h
100h	103h	GPI Interrupt Status (GPI_IS_GPP_J)—Offset 100h	00000000h
104h	107h	GPI Interrupt Status (GPI_IS_GPP_K)—Offset 104h	00000000h
110h	113h	GPI Interrupt Enable (GPI_IE_GPP_J)—Offset 110h	00000000h
114h	117h	GPI Interrupt Enable (GPI_IE_GPP_K)—Offset 114h	00000000h
120h	123h	GPI General Purpose Events Status (GPI_GPE_STS_GPP_J)—Offset 120h	00000000h
124h	127h	GPI General Purpose Events Status (GPI_GPE_STS_GPP_K)—Offset 124h	00000000h
130h	133h	GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J)—Offset 130h	00000000h
134h	137h	GPI General Purpose Events Enable (GPI_GPE_EN_GPP_K)—Offset 134h	00000000h
400h	403h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_0)—Offset 400h	44000700h
404h	407h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_0)—Offset 404h	00000018h
408h	40Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_1)—Offset 408h	44000700h
40Ch	40Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_1)—Offset 40Ch	00000019h
410h	413h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_2)—Offset 410h	44000700h
414h	417h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_2)—Offset 414h	0000001Ah
418h	41Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_3)—Offset 418h	44000700h
41Ch	41Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_3)—Offset 41Ch	0000001Bh
420h	423h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_4)—Offset 420h	44000700h
424h	427h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_4)—Offset 424h	0000001Ch
428h	42Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_5)—Offset 428h	44000700h
42Ch	42Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_5)—Offset 42Ch	0000001Dh
430h	433h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_6)—Offset 430h	44000700h
434h	437h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_6)—Offset 434h	0000001Eh
438h	43Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_7)—Offset 438h	44000700h
43Ch	43Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_7)—Offset 43Ch	0000001Fh
440h	443h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_8)—Offset 440h	44000700h

Table 18-7. Summary of GPIO Community 4 Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
444h	447h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_8)—Offset 444h	00003C20h
448h	44Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_9)—Offset 448h	44000700h
44Ch	44Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_9)—Offset 44Ch	00003C21h
450h	453h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_10)—Offset 450h	44000700h
454h	457h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_10)—Offset 454h	00003C22h
458h	45Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_11)—Offset 458h	44000700h
45Ch	45Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_11)—Offset 45Ch	00003C23h
460h	463h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_12)—Offset 460h	44000700h
464h	467h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_12)—Offset 464h	00003C24h
468h	46Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_13)—Offset 468h	44000700h
46Ch	46Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_13)—Offset 46Ch	00003C25h
470h	473h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_14)—Offset 470h	44000700h
474h	477h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_14)—Offset 474h	00003C26h
478h	47Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_15)—Offset 478h	44000700h
47Ch	47Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_15)—Offset 47Ch	00003C27h
480h	483h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_16)—Offset 480h	44000700h
484h	487h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_16)—Offset 484h	00003C28h
488h	48Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_17)—Offset 488h	44000700h
48Ch	48Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_17)—Offset 48Ch	00003C29h
490h	493h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_18)—Offset 490h	44000700h
494h	497h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_18)—Offset 494h	00003C2Ah
498h	49Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_19)—Offset 498h	44000700h
49Ch	49Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_19)—Offset 49Ch	00003C2Bh
4A0h	4A3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_20)—Offset 4A0h	44000700h
4A4h	4A7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_20)—Offset 4A4h	00003C2Ch
4A8h	4ABh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_21)—Offset 4A8h	44000700h
4ACh	4AFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_21)—Offset 4ACh	00003C2Dh
4B0h	4B3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_22)—Offset 4B0h	44000700h
4B4h	4B7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_22)—Offset 4B4h	00003C2Eh
4B8h	4BBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_23)—Offset 4B8h	44000700h
4BCh	4BFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_23)—Offset 4BCh	00003C2Fh
4C0h	4C3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_K_0)—Offset 4C0h	44000700h
4C4h	4C7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_K_0)—Offset 4C4h	00000030h
4C8h	4CBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_K_1)—Offset 4C8h	44000700h
4CCh	4CFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_K_1)—Offset 4CCh	00000031h
4D0h	4D3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_K_2)—Offset 4D0h	44000700h
4D4h	4D7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_K_2)—Offset 4D4h	00000032h
4D8h	4DBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_K_3)—Offset 4D8h	44000700h
4DCh	4DFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_K_3)—Offset 4DCh	00000033h
4E0h	4E3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_K_4)—Offset 4E0h	44000700h
4E4h	4E7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_K_4)—Offset 4E4h	00000034h

Table 18-7. Summary of GPIO Community 4 Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
4E8h	4EBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_K_5)—Offset 4E8h	44000700h
4ECh	4EFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_K_5)—Offset 4ECh	00000035h
4F0h	4F3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_K_6)—Offset 4F0h	44000700h
4F4h	4F7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_K_6)—Offset 4F4h	00000036h
4F8h	4FBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_K_7)—Offset 4F8h	44000700h
4FCh	4FFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_K_7)—Offset 4FCh	00000037h
500h	503h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_K_8)—Offset 500h	44000700h
504h	507h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_K_8)—Offset 504h	00003C38h
508h	50Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_K_9)—Offset 508h	44000700h
50Ch	50Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_K_9)—Offset 50Ch	00000039h
510h	513h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_K_10)—Offset 510h	44000700h
514h	517h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_K_10)—Offset 514h	0000003Ah

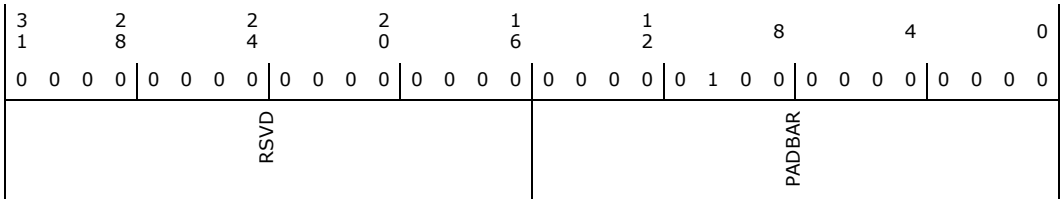
18.10.1 Pad Base Address (PADBAR)—Offset Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 400h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0h RO	Reserved
15:0	400h RO	**Pad Base Address (PADBAR):** This field provides the starting byte-align address of Pad0 register sets. It is meant for software to discover from where the very first Pad register (i.e. Pad0 register) starts to compute the next Pad address offsets.

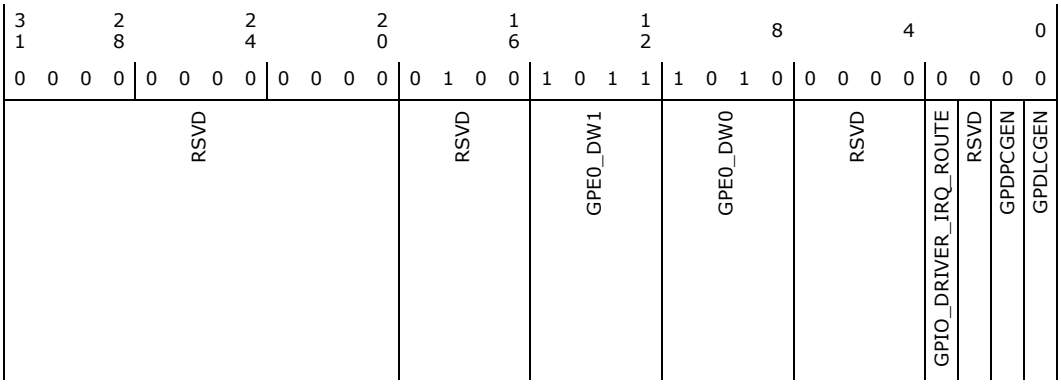
18.10.2 Miscellaneous Configuration (MISCCFG)—Offset 10h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0004BA00h



Bit Range	Default & Access	Field Name (ID): Description
31:20	0h RO	Reserved.
19:16	4h RO	Reserved
15:12	Bh RO	**GPIO Group to GPE_DW0 assignment encoding (GPE0_DW1):** This register assigns a specific GPIO Group to the ACPI GPE0[63:32]. 1h = GPP_B[23:0] mapped to GPE[55:32] and GPE[63:56] is not used 2h = GPP_C[23:0] mapped to GPE[55:32] and GPE[63:56] is not used] 3h = GPP_D[23:0] mapped to GPE[55:32] and GPE[63:56] is not used 4h = GPP_E[12:0] mapped to GPE[44:32] and GPE[63:45] is not used 5h = GPP_F[23:0] mapped to GPE[55:32] and GPE[63:56] is not used 6h = GPP_G[23:0] mapped to GPE[55:32] and GPE[63:56] is not used 7h = GPP_H[23:0] mapped to GPE[55:32] and GPE[63:56] is not used 8h = GPP_I[10:0] mapped to GPE[42:32] and GPE[63:43] is not used 9h = GPD[11:0] mapped to GPE[43:32] and GPE[63:44] is not used Ah= GPP_J[23:0] mapped to GPE[55:32] and GPE[63:56] not used Bh= GPP_K[11:0]mapped to GPE[43:32] and GPE[63:44] is not used
11:8	Ah RO	**GPIO Group to GPE_DW0 assignment encoding (GPE0_DW0):** This register assigns a specific GPIO Group to the ACPI GPE0[31:0]. 0h = GPP_A[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 1h = GPP_B[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 2h = GPP_C[23:0] mapped to GPE[23:0] and GPE[31:24] is not used] 3h = GPP_D[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 4h = GPP_E[12:0] mapped to GPE[12:0] and GPE[31:13] is not used 5h = GPP_F[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 6h = GPP_G[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 7h = GPP_H[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 8h = GPP_I[10:0] mapped to GPE[10:0] and GPE[31:11] is not used 9h = GPD[11:0] mapped to GPE[11:0] and GPE[31:12] is not used Ah= GPP_J[23:0] mapped to GPE[23:0] and GPE[31:24] is not used Bh= GPP_K[11:0]mapped to GPE[11:0] and GPE[31:12] is not used
7:4	0h RO	Reserved
3	0h RW	**GPIO Driver IRQ Route (GPIO_DRIVER_IRQ_ROUTE):** Specifies the APIC IRQ globally for all pads within the current community (GPI_IS with corresponding GPI_IE enable). 0 = IRQ14 1 = IRQ15

Bit Range	Default & Access	Field Name (ID): Description
2	0h RO	Reserved
1	0h RW	**GPIO Dynamic Partition Clock Gating Enable (GPDPCGEN):** Specifies whether the GPIO Community should take part in partition clock gating 0 = Disable participation in dynamic partition clock gating 1 = Enable participation in dynamic partition clock gating
0	0h RW	**GPIO Dynamic Local Clock Gating Enable (GPDLCGEN):** Specifies whether the GPIO Community should perform local clock gating. 0 = Disable dynamic local clock gating 1 = Enable dynamic local clock gating

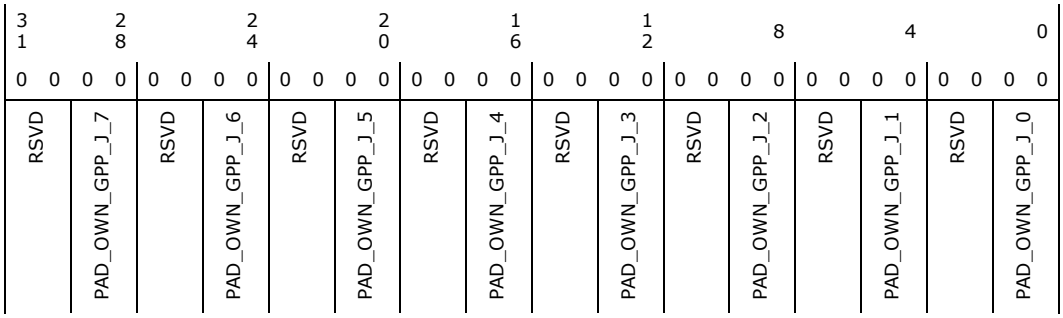
18.10.3 Pad Ownership (PAD_OWN_GPP_J_0)—Offset 20h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RW	**Pad Ownership (PAD_OWN_GPP_J_7):** Same description as bit 0, except that the bit field applies to GPP_J7.
27:26	0h RO	Reserved
25:24	0h RW	**Pad Ownership (PAD_OWN_GPP_J_6):** Same description as bit 0, except that the bit field applies to GPP_J6.
23:22	0h RO	Reserved
21:20	0h RW	**Pad Ownership (PAD_OWN_GPP_J_5):** Same description as bit 0, except that the bit field applies to GPP_J5.
19:18	0h RO	Reserved
17:16	0h RW	**Pad Ownership (PAD_OWN_GPP_J_4):** Same description as bit 0, except that the bit field applies to GPP_J4.
15:14	0h RO	Reserved
13:12	0h RW	**Pad Ownership (PAD_OWN_GPP_J_3):** Same description as bit 0, except that the bit field applies to GPP_J3.

Bit Range	Default & Access	Field Name (ID): Description
11:10	0h RO	Reserved
9:8	0h RW	**Pad Ownership (PAD_OWN_GPP_J_2):** Same description as bit 0, except that the bit field applies to GPP_J2.
7:6	0h RO	Reserved
5:4	0h RW	**Pad Ownership (PAD_OWN_GPP_J_1):** Same description as bit 0, except that the bit field applies to GPP_J1.
3:2	0h RO	Reserved
1:0	0h RW	**Pad Ownership (PAD_OWN_GPP_J_0):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = Intel ME GPIO Mode. Intel ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad

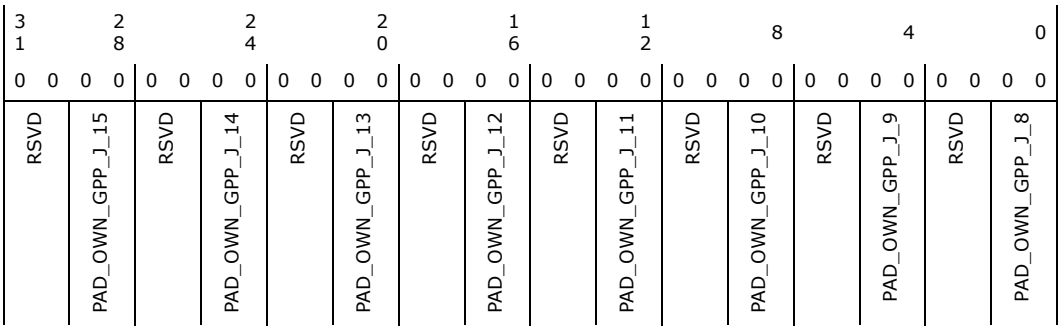
18.10.4 Pad Ownership (PAD_OWN_GPP_J_1)—Offset 24h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RW	**Pad Ownership (PAD_OWN_GPP_J_15):** Same description as bit 0, except that the bit field applies to GPP_J15.
27:26	0h RO	Reserved
25:24	0h RW	**Pad Ownership (PAD_OWN_GPP_J_14):** Same description as bit 0, except that the bit field applies to GPP_J14.
23:22	0h RO	Reserved
21:20	0h RW	**Pad Ownership (PAD_OWN_GPP_J_13):** Same description as bit 0, except that the bit field applies to GPP_J13.

Bit Range	Default & Access	Field Name (ID): Description
19:18	0h RO	Reserved
17:16	0h RW	**Pad Ownership (PAD_OWN_GPP_J_12):** Same description as bit 0, except that the bit field applies to GPP_J12.
15:14	0h RO	Reserved
13:12	0h RW	**Pad Ownership (PAD_OWN_GPP_J_11):** Same description as bit 0, except that the bit field applies to GPP_J11.
11:10	0h RO	Reserved
9:8	0h RW	**Pad Ownership (PAD_OWN_GPP_J_10):** Same description as bit 0, except that the bit field applies to GPP_J10.
7:6	0h RO	Reserved
5:4	0h RW	**Pad Ownership (PAD_OWN_GPP_J_9):** Same description as bit 0, except that the bit field applies to GPP_J9.
3:2	0h RO	Reserved
1:0	0h RW	**Pad Ownership (PAD_OWN_GPP_J_8):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = Intel ME GPIO Mode. Intel ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad

18.10.5 Pad Ownership (PAD_OWN_GPP_J_2)—Offset 28h

Same description as PAD_OWN_GPP_J_0, except that this register is for GPP_J[23:16].

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	28	24	20	16	12	8	4	0
0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0
RSVD	PAD_OWN_GPP_J_23	RSVD	PAD_OWN_GPP_J_22	RSVD	PAD_OWN_GPP_J_21	RSVD	PAD_OWN_GPP_J_20	RSVD	PAD_OWN_GPP_J_19	RSVD	PAD_OWN_GPP_J_18	RSVD	PAD_OWN_GPP_J_17	RSVD	PAD_OWN_GPP_J_16

Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RW	**Pad Ownership (PAD_OWN_GPP_J_23):** Same description as bit 0, except that the bit field applies to GPP_J23.
27:26	0h RO	Reserved
25:24	0h RW	**Pad Ownership (PAD_OWN_GPP_J_22):** Same description as bit 0, except that the bit field applies to GPP_J22.
23:22	0h RO	Reserved
21:20	0h RW	**Pad Ownership (PAD_OWN_GPP_J_21):** Same description as bit 0, except that the bit field applies to GPP_J21.
19:18	0h RO	Reserved
17:16	0h RW	**Pad Ownership (PAD_OWN_GPP_J_20):** Same description as bit 0, except that the bit field applies to GPP_J20.
15:14	0h RO	Reserved
13:12	0h RW	**Pad Ownership (PAD_OWN_GPP_J_19):** Same description as bit 0, except that the bit field applies to GPP_J19.
11:10	0h RO	Reserved
9:8	0h RW	**Pad Ownership (PAD_OWN_GPP_J_18):** Same description as bit 0, except that the bit field applies to GPP_J18.
7:6	0h RO	Reserved
5:4	0h RW	**Pad Ownership (PAD_OWN_GPP_J_17):** Same description as bit 0, except that the bit field applies to GPP_J17.
3:2	0h RO	Reserved
1:0	0h RW	**Pad Ownership (PAD_OWN_GPP_J_17):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = Intel ME GPIO Mode. Intel ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad

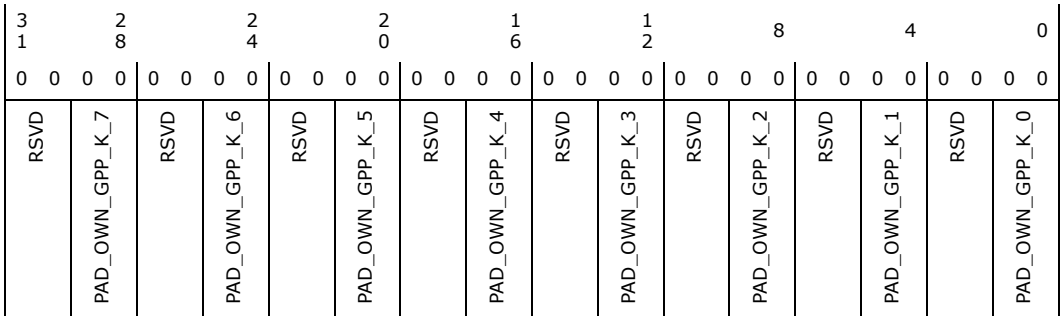
18.10.6 Pad Ownership (PAD_OWN_GPP_K_0)—Offset 2Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RW	**Pad Ownership (PAD_OWN_GPP_K_7):** Same description as bit 0, except that the bit field applies to GPP_K7.
27:26	0h RO	Reserved
25:24	0h RW	**Pad Ownership (PAD_OWN_GPP_K_6):** Same description as bit 0, except that the bit field applies to GPP_K6.
23:22	0h RO	Reserved
21:20	0h RW	**Pad Ownership (PAD_OWN_GPP_K_5):** Same description as bit 0, except that the bit field applies to GPP_K5.
19:18	0h RO	Reserved
17:16	0h RW	**Pad Ownership (PAD_OWN_GPP_K_4):** Same description as bit 0, except that the bit field applies to GPP_K4.
15:14	0h RO	Reserved
13:12	0h RW	**Pad Ownership (PAD_OWN_GPP_K_3):** Same description as bit 0, except that the bit field applies to GPP_K3.
11:10	0h RO	Reserved
9:8	0h RW	**Pad Ownership (PAD_OWN_GPP_K_2):** Same description as bit 0, except that the bit field applies to GPP_K2.
7:6	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
5:4	0h RW	**Pad Ownership (PAD_OWN_GPP_K_1):** Same description as bit 0, except that the bit field applies to GPP_K1.
3:2	0h RO	Reserved
1:0	0h RW	**Pad Ownership (PAD_OWN_GPP_K_0):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = Intel ME GPIO Mode. Intel ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad

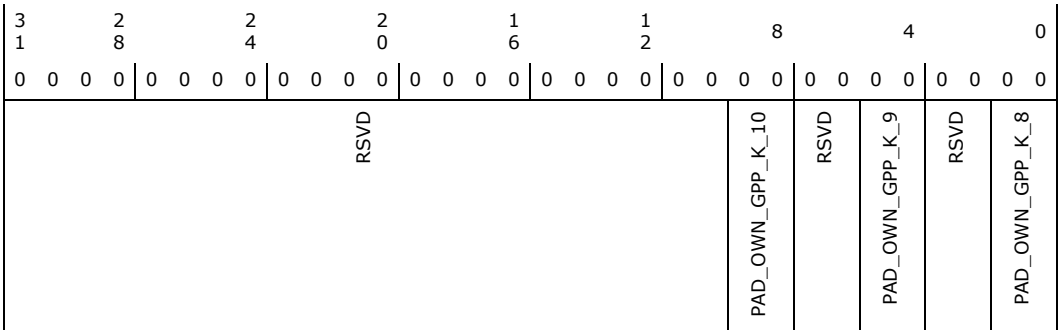
18.10.7 Pad Ownership (PAD_OWN_GPP_K_1)—Offset 30h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:10	0h RO	Reserved
9:8	0h RW	**Pad Ownership (PAD_OWN_GPP_K_10):** Same description as bit 0, except that the bit field applies to GPP_K10.
7:6	0h RO	Reserved
5:4	0h RW	**Pad Ownership (PAD_OWN_GPP_K_9):** Same description as bit 0, except that the bit field applies to GPP_K9.
3:2	0h RO	Reserved
1:0	0h RW	**Pad Ownership (PAD_OWN_GPP_K_8):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = Intel ME GPIO Mode. Intel ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad

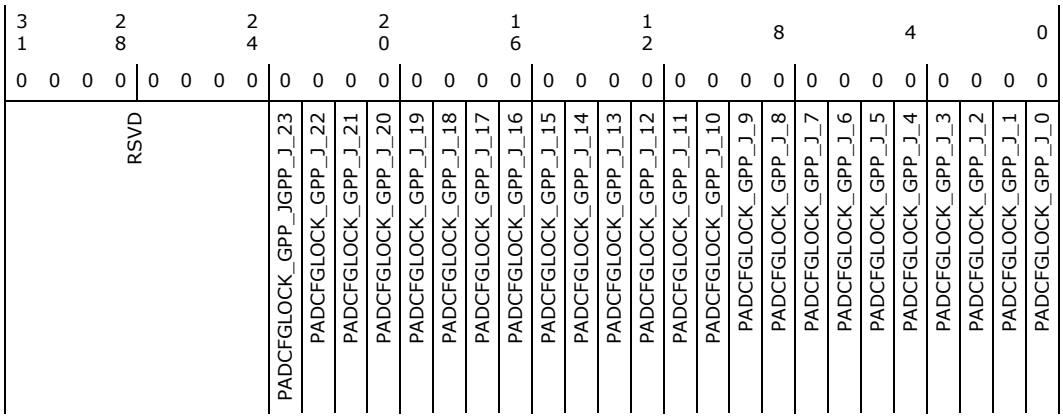
18.10.8 Pad Configuration Lock (PADCFGLOCK_GPP_J)—Offset 60h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_J_23):** Applied to GPP_J23. Same description as PADCFGLOCK_GPP_J_0.
22	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_J_22):** Applied to GPP_J22. Same description as PADCFGLOCK_GPP_J_0.
21	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_J_21):** Applied to GPP_J21. Same description as PADCFGLOCK_GPP_J_0.
20	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_J_20):** Applied to GPP_J20. Same description as PADCFGLOCK_GPP_J_0.
19	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_J_19):** Applied to GPP_J19. Same description as PADCFGLOCK_GPP_J_0.
18	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_J_18):** Applied to GPP_J18. Same description as PADCFGLOCK_GPP_J_0.
17	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_J_17):** Applied to GPP_J17. Same description as PADCFGLOCK_GPP_J_0.
16	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_J_16):** Applied to GPP_J16. Same description as PADCFGLOCK_GPP_J_0.
15	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_J_15):** Applied to GPP_J15. Same description as PADCFGLOCK_GPP_J_0.
14	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_J_14):** Applied to GPP_J14. Same description as PADCFGLOCK_GPP_J_0.
13	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_J_13):** Applied to GPP_J13. Same description as PADCFGLOCK_GPP_J_0.
12	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_J_12):** Applied to GPP_J12. Same description as PADCFGLOCK_GPP_J_0.

Bit Range	Default & Access	Field Name (ID): Description
11	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_J_11):** Applied to GPP_J11. Same description as PADCFGLOCK_GPP_J_0.
10	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_J_10):** Applied to GPP_J10. Same description as PADCFGLOCK_GPP_J_0.
9	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_J_9):** Applied to GPP_J9. Same description as PADCFGLOCK_GPP_J_0.
8	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_J_8):** Applied to GPP_J8. Same description as PADCFGLOCK_GPP_J_0.
7	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_J_7):** Applied to GPP_J7. Same description as PADCFGLOCK_GPP_J_0.
6	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_J_6):** Applied to GPP_J6. Same description as PADCFGLOCK_GPP_J_0.
5	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_J_5):** Applied to GPP_J5. Same description as PADCFGLOCK_GPP_J_0.
4	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_J_4):** Applied to GPP_J4. Same description as PADCFGLOCK_GPP_J_0.
3	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_J_3):** Applied to GPP_J3. Same description as PADCFGLOCK_GPP_J_0.
2	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_J_2):** Applied to GPP_J2. Same description as PADCFGLOCK_GPP_J_0.
1	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_J_1):** Applied to GPP_J1. Same description as PADCFGLOCK_GPP_J_0.
0	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_J_0):** Pad Configuration Lock locks specific register fields in the GPP specific registers from being configured. The registers affected become Read-Only and software writes to these registers have no effect. 0 = Unlock 1 = Lock the following register fields as read-only (RO): - Pad Configuration registers (exclude GPIOTXState) - GPI_NMI_EN Register (if implemented) - GPI_SMI_EN Register (if implemented) - GPI_GPE_EN Register (if implemented) When PadCfgLock is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

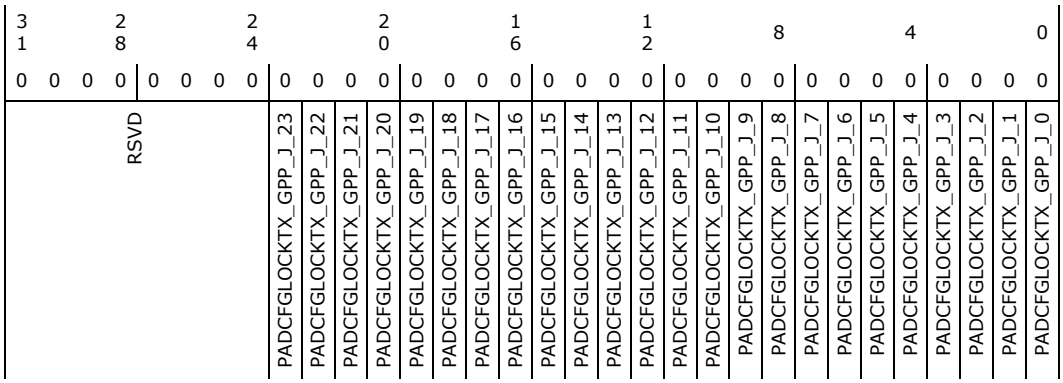
18.10.9 Pad Configuration Lock (PADCFGLOCKTX_GPP_J)—Offset 64h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_J_23):** Applied to GPP_J23. Same description as PADCFGLOCKTX_GPP_J_0.
22	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_J_22):** Applied to GPP_J22. Same description as PADCFGLOCKTX_GPP_J_0.
21	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_J_21):** Applied to GPP_J21. Same description as PADCFGLOCKTX_GPP_J_0.
20	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_J_20):** Applied to GPP_J20. Same description as PADCFGLOCKTX_GPP_J_0.
19	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_J_19):** Applied to GPP_J19. Same description as PADCFGLOCKTX_GPP_J_0.
18	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_J_18):** Applied to GPP_J18. Same description as PADCFGLOCKTX_GPP_J_0.
17	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_J_17):** Applied to GPP_J17. Same description as PADCFGLOCKTX_GPP_J_0.
16	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_J_16):** Applied to GPP_J16. Same description as PADCFGLOCKTX_GPP_J_0.
15	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_J_15):** Applied to GPP_J15. Same description as PADCFGLOCKTX_GPP_J_0.
14	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_J_14):** Applied to GPP_J14. Same description as PADCFGLOCKTX_GPP_J_0.
13	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_J_13):** Applied to GPP_J13. Same description as PADCFGLOCKTX_GPP_J_0.
12	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_J_12):** Applied to GPP_J12. Same description as PADCFGLOCKTX_GPP_J_0.
11	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_J_11):** Applied to GPP_J11. Same description as PADCFGLOCKTX_GPP_J_0.

Bit Range	Default & Access	Field Name (ID): Description
10	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_J_10):** Applied to GPP_J10. Same description as PADCFGLOCKTX_GPP_J_0.
9	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_J_9):** Applied to GPP_J9. Same description as PADCFGLOCKTX_GPP_J_0.
8	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_J_8):** Applied to GPP_J8. Same description as PADCFGLOCKTX_GPP_J_0.
7	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_J_7):** Applied to GPP_J7. Same description as PADCFGLOCKTX_GPP_J_0.
6	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_J_6):** Applied to GPP_J6. Same description as PADCFGLOCKTX_GPP_J_0.
5	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_J_5):** Applied to GPP_J5. Same description as PADCFGLOCKTX_GPP_J_0.
4	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_J_4):** Applied to GPP_J4. Same description as PADCFGLOCKTX_GPP_J_0.
3	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_J_3):** Applied to GPP_J3. Same description as PADCFGLOCKTX_GPP_J_0.
2	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_J_2):** Applied to GPP_J2. Same description as PADCFGLOCKTX_GPP_J_0.
1	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_J_1):** Applied to GPP_J1. Same description as PADCFGLOCKTX_GPP_J_0.
0	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_J_0):** PadCfgLockTx locks the GPIOTxState bit from being configured. The GPIOTxState register becomes Read-Only and software writes to the register have no effect. 0 = Unlock 1 = Locks the Pad Configuration GPIOTXState field as read-only (RO) When PadCfgLockTx is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

18.10.10 Pad Configuration Lock (PADCFGLOCK_GPP_K)—Offset 68h

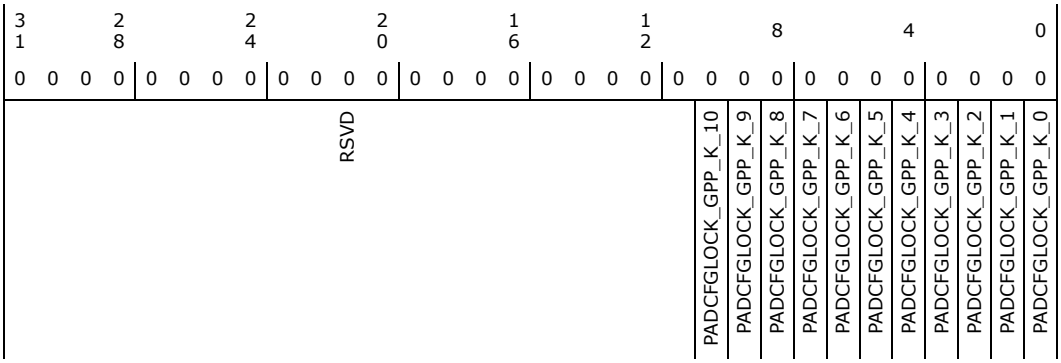
Same description as PADCFGLOCK_GPP_J register, except this register applies to GPP_K[10:0].

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:11	0h RO	Reserved
10	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_K_10):** Applied to GPP_K10. Same description as PADCFGLOCK_GPP_K_0.
9	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_K_9):** Applied to GPP_K9. Same description as PADCFGLOCK_GPP_K_0.
8	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_K_8):** Applied to GPP_K8. Same description as PADCFGLOCK_GPP_K_0.
7	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_K_7):** Applied to GPP_K7. Same description as PADCFGLOCK_GPP_K_0.
6	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_K_6):** Applied to GPP_K6. Same description as PADCFGLOCK_GPP_K_0.
5	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_K_5):** Applied to GPP_K5. Same description as PADCFGLOCK_GPP_K_0.
4	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_K_4):** Applied to GPP_K4. Same description as PADCFGLOCK_GPP_K_0.
3	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_K_3):** Applied to GPP_K3. Same description as PADCFGLOCK_GPP_K_0.
2	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_K_2):** Applied to GPP_K2. Same description as PADCFGLOCK_GPP_K_0.
1	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_K_1):** Applied to GPP_K1. Same description as PADCFGLOCK_GPP_K_0.
0	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_K_0):** Pad Configuration Lock locks specific register fields in the GPP specific registers from being configured. The registers affected become Read-Only and software writes to these registers have no effect. 0 = Unlock 1 = Lock the following register fields as read-only (RO): - Pad Configuration registers (exclude GPIOTXState) - GPI_NMI_EN Register (if implemented) - GPI_SMI_EN Register (if implemented) - GPI_GPE_EN Register (if implemented) When PadCfgLock is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

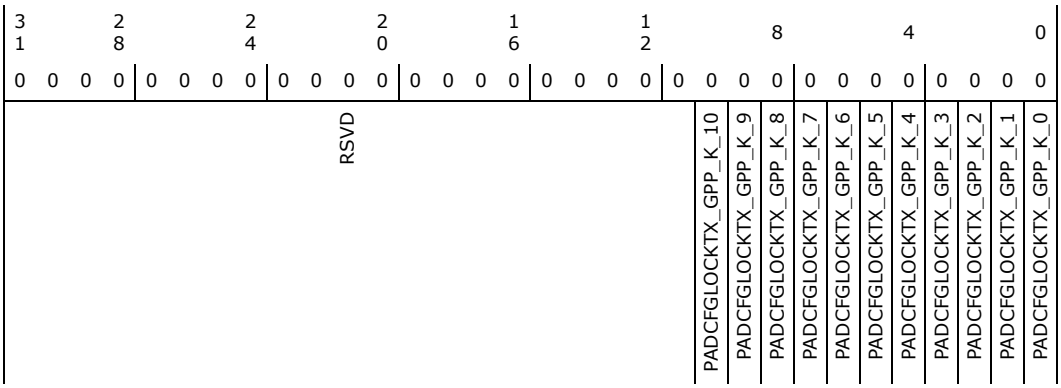
18.10.11 Pad Configuration Lock (PADCFGLOCKTX_GPP_K)—Offset 6Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:11	0h RO	Reserved
10	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_K_10):** Applied to GPP_K10. Same description as PADCFGLOCKTX_GPP_K_0.
9	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_K_9):** Applied to GPP_K9. Same description as PADCFGLOCKTX_GPP_K_0.
8	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_K_8):** Applied to GPP_K8. Same description as PADCFGLOCKTX_GPP_K_0.
7	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_K_7):** Applied to GPP_K7. Same description as PADCFGLOCKTX_GPP_K_0.
6	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_K_6):** Applied to GPP_K6. Same description as PADCFGLOCKTX_GPP_K_0.
5	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_K_5):** Applied to GPP_K5. Same description as PADCFGLOCKTX_GPP_K_0.
4	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_K_4):** Applied to GPP_K4. Same description as PADCFGLOCKTX_GPP_K_0.
3	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_K_3):** Applied to GPP_K3. Same description as PADCFGLOCKTX_GPP_K_0.
2	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_K_2):** Applied to GPP_K2. Same description as PADCFGLOCKTX_GPP_K_0.
1	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_K_1):** Applied to GPP_K1. Same description as PADCFGLOCKTX_GPP_K_0.
0	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_K_0):** PadCfgLockTx locks the GPIOTxState bit from being configured. The GPIOTxState register becomes Read-Only and software writes to the register have no effect. 0 = Unlock 1 = Locks the Pad Configuration GPIOTXState field as read-only (RO) When PadCfgLockTx is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

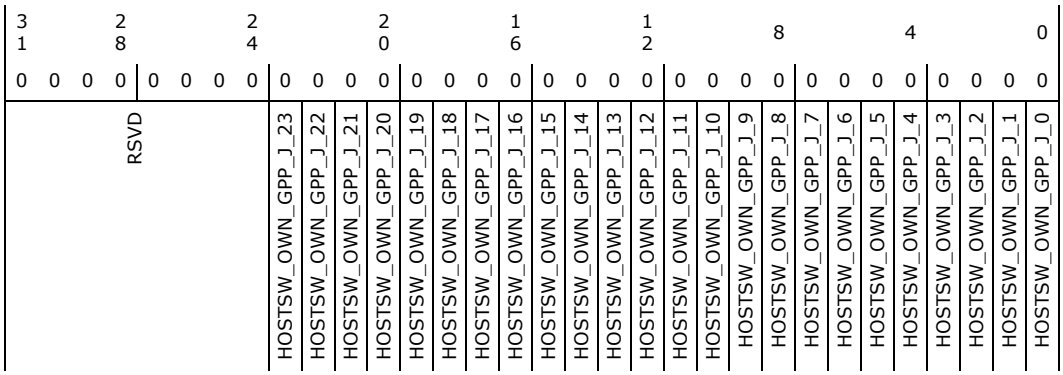
18.10.12 Host Software Pad Ownership (HOSTSW_OWN_GPP_J)—Offset 80h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_J_23):** Applied to GPP_J23. Same description as bit 0.
22	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_J_22):** Applied to GPP_J22. Same description as bit 0.
21	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_J_21):** Applied to GPP_J21. Same description as bit 0.
20	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_J_20):** Applied to GPP_J20. Same description as bit 0.
19	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_J_19):** Applied to GPP_J19. Same description as bit 0.
18	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_J_18):** Applied to GPP_J18. Same description as bit 0.
17	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_J_17):** Applied to GPP_J17. Same description as bit 0.
16	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_J_16):** Applied to GPP_J16. Same description as bit 0.
15	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_J_15):** Applied to GPP_J15. Same description as bit 0.
14	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_J_14):** Applied to GPP_J14. Same description as bit 0.
13	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_J_13):** Applied to GPP_J13. Same description as bit 0.
12	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_J_12):** Applied to GPP_J12. Same description as bit 0.
11	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_J_11):** Applied to GPP_J11. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
10	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_J_10):** Applied to GPP_J10. Same description as bit 0.
9	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_J_9):** Applied to GPP_J9. Same description as bit 0.
8	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_J_8):** Applied to GPP_J8. Same description as bit 0.
7	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_J_7):** Applied to GPP_J7. Same description as bit 0.
6	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_J_6):** Applied to GPP_J6. Same description as bit 0.
5	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_J_5):** Applied to GPP_J5. Same description as bit 0.
4	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_J_4):** Applied to GPP_J4. Same description as bit 0.
3	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_J_3):** Applied to GPP_J3. Same description as bit 0.
2	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_J_2):** Applied to GPP_J2. Same description as bit 0.
1	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_J_1):** Applied to GPP_J1. Same description as bit 0.
0	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_J_0):** This register determines the appropriate host status bit update when a pad is under host ownership. 0 = ACPI Mode. GPIO input event updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. GPI_STS update is masked. 1 = GPIO Driver Mode. GPIO input event updates are limited to GPI_STS. GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS updates are masked.

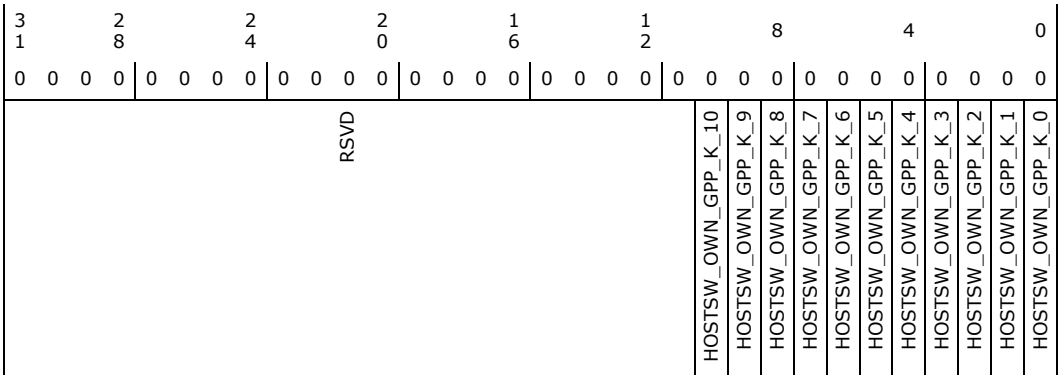
18.10.13 Host Software Pad Ownership (HOSTSW_OWN_GPP_K)—Offset 84h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:11	0h RO	Reserved
10	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_K_10):** Applied to GPP_K10. Same description as bit 0.
9	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_K_9):** Applied to GPP_K9. Same description as bit 0.
8	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_K_8):** Applied to GPP_K8. Same description as bit 0.
7	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_K_7):** Applied to GPP_K7. Same description as bit 0.
6	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_K_6):** Applied to GPP_K6. Same description as bit 0.
5	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_K_5):** Applied to GPP_K5. Same description as bit 0.
4	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_K_4):** Applied to GPP_K4. Same description as bit 0.
3	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_K_3):** Applied to GPP_K3. Same description as bit 0.
2	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_K_2):** Applied to GPP_K2. Same description as bit 0.
1	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_K_1):** Applied to GPP_K1. Same description as bit 0.
0	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_K_0):** This register determines the appropriate host status bit update when a pad is under host ownership. 0 = ACPI Mode. GPIO input event updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. GPI_STS update is masked. 1 = GPIO Driver Mode. GPIO input event updates are limited to GPI_STS. GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS updates are masked.

18.10.14 GPI Interrupt Status (GPI_IS_GPP_J)—Offset 100h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	28	24	20	16	12	8	4	0
0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0
RSVD		GPI_INT_STS_GPP_J_23	GPI_INT_STS_GPP_J_22	GPI_INT_STS_GPP_J_21	GPI_INT_STS_GPP_J_20	GPI_INT_STS_GPP_J_19	GPI_INT_STS_GPP_J_18	GPI_INT_STS_GPP_J_17	GPI_INT_STS_GPP_J_16	GPI_INT_STS_GPP_J_15	GPI_INT_STS_GPP_J_14	GPI_INT_STS_GPP_J_13	GPI_INT_STS_GPP_J_12	GPI_INT_STS_GPP_J_11	GPI_INT_STS_GPP_J_10	GPI_INT_STS_GPP_J_9	GPI_INT_STS_GPP_J_8	GPI_INT_STS_GPP_J_7	GPI_INT_STS_GPP_J_6	GPI_INT_STS_GPP_J_5	GPI_INT_STS_GPP_J_4	GPI_INT_STS_GPP_J_3	GPI_INT_STS_GPP_J_2	GPI_INT_STS_GPP_J_1	GPI_INT_STS_GPP_J_0

Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_J_23):** Applied to GPP_J23. Same description as bit 0.
22	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_J_22):** Applied to GPP_J22. Same description as bit 0.
21	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_J_21):** Applied to GPP_J21. Same description as bit 0.
20	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_J_20):** Applied to GPP_J20. Same description as bit 0.
19	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_J_19):** Applied to GPP_J19. Same description as bit 0.
18	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_J_18):** Applied to GPP_J18. Same description as bit 0.
17	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_J_17):** Applied to GPP_J17. Same description as bit 0.
16	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_J_16):** Applied to GPP_J16. Same description as bit 0.
15	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_J_15):** Applied to GPP_J15. Same description as bit 0.
14	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_J_14):** Applied to GPP_J14. Same description as bit 0.
13	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_J_13):** Applied to GPP_J13. Same description as bit 0.
12	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_J_12):** Applied to GPP_J12. Same description as bit 0.
11	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_J_11):** Applied to GPP_J11. Same description as bit 0.
10	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_J_10):** Applied to GPP_J10. Same description as bit 0.
9	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_J_9):** Applied to GPP_J9. Same description as bit 0.
8	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_J_8):** Applied to GPP_J8. Same description as bit 0.
7	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_J_7):** Applied to GPP_J7. Same description as bit 0.
6	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_J_6):** Applied to GPP_J6. Same description as bit 0.
5	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_J_5):** Applied to GPP_J5. Same description as bit 0.
4	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_J_4):** Applied to GPP_J4. Same description as bit 0.
3	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_J_3):** Applied to GPP_J3. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
2	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_J_2):** Applied to GPP_J2. Same description as bit 0.
1	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_J_1):** Applied to GPP_J1. Same description as bit 0.
0	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_J_0):** GPI Interrupt Status (GPI_INT_STS) This bit is set to '1' by hardware when either an edge or a level event is detected (See RxEdCfg RxInv) on pad and all the following conditions are true: - The corresponding pad is used in GPIO input mode - HOSTSW_OWN = 1 (i.e. Host GPIO Driver Mode). Writing a value of 1 will clear the bit while writing a value of 0 has no effect. 0 = No interrupt 1 = Interrupt asserts The state of GPI_INT_EN[x] does not prevent the setting of GPI_INT_STS[x].

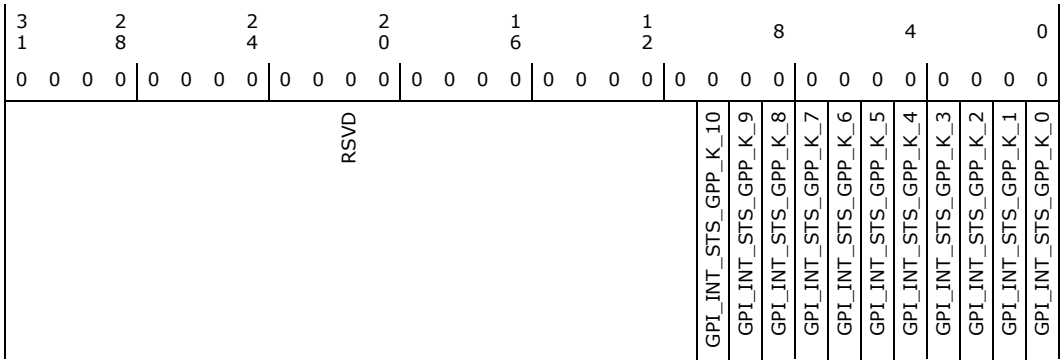
18.10.15 GPI Interrupt Status (GPI_IS_GPP_K)—Offset 104h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:11	0h RO	Reserved
10	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_K_10):** Applied to GPP_K10. Same description as bit 0.
9	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_K_9):** Applied to GPP_K9. Same description as bit 0.
8	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_K_8):** Applied to GPP_K8. Same description as bit 0.
7	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_K_7):** Applied to GPP_K7. Same description as bit 0.
6	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_K_6):** Applied to GPP_K6. Same description as bit 0.
5	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_K_5):** Applied to GPP_K5. Same description as bit 0.
4	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_K_4):** Applied to GPP_K4. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
3	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_K_3):** Applied to GPP_K3. Same description as bit 0.
2	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_K_2):** Applied to GPP_K2. Same description as bit 0.
1	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_K_1):** Applied to GPP_K1. Same description as bit 0.
0	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_K_0):** GPI Interrupt Status (GPI_INT_STS) This bit is set to '1' by hardware when either an edge or a level event is detected (See RxEdCfg RxInv) on pad and all the following conditions are true: - The corresponding pad is used in GPIO input mode - HOSTSW_OWN = 1 (i.e. Host GPIO Driver Mode). Writing a value of 1 will clear the bit while writing a value of 0 has no effect. 0 = No interrupt 1 = Interrupt asserts The state of GPI_INT_EN[x] does not prevent the setting of GPI_INT_STS[x].

18.10.16 GPI Interrupt Enable (GPI_IE_GPP_J)—Offset 110h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD								GPI_INT_EN_GPP_J_23	GPI_INT_EN_GPP_J_22	GPI_INT_EN_GPP_J_21	GPI_INT_EN_GPP_J_20	GPI_INT_EN_GPP_J_19	GPI_INT_EN_GPP_J_18	GPI_INT_EN_GPP_J_17	GPI_INT_EN_GPP_J_16	GPI_INT_EN_GPP_J_15	GPI_INT_EN_GPP_J_14	GPI_INT_EN_GPP_J_13	GPI_INT_EN_GPP_J_12	GPI_INT_EN_GPP_J_11	GPI_INT_EN_GPP_J_10	GPI_INT_EN_GPP_J_9	GPI_INT_EN_GPP_J_8	GPI_INT_EN_GPP_J_7	GPI_INT_EN_GPP_J_6	GPI_INT_EN_GPP_J_5	GPI_INT_EN_GPP_J_4	GPI_INT_EN_GPP_J_3	GPI_INT_EN_GPP_J_2	GPI_INT_EN_GPP_J_1	GPI_INT_EN_GPP_J_0

Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_J_23):** Applied to GPP_J23. Same description as bit 0.
22	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_J_22):** Applied to GPP_J22. Same description as bit 0.
21	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_J_21):** Applied to GPP_J21. Same description as bit 0.
20	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_J_20):** Applied to GPP_J20. Same description as bit 0.
19	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_J_19):** Applied to GPP_J19. Same description as bit 0.
18	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_J_18):** Applied to GPP_J18. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_J_17):** Applied to GPP_J17. Same description as bit 0.
16	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_J_16):** Applied to GPP_J16. Same description as bit 0.
15	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_J_15):** Applied to GPP_J15. Same description as bit 0.
14	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_J_14):** Applied to GPP_J14. Same description as bit 0.
13	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_J_13):** Applied to GPP_J13. Same description as bit 0.
12	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_J_12):** Applied to GPP_J12. Same description as bit 0.
11	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_J_11):** Applied to GPP_J11. Same description as bit 0.
10	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_J_10):** Applied to GPP_J10. Same description as bit 0.
9	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_J_9):** Applied to GPP_J9. Same description as bit 0.
8	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_J_8):** Applied to GPP_J8. Same description as bit 0.
7	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_J_7):** Applied to GPP_J7. Same description as bit 0.
6	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_J_6):** Applied to GPP_J6. Same description as bit 0.
5	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_J_5):** Applied to GPP_J5. Same description as bit 0.
4	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_J_4):** Applied to GPP_J3. Same description as bit 0.
3	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_J_3):** Applied to GPP_J3. Same description as bit 0.
2	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_J_2):** Applied to GPP_J2. Same description as bit 0.
1	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_J_1):** Applied to GPP_J1. Same description as bit 0.
0	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_J_0):** This bit is used to enable/ disable the generation of APIC interrupt when the corresponding GPI_INT_STS bit is set. 0 = disable interrupt generation 1 = enable interrupt generation

18.10.17 GPI Interrupt Enable (GPI_IE_GPP_K)—Offset 114h

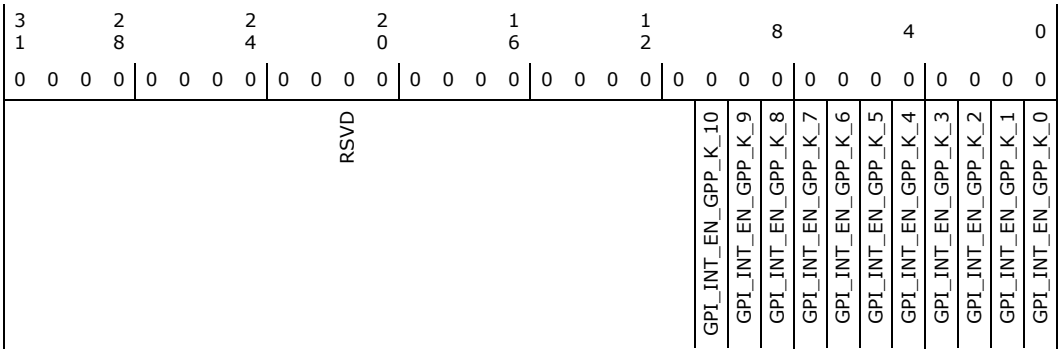
Same description as GPI_IE_GPP_J register, except that this register is for GPP_K[10:0].

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
10	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_K_10):** Applied to GPP_K10. Same description as bit 0.
9	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_K_9):** Applied to GPP_K9. Same description as bit 0.
8	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_K_8):** Applied to GPP_K8. Same description as bit 0.
7	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_K_7):** Applied to GPP_K7. Same description as bit 0.
6	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_K_6):** Applied to GPP_K6. Same description as bit 0.
5	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_K_5):** Applied to GPP_K5. Same description as bit 0.
4	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_K_4):** Applied to GPP_K4. Same description as bit 0.
3	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_K_3):** Applied to GPP_K3. Same description as bit 0.
2	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_K_2):** Applied to GPP_K2. Same description as bit 0.
1	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_K_1):** Applied to GPP_K1. Same description as bit 0.
0	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_K_0):** This bit is used to enable/ disable the generation of APIC interrupt when the corresponding GPI_INT_STS bit is set. 0 = disable interrupt generation 1 = enable interrupt generation

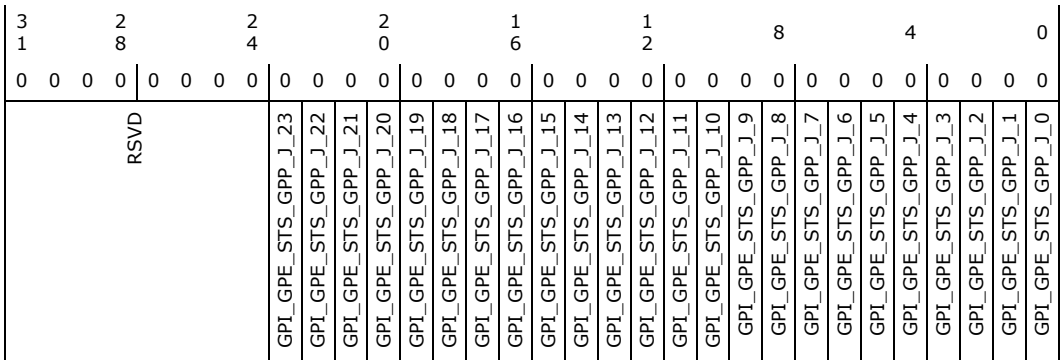
18.10.18 GPI General Purpose Events Status (GPI_GPE_STS_GPP_J)—Offset 120h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_J_23):** Applied to GPP_J23. Same description as bit 0.
22	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_J_22):** Applied to GPP_J22. Same description as bit 0.
21	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_J_21):** Applied to GPP_J21. Same description as bit 0.
20	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_J_20):** Applied to GPP_J20. Same description as bit 0.
19	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_J_19):** Applied to GPP_J19. Same description as bit 0.
18	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_J_18):** Applied to GPP_J18. Same description as bit 0.
17	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_J_17):** Applied to GPP_J17. Same description as bit 0.
16	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_J_16):** Applied to GPP_J16. Same description as bit 0.
15	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_J_15):** Applied to GPP_J15. Same description as bit 0.
14	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_J_14):** Applied to GPP_J14. Same description as bit 0.
13	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_J_13):** Applied to GPP_J13. Same description as bit 0.
12	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_J_12):** Applied to GPP_J12. Same description as bit 0.
11	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_J_11):** Applied to GPP_J11. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
10	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_J_10):** Applied to GPP_J10. Same description as bit 0.
9	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_J_9):** Applied to GPP_J9. Same description as bit 0.
8	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_J_8):** Applied to GPP_J8. Same description as bit 0.
7	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_J_7):** Applied to GPP_J7. Same description as bit 0.
6	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_J_6):** Applied to GPP_J6. Same description as bit 0.
5	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_J_5):** Applied to GPP_J5. Same description as bit 0.
4	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_J_4):** Applied to GPP_J4. Same description as bit 0.
3	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_J_3):** Applied to GPP_J3. Same description as bit 0.
2	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_J_2):** Applied to GPP_J2. Same description as bit 0.
1	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_J_1):** Applied to GPP_J1. Same description as bit 0.
0	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_J_0):** These bits are set any time the corresponding GPIO pad is set up as an input, under host ownership and the corresponding GPIO signal is high (or low if the corresponding RXINV bit is set). If the corresponding enable bit is set in the GPI_GPE_EN register, then when the GPI_GPE_STS bit is set: - If the system is in an S3-S5 state, the event will also wake the system. - If the system is in an S0 state (or upon waking back to an S0 state), an SCI will be caused, depending on the GPIRoutSCI bit for the corresponding pad. These bits are sticky bits and are cleared by writing a 1 back to this bit position. The state of GPI_GPE_EN does not prevent the setting of GPI_GPE_STS.

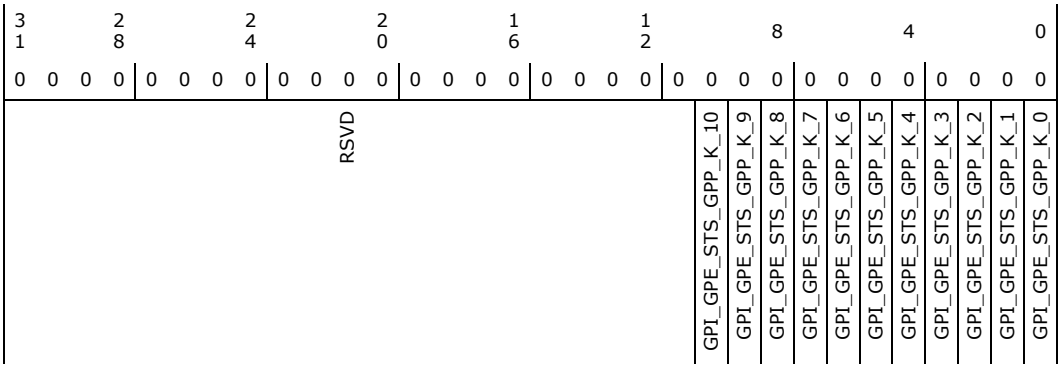
18.10.19 GPI General Purpose Events Status (GPI_GPE_STS_GPP_K)—Offset 124h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
10	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_K_10):** Applied to GPP_K10. Same description as bit 0.
9	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_K_9):** Applied to GPP_K9. Same description as bit 0.
8	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_K_8):** Applied to GPP_K8. Same description as bit 0.
7	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_K_7):** Applied to GPP_K7. Same description as bit 0.
6	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_K_6):** Applied to GPP_K6. Same description as bit 0.
5	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_K_5):** Applied to GPP_K5. Same description as bit 0.
4	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_K_4):** Applied to GPP_K4. Same description as bit 0.
3	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_K_3):** Applied to GPP_K3. Same description as bit 0.
2	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_K_2):** Applied to GPP_K2. Same description as bit 0.
1	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_K_1):** Applied to GPP_K1. Same description as bit 0.
0	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_K_0):** These bits are set any time the corresponding GPIO pad is set up as an input, under host ownership and the corresponding GPIO signal is high (or low if the corresponding RXINV bit is set). If the corresponding enable bit is set in the GPI_GPE_EN register, then when the GPI_GPE_STS bit is set: - If the system is in an S3-S5 state, the event will also wake the system. - If the system is in an S0 state (or upon waking back to an S0 state), an SCI will be caused, depending on the GPIRoutSCI bit for the corresponding pad. These bits are sticky bits and are cleared by writing a 1 back to this bit position. The state of GPI_GPE_EN does not prevent the setting of GPI_GPE_STS.

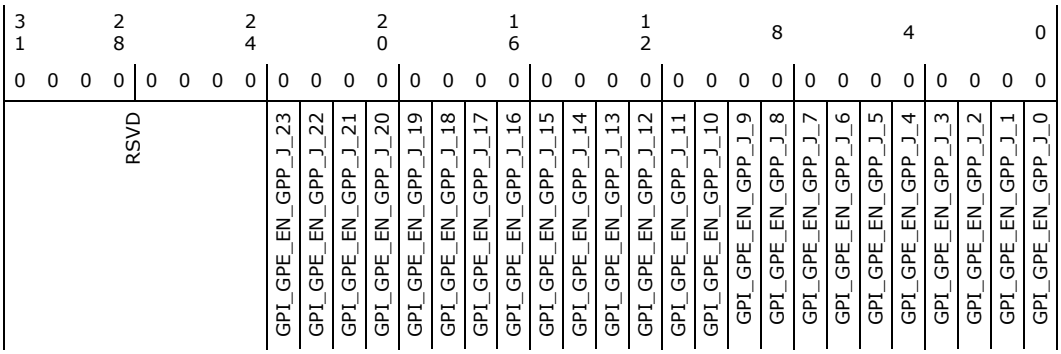
18.10.20 GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J)—Offset 130h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J_23):** Applied to GPP_J23. Same description as bit 0.
22	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J_22):** Applied to GPP_J22. Same description as bit 0.
21	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J_21):** Applied to GPP_J21. Same description as bit 0.
20	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J_20):** Applied to GPP_J20. Same description as bit 0.
19	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J_19):** Applied to GPP_J19. Same description as bit 0.
18	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J_18):** Applied to GPP_J18. Same description as bit 0.
17	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J_17):** Applied to GPP_J17. Same description as bit 0.
16	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J_16):** Applied to GPP_J16. Same description as bit 0.
15	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J_15):** Applied to GPP_J15. Same description as bit 0.
14	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J_14):** Applied to GPP_J14. Same description as bit 0.
13	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J_13):** Applied to GPP_J13. Same description as bit 0.
12	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J_12):** Applied to GPP_J12. Same description as bit 0.
11	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J_11):** Applied to GPP_J11. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
10	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J_10):** Applied to GPP_J10. Same description as bit 0.
9	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J_9):** Applied to GPP_J9. Same description as bit 0.
8	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J_8):** Applied to GPP_J8. Same description as bit 0.
7	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J_7):** Applied to GPP_J7. Same description as bit 0.
6	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J_6):** Applied to GPP_J6. Same description as bit 0.
5	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J_5):** Applied to GPP_J5. Same description as bit 0.
4	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J_4):** Applied to GPP_J4. Same description as bit 0.
3	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J_3):** Applied to GPP_J3. Same description as bit 0.
2	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J_2):** Applied to GPP_J2. Same description as bit 0.
1	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J_1):** Applied to GPP_J1. Same description as bit 0.
0	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_J_0):** This bit is used to enable/disable the generation of GPE to cause SCI and/or wake when the corresponding GPI_GPE_STS bit is set. 0 = disable GPE generation 1 = enable GPE generation ***Note:*** The pad must also be routed for GPE functionality in order for GPE to be generated, i.e. the corresponding GPIRoutSCI must be set to '1'.

18.10.21 GPI General Purpose Events Enable (GPI_GPE_EN_GPP_K)—Offset 134h

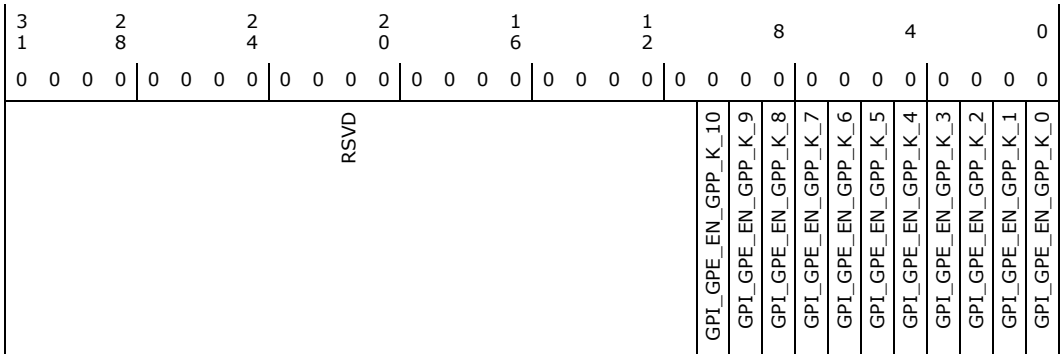
Same description as PI_GPI_GPE_EN_GPP_J register, except that this is for GPP_K[10:0].

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:11	0h RO	Reserved
10	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_K_10):** Applied to GPP_K10. Same description as bit 0.
9	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_K_9):** Applied to GPP_K9. Same description as bit 0.
8	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_K_8):** Applied to GPP_K8. Same description as bit 0.
7	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_K_7):** Applied to GPP_K7. Same description as bit 0.
6	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_K_6):** Applied to GPP_K6. Same description as bit 0.
5	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_K_5):** Applied to GPP_K5. Same description as bit 0.
4	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_K_4):** Applied to GPP_K4. Same description as bit 0.
3	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_K_3):** Applied to GPP_K3. Same description as bit 0.
2	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_K_2):** Applied to GPP_K2. Same description as bit 0.
1	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_K_1):** Applied to GPP_K1. Same description as bit 0.
0	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_K_0):** This bit is used to enable/disable the generation of GPE to cause SCI and/or wake when the corresponding GPI_GPE_STS bit is set. 0 = disable GPE generation 1 = enable GPE generation ***Note:*** The pad must also be routed for GPE functionality in order for GPE to be generated, i.e. the corresponding GPIRoutSCI must be set to '1'.

18.10.22 Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_0)—Offset 400h

This register applies to GPP_J0.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD	PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD	GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.23 Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_0)—Offset 404h

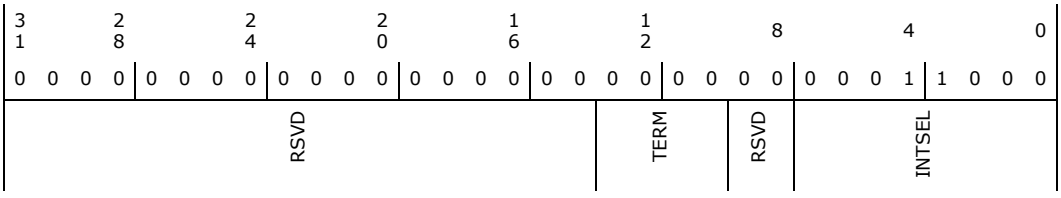
This register applies to GPP_J0.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 18h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	18h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.10.24 Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_1)—Offset 408h

This register applies to GPP_J1 and has the same description as PAD_CFG_DW0_GPP_J_0.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG		PreGfRXSel	RXINV	RSVD		GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD							PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD					GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.25 Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_1)—Offset 40Ch

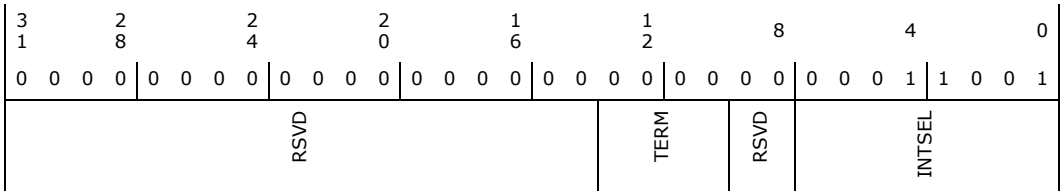
This register applies to GPP_J1.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 19h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	19h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.10.26 Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_2)—Offset 410h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.27 Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_2)—Offset 414h

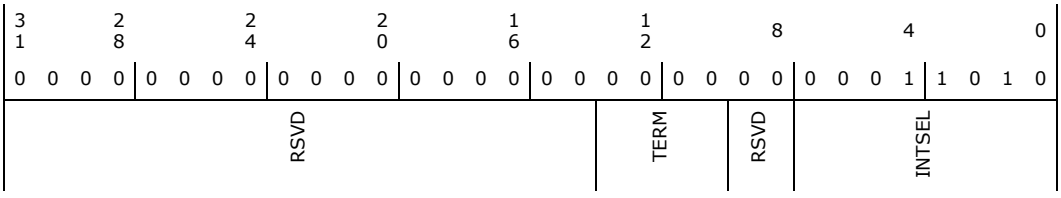
This register applies to GPP_J2.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Ah



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Ah RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.10.28 Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_3)—Offset 418h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG		PreGfRXSel	RXINV	RSVD		GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD							PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD					GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.29 Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_3)—Offset 41Ch

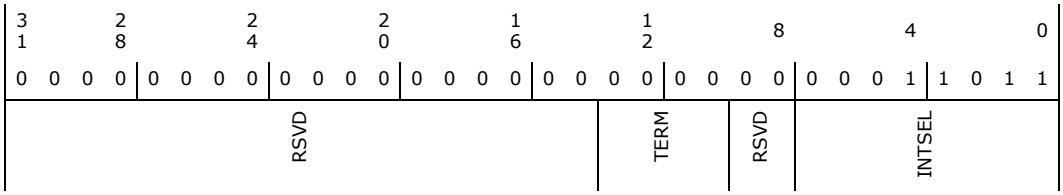
This register applies to GPP_J3.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Bh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Bh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

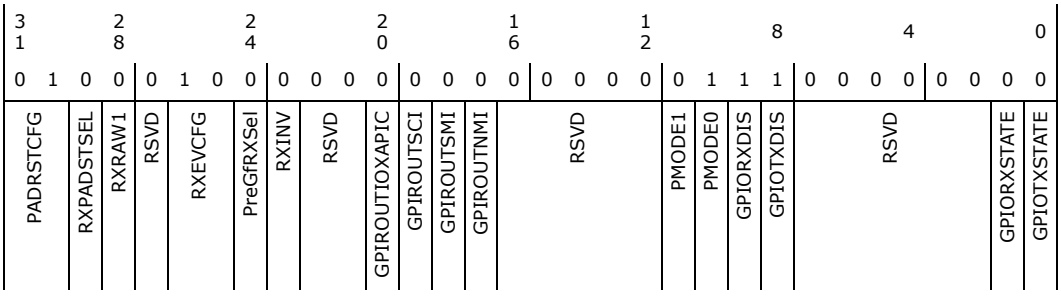
18.10.30 Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_4)—Offset 420h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', 1 = Routing can cause NMI. ***Note:*** GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.31 Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_4)—Offset 424h

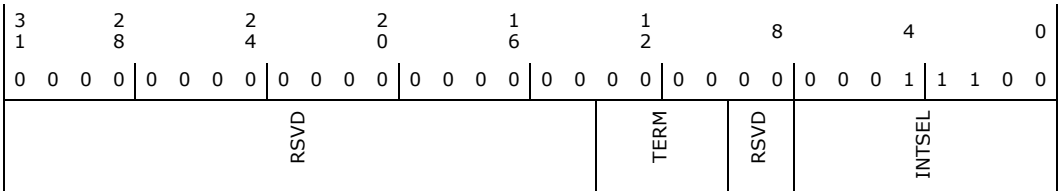
This register applies to GPP_J4.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Ch



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Ch RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.10.32 Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_5)—Offset 428h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG		PreGfRXSel	RXINV	RSVD		GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD							PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD					GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.33 Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_5)—Offset 42Ch

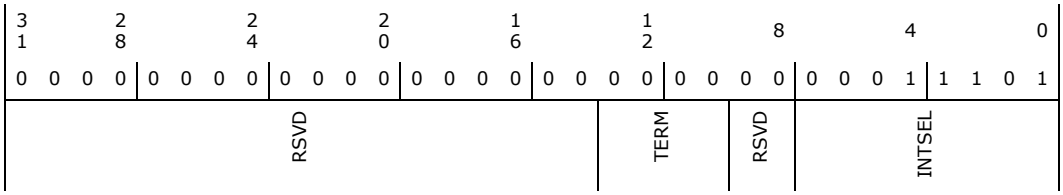
This register applies to GPP_J5.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Dh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Dh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.10.34 Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_6)—Offset 430h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.35 Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_6)—Offset 434h

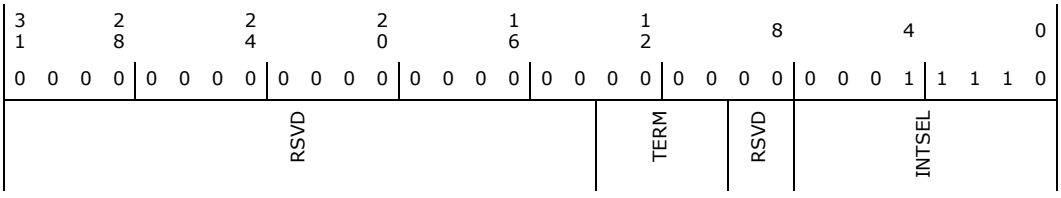
This register applies to GPP_J6.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Eh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Eh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.10.36 Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_7)—Offset 438h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG		PreGfRXSel	RXINV	RSVD		GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD							PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD					GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.37 Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_7)—Offset 43Ch

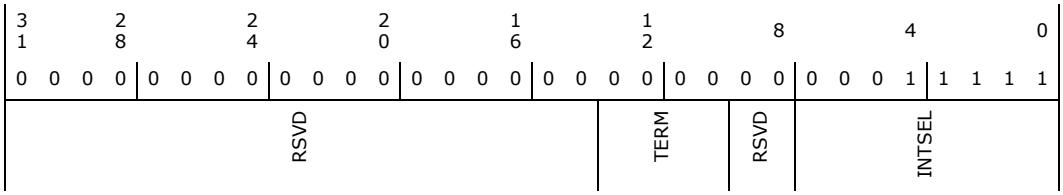
This register applies to GPP_J7.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Fh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	1Fh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.10.38 Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_8)—Offset 440h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.39 Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_8)—Offset 444h

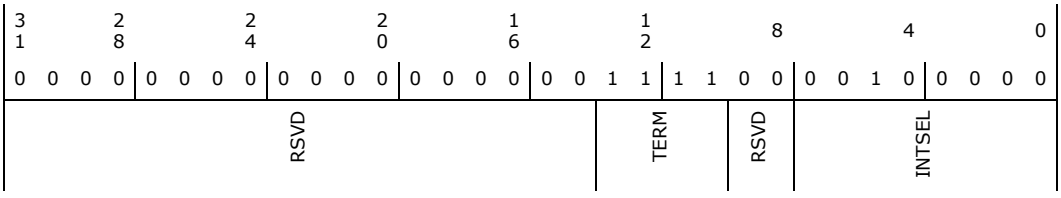
This register applies to GPP_J8.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003C20h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	1111b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu 1111: Native function selected by Pad Mode controls the termination All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	20h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.10.40 Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_9)—Offset 448h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG		PreGfRXSel	RXINV	RSVD		GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD					PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD					GPIORXSTATE	GPIOTXSTATE		

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.41 Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_9)—Offset 44Ch

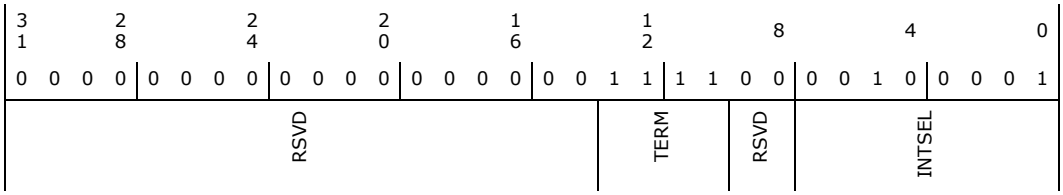
This register applies to GPP_J9.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003C21h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	1111b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu 1111: Native function selected by Pad Mode controls the termination All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	21h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

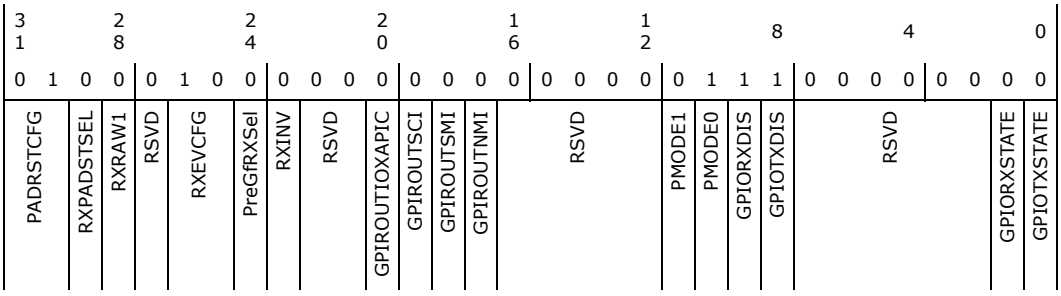
18.10.42 Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_10)—Offset 450h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.43 Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_10)—Offset 454h

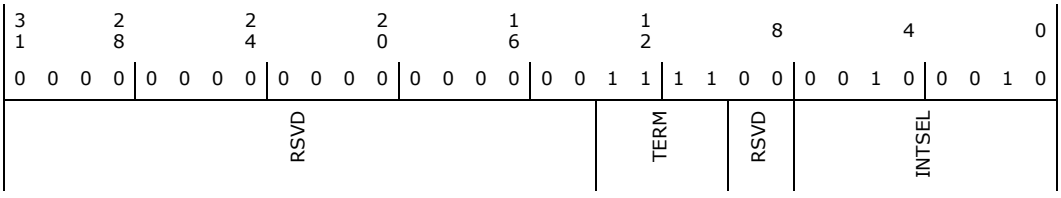
This register applies to GPP_J10.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003C22h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	1111b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu 1111: Native Function selected by Pad Mode controls the termination All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	22h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.10.44 Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_11)—Offset 458h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG		PreGfRXSel	RXINV	RSVD		GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD					PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD					GPIORXSTATE	GPIOTXSTATE		

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.45 Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_11)—Offset 45Ch

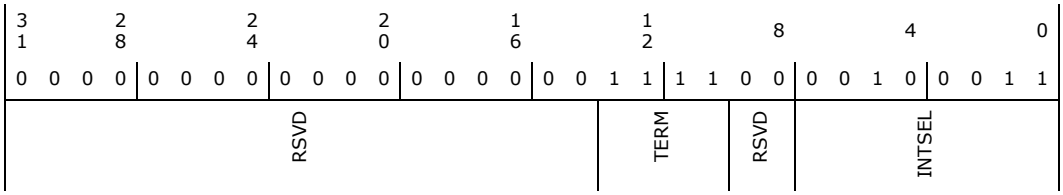
This register applies to GPP_J11.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003C23h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	1111b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu 1111: Native function selected by Pad Mode controls termination All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	23h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

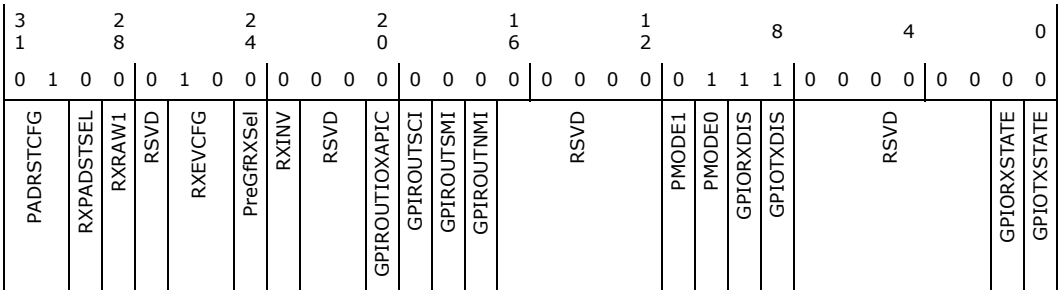
18.10.46 Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_12)—Offset 460h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.47 Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_12)—Offset 464h

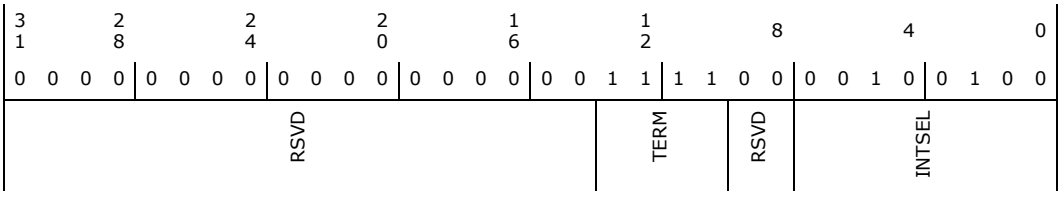
This register applies to GPP_J12.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003C24h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu 1111: Native function selected by Pad Mode selects termination All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	24h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

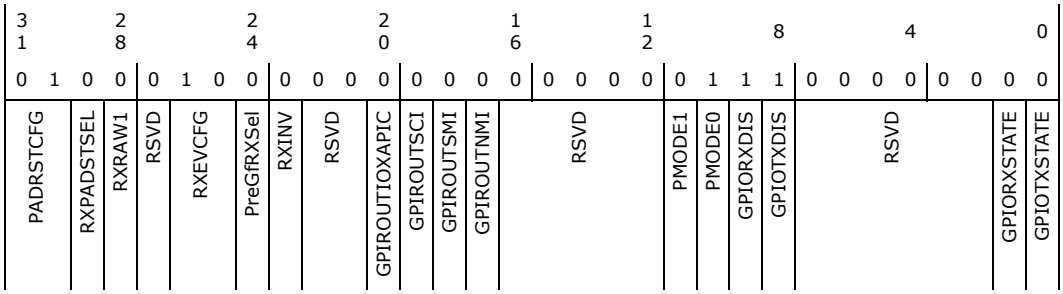
18.10.48 Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_13)—Offset 468h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.49 Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_13)—Offset 46Ch

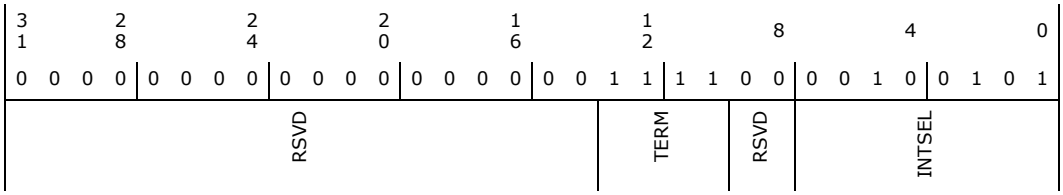
This register applies to GPP_J13.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003C25h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu 1111: Native Function selected by Pad Mode determines termination All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	25h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

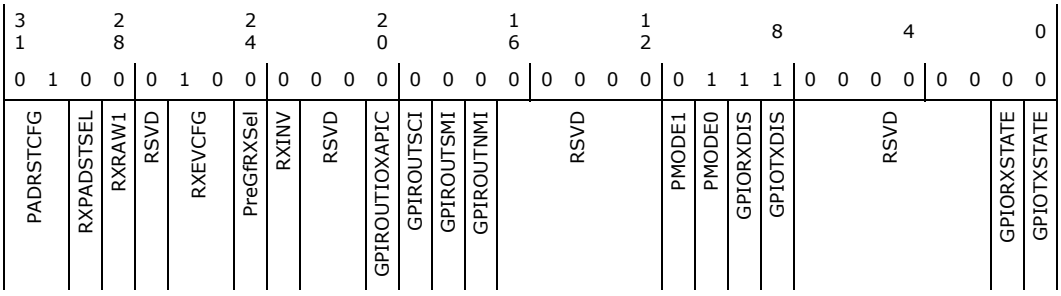
18.10.50 Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_14)—Offset 470h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.51 Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_14)—Offset 474h

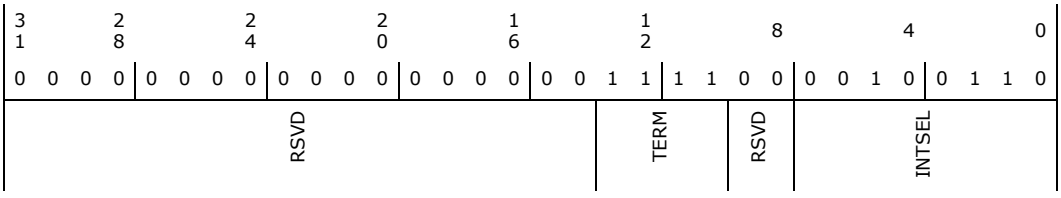
This register applies to GPP_J14.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003C26h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	1111b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu 1111: The Native fiction selected by Pad Mode determines the termination. All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	26h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.10.52 Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_15)—Offset 478h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG		PreGfRXSel	RXINV	RSVD		GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD					PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD					GPIORXSTATE	GPIOTXSTATE		

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.53 Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_15)—Offset 47Ch

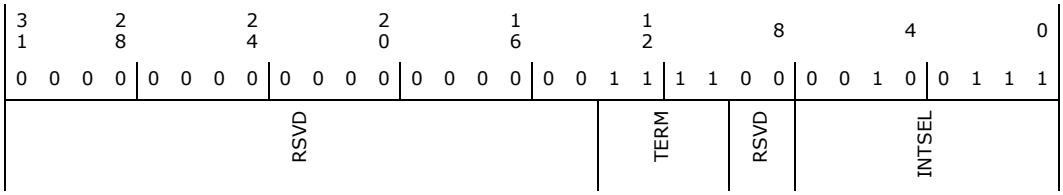
This register applies to GPP_J15.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003C27h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	1111b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu 1111: The Native function selected by Pad Mode determines the termination. All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	27h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.10.54 Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_16)—Offset 480h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.55 Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_16)—Offset 484h

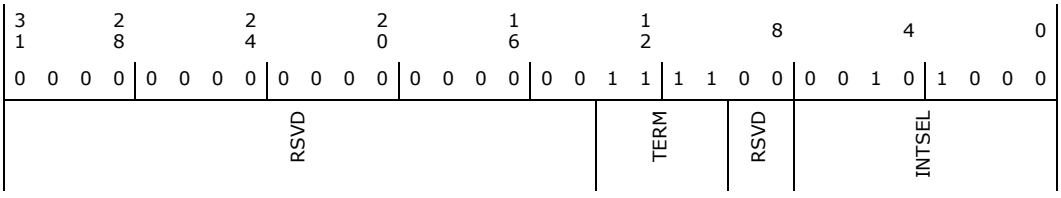
This register applies to GPP_J16.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003C28h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	1111b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu 1111: The Native function selected by Pad Mode determines the termination. All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	28h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.10.56 Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_17)—Offset 488h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG		PreGfRXSel	RXINV	RSVD		GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD					PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD					GPIORXSTATE	GPIOTXSTATE		

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SC ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.57 Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_17)—Offset 48Ch

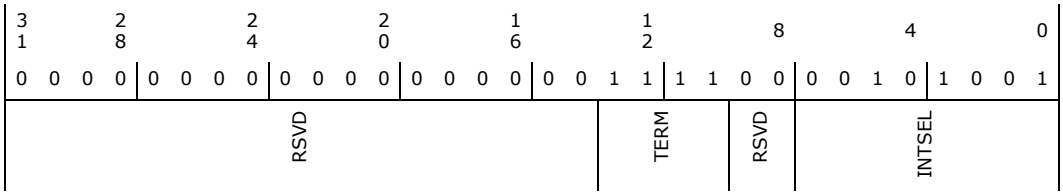
This register applies to GPP_J17.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003C29h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	1111b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu 1111: The Native function selected by Pad Mode determines the termination. All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	29h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.10.58 Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_18)—Offset 490h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.59 Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_18)—Offset 494h

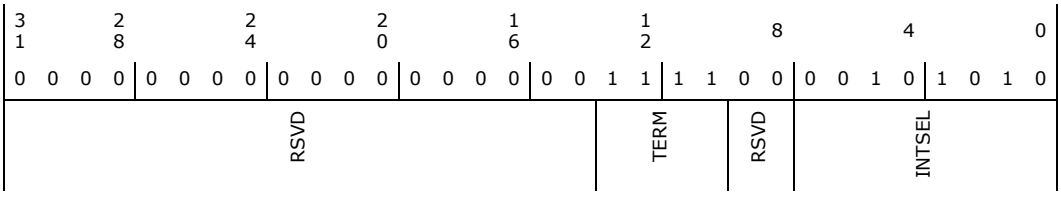
This register applies to GPP_J18.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003C2Ah



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	1111b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu 1111: Native function selected by Pad Mode determines the termination. All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	2Ah RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.10.60 Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_19)—Offset 498h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG		PreGfRXSel	RXINV	RSVD		GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD					PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD					GPIORXSTATE	GPIOTXSTATE		

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.61 Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_19)—Offset 49Ch

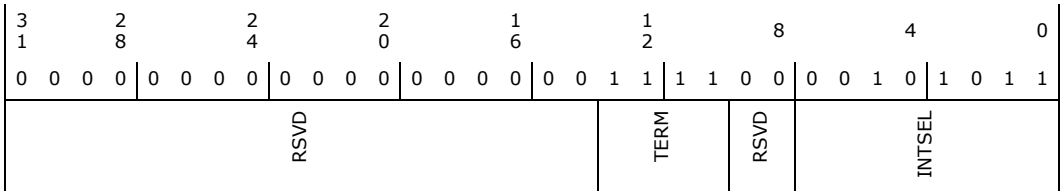
This register applies to GPP_J19.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003C2Bh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	1111b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu 1111: Native function selected by Pad Mode determines the termination. All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	2Bh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

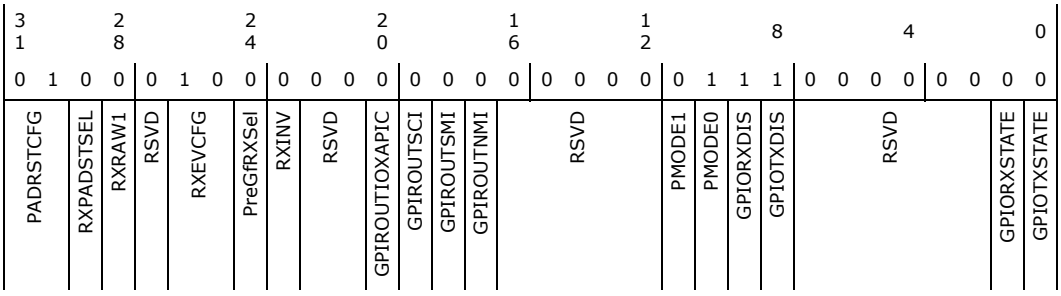
18.10.62 Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_20)—Offset 4A0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.63 Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_20)—Offset 4A4h

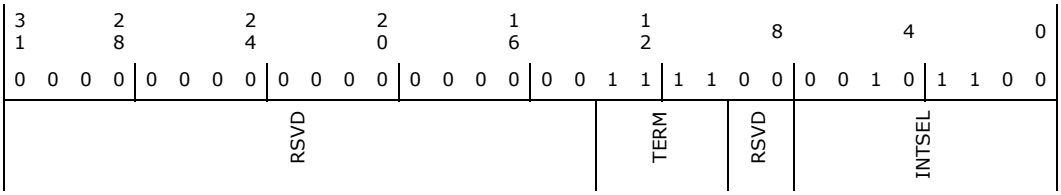
This register applies to GPP_J20.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003C2Ch



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	1111b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu 1111: Native function selected by Pad Mode determines the termination. All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	2Ch RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

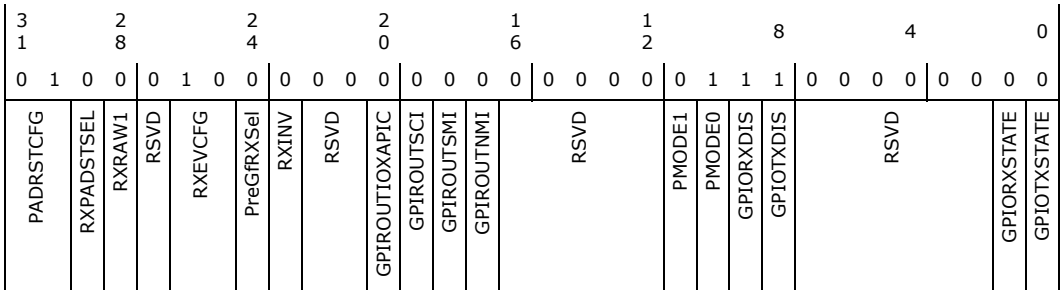
18.10.64 Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_21)—Offset 4A8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.65 Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_21)—Offset 4ACh

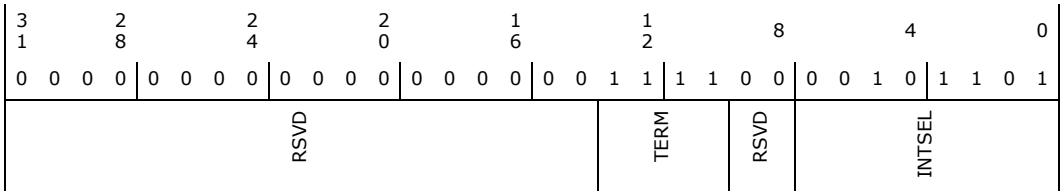
This register applies to GPP_J21.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003C2Dh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	1111b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu 1111: Native function selected by Pad Mode determines termination. All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	2Dh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

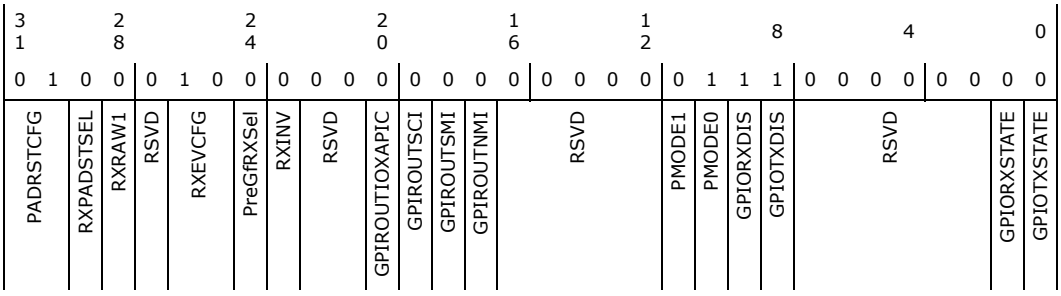
18.10.66 Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_22)—Offset 4B0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.67 Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_22)—Offset 4B4h

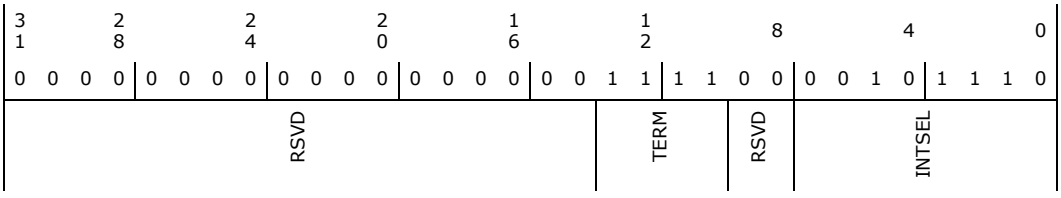
This register applies to GPP_J22.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003C2Eh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	1111b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu 1111: Native function selected by Pad Mode determines termination. All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	2Eh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

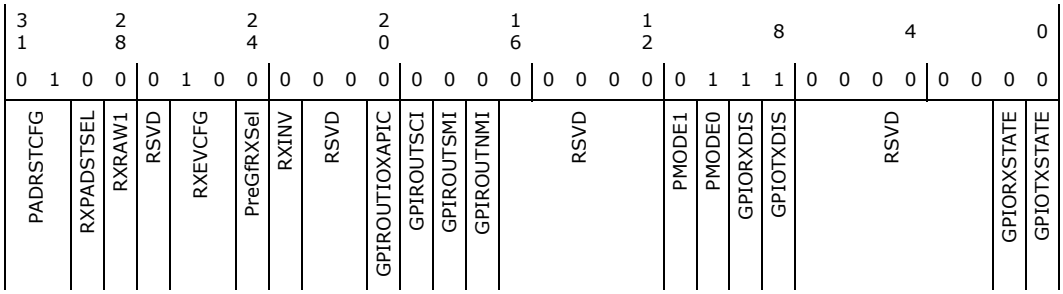
18.10.68 Pad Configuration DW0 (PAD_CFG_DW0_GPP_J_23)—Offset 4B8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.69 Pad Configuration DW1 (PAD_CFG_DW1_GPP_J_23)—Offset 4BCh

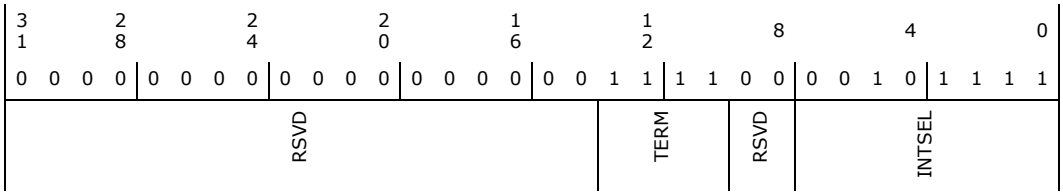
This register applies to GPP_J23.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003C2Fh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	1111b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu 1111: Native function selected by Pad Mode determines termination. All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	2Fh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

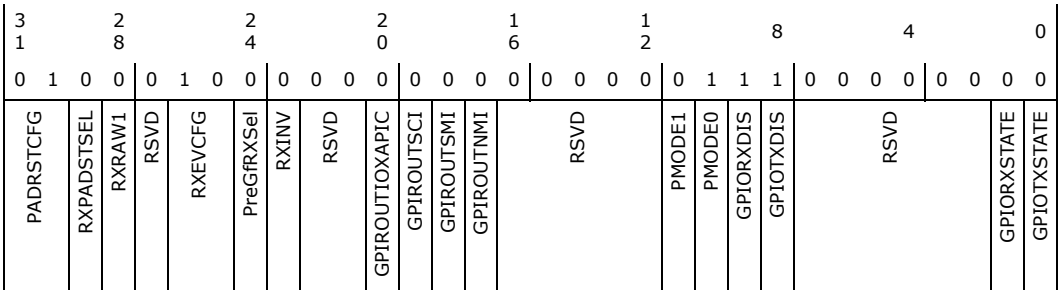
18.10.70 Pad Configuration DW0 (PAD_CFG_DW0_GPP_K_0)—Offset 4C0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.71 Pad Configuration DW1 (PAD_CFG_DW1_GPP_K_0)—Offset 4C4h

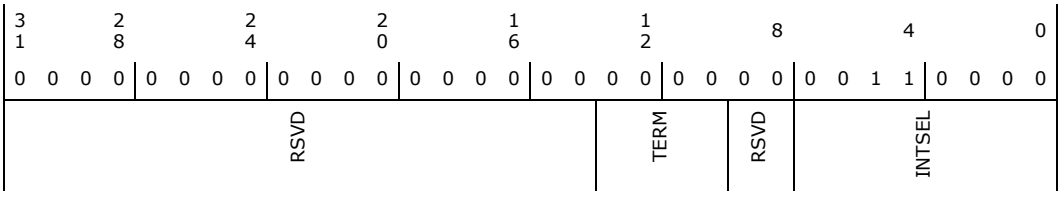
This register applies to GPP_K0.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 30h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	30h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.10.72 Pad Configuration DW0 (PAD_CFG_DW0_GPP_K_1)—Offset 4C8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG		PreGfRXSel	RXINV	RSVD		GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD							PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD					GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.73 Pad Configuration DW1 (PAD_CFG_DW1_GPP_K_1)—Offset 4CCh

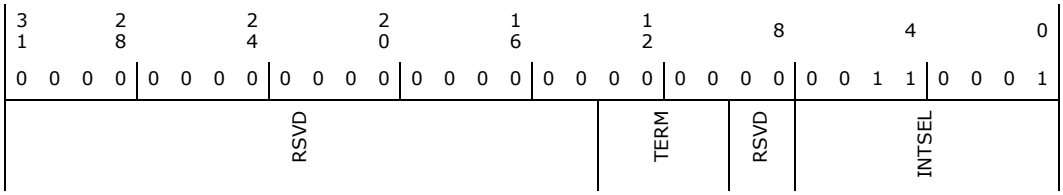
This register applies to GPP_K1.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 31h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	31h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.10.74 Pad Configuration DW0 (PAD_CFG_DW0_GPP_K_2)—Offset 4D0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.75 Pad Configuration DW1 (PAD_CFG_DW1_GPP_K_2)—Offset 4D4h

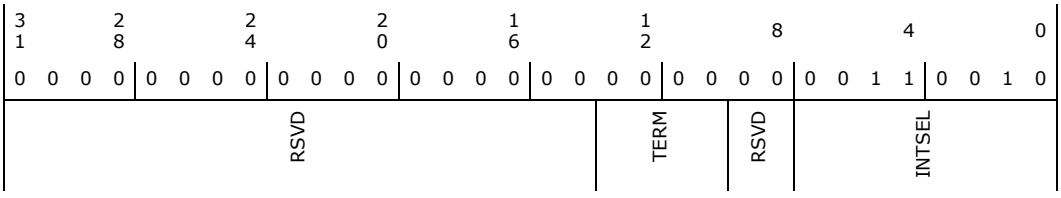
This register applies to GPP_K2.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 32h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	32h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.10.76 Pad Configuration DW0 (PAD_CFG_DW0_GPP_K_3)—Offset 4D8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG			PreGfRXSel	RXINV	RSVD			GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD				PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD								GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.77 Pad Configuration DW1 (PAD_CFG_DW1_GPP_K_3)—Offset 4DCh

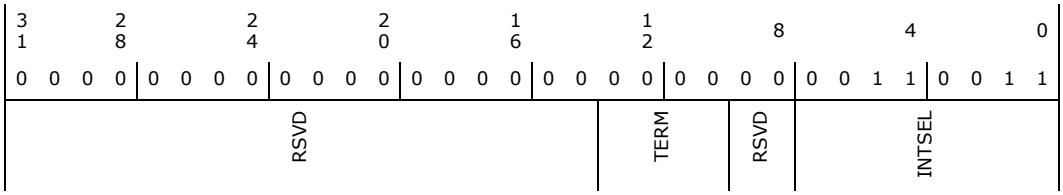
This register applies to GPP_K3.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 33h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	33h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.10.78 Pad Configuration DW0 (PAD_CFG_DW0_GPP_K_4)—Offset 4E0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.79 Pad Configuration DW1 (PAD_CFG_DW1_GPP_K_4)—Offset 4E4h

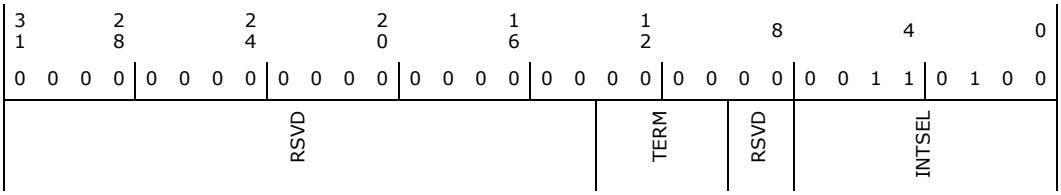
This register applies to GPP_K4.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 34h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	34h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.10.80 Pad Configuration DW0 (PAD_CFG_DW0_GPP_K_5)—Offset 4E8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG			PreGfRXSel	RXINV	RSVD			GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD				PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD								GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.81 Pad Configuration DW1 (PAD_CFG_DW1_GPP_K_5)—Offset 4ECh

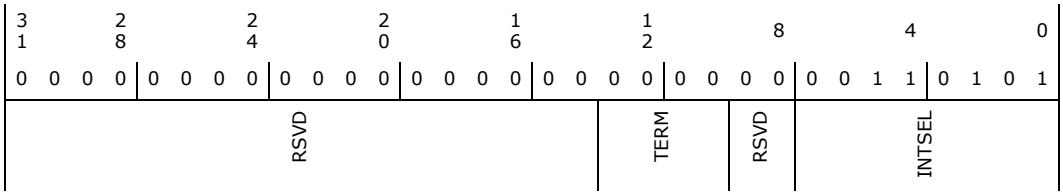
This register applies to GPP_K5.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 35h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	35h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.10.82 Pad Configuration DW0 (PAD_CFG_DW0_GPP_K_6)—Offset 4F0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD		RXEVCFG	PreGfRXSel	RXINV	RSVD		GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD							PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD					GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.83 Pad Configuration DW1 (PAD_CFG_DW1_GPP_K_6)—Offset 4F4h

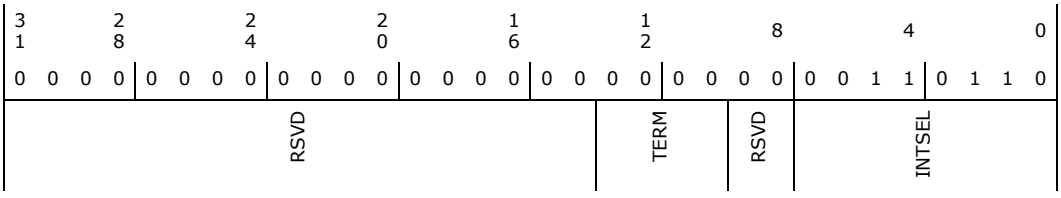
This register applies to GPP_K6.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 36h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	36h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.10.84 Pad Configuration DW0 (PAD_CFG_DW0_GPP_K_7)—Offset 4F8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG			PreGfRXSel	RXINV	RSVD			GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD				PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.85 Pad Configuration DW1 (PAD_CFG_DW1_GPP_K_7)—Offset 4FCh

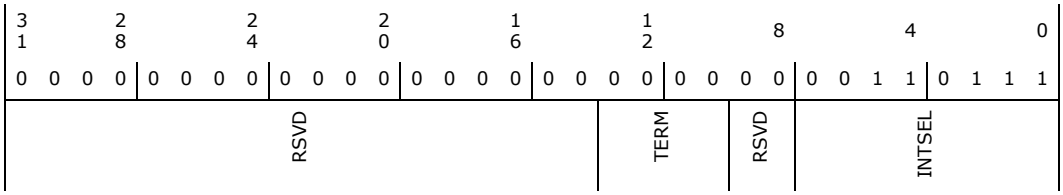
This register applies to GPP_K7.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 37h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	37h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.10.86 Pad Configuration DW0 (PAD_CFG_DW0_GPP_K_8)—Offset 500h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.87 Pad Configuration DW1 (PAD_CFG_DW1_GPP_K_8)—Offset 504h

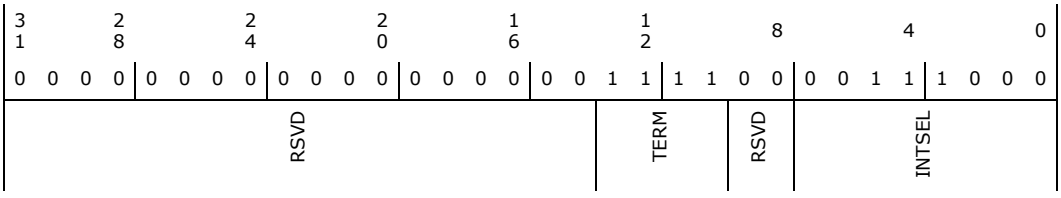
This register applies to GPP_K8.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00003C38h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	1111b RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu 1111: Native function selected by Pad Mode determines the termination. All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	38h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.10.88 Pad Configuration DW0 (PAD_CFG_DW0_GPP_K_9)—Offset 508h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG		PreGfRXSel	RXINV	RSVD		GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD					PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD					GPIORXSTATE	GPIOTXSTATE		

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.89 Pad Configuration DW1 (PAD_CFG_DW1_GPP_K_9)—Offset 50Ch

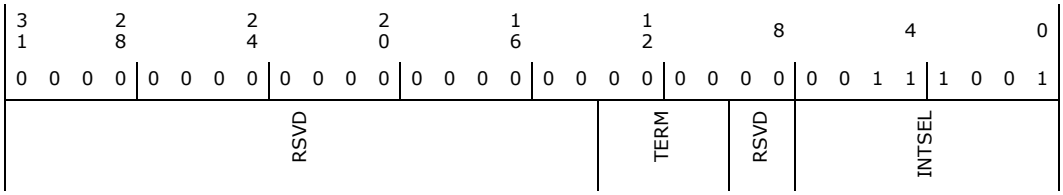
This register applies to GPP_K9.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 39h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	39h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

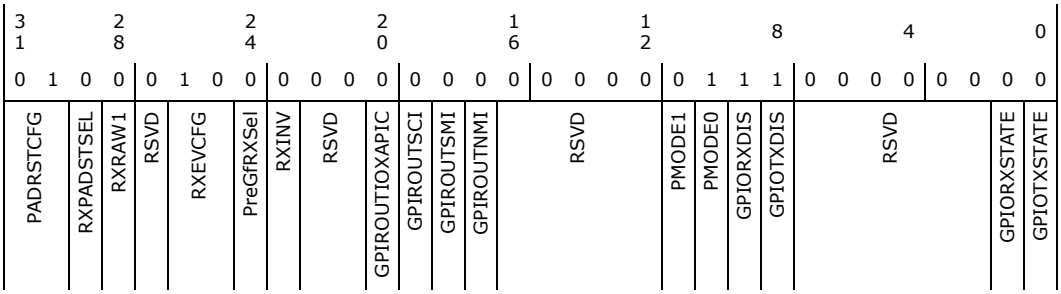
18.10.90 Pad Configuration DW0 (PAD_CFG_DW0_GPP_K_10)—Offset 510h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit lock-down These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e., Hi-Z.
7:2	0h RO	Reserved
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

18.10.91 Pad Configuration DW1 (PAD_CFG_DW1_GPP_K_10)—Offset 514h

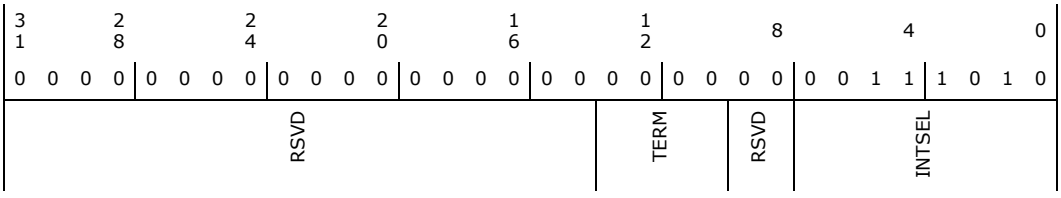
This register applies to GPP_K10.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 3Ah



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved
7:0	3Ah RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11 GPIO Community 5 Registers Summary

Community 5 Registers are for GPP_G, GPP_H and GPP_L groups.

Table 18-8. Summary of GPIO Community 5 Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
Ch	Fh	Pad Base Address (PADBAR)—Offset Ch	00000400h
10h	13h	Miscellaneous Configuration (MISCCFG)—Offset 10h	00007600h
20h	23h	Pad Ownership (PAD_OWN_GPP_G_0)—Offset 20h	00000000h
24h	27h	Pad Ownership (PAD_OWN_GPP_G_1)—Offset 24h	00000000h
28h	2Bh	Pad Ownership (PAD_OWN_GPP_G_2)—Offset 28h	00000000h
2C	2F	Pad Ownership (PAD_OWN_GPP_H_0)—Offset 2Ch	00000000h
30	33	Pad Ownership (PAD_OWN_GPP_H_1)—Offset 30h	00000000h
34	37	Pad Ownership (PAD_OWN_GPP_H_2)—Offset 34h	00000000h
38	3B	Pad Ownership (PAD_OWN_GPP_L_0)—Offset 38h	00000000h
3C	3F	Pad Ownership (PAD_OWN_GPP_L_1)—Offset 3Ch	00000000h
40	43	Pad Ownership (PAD_OWN_GPP_L_2)—Offset 40h	00000000h
60h	63	Pad Configuration Lock (PADCFGLOCK_GPP_G)—Offset 60h	00000000h
64	67	Pad Configuration Lock (PADCFGLOCKTX_GPP_G)—Offset 64h	00000000h
68	6B	Pad Configuration Lock (PADCFGLOCK_GPP_H)—Offset 68h	00000000h
6C	6F	Pad Configuration Lock (PADCFGLOCKTX_GPP_H)—Offset 6Ch	00000000h
70	73	Pad Configuration Lock (PADCFGLOCK_GPP_L)—Offset 70h	00000000h
74	77	Pad Configuration Lock (PADCFGLOCKTX_GPP_L)—Offset 74h	00000000h
80h	83h	Host Software Pad Ownership (HOSTSW_OWN_GPP_G)—Offset 80h	00000000h
84h	87h	Host Software Pad Ownership (HOSTSW_OWN_GPP_H)—Offset 84h	00000000h

Table 18-8. Summary of GPIO Community 5 Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
88	8Bh	Host Software Pad Ownership (HOSTSW_OWN_GPP_L)—Offset 88h	00000000h
100h	103h	GPI Interrupt Status (GPI_IS_GPP_G)—Offset 100h	00000000h
104h	107h	GPI Interrupt Status (GPI_IS_GPP_H)—Offset 104h	00000000h
108h	10Bh	GPI Interrupt Status (GPI_IS_GPP_L)—Offset 108h	00000000h
110h	113h	GPI Interrupt Enable (GPI_IE_GPP_G)—Offset 110h	00000000h
114h	117h	GPI Interrupt Enable (GPI_IE_GPP_H)—Offset 114h	00000000h
118h	11Ch	GPI Interrupt Enable (GPI_IE_GPP_L)—Offset 118h	00000000h
120h	123h	GPI General Purpose Events Status (GPI_GPE_STS_GPP_G)—Offset 120h	00000000h
124h	127h	GPI General Purpose Events Status (GPI_GPE_STS_GPP_H)—Offset 124h	00000000h
128h	12Ch	GPI General Purpose Events Status (GPI_GPE_STS_GPP_L)—Offset 128h	00000000h
130h	133h	GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G)—Offset 130h	00000000h
134h	137h	GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H)—Offset 134h	00000000h
138h	13Bh	GPI General Purpose Events Enable (GPI_GPE_EN_GPP_L)—Offset 133h	00000000h
400h	403h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_0)—Offset 400h	44000300h
404h	407h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_0)—Offset 404h	0000006Dh
408h	40Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_1)—Offset 408h	44000300h
40Ch	40Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_1)—Offset 40Ch	0000006Eh
410h	413h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_2)—Offset 410h	44000300h
414h	417h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_2)—Offset 414h	0000006Fh
418h	41Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_3)—Offset 418h	44000300h
41Ch	41Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_3)—Offset 41Ch	00000070h
420h	423h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_4)—Offset 420h	44000300h
424h	427h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_4)—Offset 424h	00000071h
428h	42Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_5)—Offset 428h	44000300h
42Ch	42Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_5)—Offset 42Ch	00000072h
430h	433h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_6)—Offset 430h	44000300h
434h	437h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_6)—Offset 434h	00000073h
438h	43Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_7)—Offset 438h	44000300h
43Ch	43Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_7)—Offset 43Ch	00000074h
440h	443h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_8)—Offset 440h	44000700h
444h	447h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_8)—Offset 444h	00000075h
448h	44Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_9)—Offset 448h	44000700h
44Ch	44Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_9)—Offset 44Ch	00000076h
450h	453h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_10)—Offset 450h	44000700h
454h	457h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_10)—Offset 454h	00000077h
458h	45Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_11)—Offset 458h	44000700h
45Ch	45Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_11)—Offset 45Ch	0000002Ch
460h	463h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_12)—Offset 460h	44000300h
464h	467h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_12)—Offset 464h	0000002Dh
468h	46Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_13)—Offset 468h	44000300h
46Ch	46Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_13)—Offset 46Ch	0000002Eh

Table 18-8. Summary of GPIO Community 5 Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
470h	473h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_14)—Offset 470h	44000300h
474h	477h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_14)—Offset 474h	0000002Fh
478h	47Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_15)—Offset 478h	44000300h
47Ch	47Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_15)—Offset 47Ch	00000030h
480h	483h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_16)—Offset 480h	44000300h
484h	487h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_16)—Offset 484h	00000031h
488h	48Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_17)—Offset 488h	44000300h
48Ch	48Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_17)—Offset 48Ch	00000032h
490h	493h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_18)—Offset 490h	44000300h
494h	497h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_18)—Offset 494h	00000033h
498h	49Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_19)—Offset 498h	44000300h
49Ch	49Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_19)—Offset 49Ch	00000034h
4A0h	4A3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_20)—Offset 4A0h	44000300h
4A4h	4A7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_20)—Offset 4A4h	00000035h
4A8h	4ABh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_21)—Offset 4A8h	44000300h
4ACh	4AFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_21)—Offset 4ACh	00000036h
4B0h	4B3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_22)—Offset 4B0h	44000300h
4B4h	4B7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_22)—Offset 4B4h	00000037h
4B8h	4BBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_23)—Offset 4B8h	44000300h
4BCh	4BFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_23)—Offset 4BCh	00000038h
4C0h	4C3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_0)—Offset 4C0h	44000300h
4C4h	4C7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_0)—Offset 4C4h	00000039h
4C8h	4CBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_1)—Offset 4C8h	44000300h
4CCh	4CFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_1)—Offset 4CCh	0000003Ah
4D0h	4D3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_2)—Offset 4D0h	44000300h
4D4h	4D7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_2)—Offset 4D4h	0000003Bh
4D8h	4DBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_3)—Offset 4D8h	44000300h
4DCh	4DFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_3)—Offset 4DCh	0000003Ch
4E0h	4E3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_4)—Offset 4E0h	44000300h
4E4h	4E7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_4)—Offset 4E4h	0000003Dh
4E8h	4EBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_5)—Offset 4E8h	44000300h
4ECh	4EFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_5)—Offset 4ECh	0000003Eh
4F0h	4F3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_6)—Offset 4F0h	44000300h
4F4h	4F7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_6)—Offset 4F4h	0000003Fh
4F8h	4FBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_7)—Offset 4F8h	44000300h
4FCh	4FFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_7)—Offset 4FCh	00000040h
500h	503h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_8)—Offset 500h	44000300h
504h	507h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_8)—Offset 504h	00000041h
508h	50Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_9)—Offset 508h	44000300h
50Ch	50Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_9)—Offset 50Ch	00000042h
510h	513h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_10)—Offset 510h	44000300h

Table 18-8. Summary of GPIO Community 5 Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
514h	517h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_10)—Offset 514h	00000043h
518h	51Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_11)—Offset 518h	44000300h
51Ch	51Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_11)—Offset 51Ch	00000044h
520h	523h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_12)—Offset 520h	44000200h
524h	527h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_12)—Offset 524h	00000045h
528h	52Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_13)—Offset 528h	44000300h
52Ch	52Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_13)—Offset 52Ch	00000046h
530h	533h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_14)—Offset 530h	44000300h
534h	537h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_14)—Offset 534h	00000047h
538h	53Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_15)—Offset 538h	44000300h
53Ch	53Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_15)—Offset 53Ch	00000048h
540h	543h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_16)—Offset 540h	44000300h
544h	547h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_16)—Offset 544h	00000049h
548h	54Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_17)—Offset 548h	44000300h
54Ch	54Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_17)—Offset 54Ch	0000004Ah
550h	553h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_18)—Offset 550h	44000300h
554h	557h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_18)—Offset 554h	0000004Bh
558h	55Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_19)—Offset 558h	44000300h
55Ch	55Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_19)—Offset 55Ch	0000004Ch
560h	563h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_20)—Offset 560h	44000300h
564h	567h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_20)—Offset 564h	0000004Dh
568h	56Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_21)—Offset 568h	44000300h
56Ch	56Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_21)—Offset 56Ch	0000004Eh
570h	573h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_22)—Offset 570h	44000300h
574h	577h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_22)—Offset 574h	0000004Fh
578h	57Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_23)—Offset 578h	44000300h
57Ch	57Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_23)—Offset 57Ch	00000050h
580h	583h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_0)—Offset 580h	44000300h
584h	587h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_0)—Offset 584h	00000018h
588h	58Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_1)—Offset 588h	44000700h
58Ch	58Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_1)—Offset 58Ch	00000019h
590h	593h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_2)—Offset 590h	44000600h
594h	597h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_2)—Offset 594h	0000001Ah
598h	59Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_3)—Offset 598h	44000600h
59Ch	59Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_3)—Offset 59Ch	0000001Bh
5A0h	5A3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_4)—Offset 5A0h	44000600h
5A4h	5A7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_4)—Offset 5A4h	0000001Ch
5A8h	5ABh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_5)—Offset 5A8h	44000600h
5ACh	5AFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_5)—Offset 5ACh	0000001Dh
5B0h	5B3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_6)—Offset 5B0h	44000600h
5B4h	5B7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_6)—Offset 5B4h	0000001Eh

Table 18-8. Summary of GPIO Community 5 Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
5B8h	5BBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_7)—Offset 5B8h	44000600h
5BCh	5BFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_7)—Offset 5BCh	0000001Fh
5C0h	5C3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_8)—Offset 5C0h	44000600h
5C4h	5C7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_8)—Offset 5C4h	00000020h
5C8h	5CBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_9)—Offset 5C8h	44000600h
5CCh	5CFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_9)—Offset 5CCh	00000021h
5D0h	5D3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_10)—Offset 5D0h	44000600h
5D4h	5D7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_10)—Offset 5D4h	00000022h
5D8h	5DBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_11)—Offset 5D8h	44000600h
5DCh	5DFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_11)—Offset 5DCh	00000023h
5E0h	5E3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_12)—Offset 5E0h	44000600h
5E4h	5E7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_12)—Offset 5E4h	00000024h
5E8h	5EBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_13)—Offset 5E8h	44000600h
5ECh	5EFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_13)—Offset 5ECh	00000025h
5F0h	5F3h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_14)—Offset 5F0h	44000600h
5F4h	5F7h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_14)—Offset 5F4h	00000026h
5F8h	5FBh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_15)—Offset 5F8h	44000600h
5FCh	5FFh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_15)—Offset 5FCh	00000027h
600h	603h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_16)—Offset 600h	44000600h
604h	607h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_16)—Offset 604h	00000028h
608h	60Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_17)—Offset 608h	44000600h
60Ch	60Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_17)—Offset 60Ch	00000029h
610h	613h	Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_18)—Offset 610h	44000600h
614h	617h	Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_18)—Offset 614h	0000002Ah
618h	61Bh	Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_19)—Offset 618h	44000600h
61Ch	61Fh	Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_19)—Offset 61Ch	0000002Bh

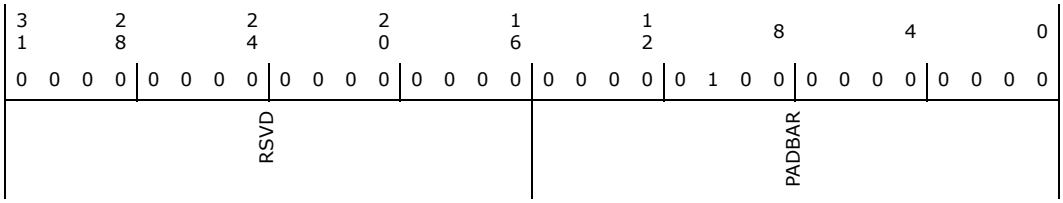
18.11.1 Pad Base Address (PADBAR)—Offset Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 400h



Bit Range	Default & Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:0	400h RO	**Pad Base Address (PADBAR):** This field provides the starting byte-align address of Pad0 register sets. It is meant for software to discover from where the very first Pad register (i.e. Pad0 register) starts to compute the next Pad address offsets.

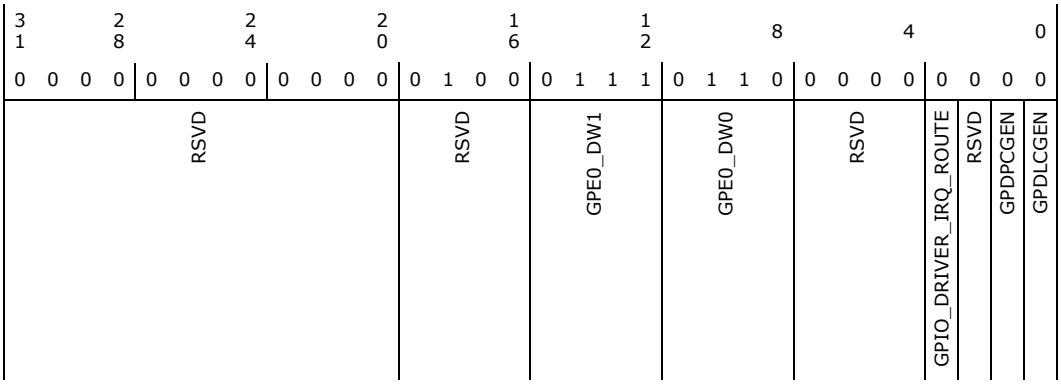
18.11.2 Miscellaneous Configuration (MISCCFG)—Offset 10h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 00007600h



Bit Range	Default & Access	Field Name (ID): Description
31:20	0h RO	Reserved.
19:16	Ch RO	Reserved
15:12	7h RO	**GPIO Group to GPE_DW0 assignment encoding (GPE0_DW1):** This register assigns a specific GPIO Group to the ACPI GPE0[63:32]. 1h = GPP_B[23:0] mapped to GPE[55:32] and GPE[63:56] is not used 2h = GPP_C[23:0] mapped to GPE[55:32] and GPE[63:56] is not used] 3h = GPP_D[23:0] mapped to GPE[55:32] and GPE[63:56] is not used 4h = GPP_E[12:0] mapped to GPE[44:32] and GPE[63:45] is not used 5h = GPP_F[23:0] mapped to GPE[55:32] and GPE[63:56] is not used 6h = GPP_G[23:0] mapped to GPE[55:32] and GPE[63:56] is not used 7h = GPP_H[23:0] mapped to GPE[55:32] and GPE[63:56] is not used 8h = GPP_I[10:0] mapped to GPE[42:32] and GPE[63:43] is not used 9h = GPD[11:0] mapped to GPE[43:32] and GPE[63:44] is not used Ah= GPP_J[23:0] mapped to GPE[55:32] and GPE[63:56] not used Bh= GPP_K[11:0]mapped to GPE[43:32] and GPE[63:44] is not used
11:8	6h RO	**GPIO Group to GPE_DW0 assignment encoding (GPE0_DW0):** This register assigns a specific GPIO Group to the ACPI GPE0[31:0]. 0h = GPP_A[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 1h = GPP_B[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 2h = GPP_C[23:0] mapped to GPE[23:0] and GPE[31:24] is not used] 3h = GPP_D[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 4h = GPP_E[12:0] mapped to GPE[12:0] and GPE[31:13] is not used 5h = GPP_F[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 6h = GPP_G[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 7h = GPP_H[23:0] mapped to GPE[23:0] and GPE[31:24] is not used 8h = GPP_I[10:0] mapped to GPE[10:0] and GPE[31:11] is not used 9h = GPD[11:0] mapped to GPE[11:0] and GPE[31:12] is not used Ah= GPP_J[23:0] mapped to GPE[23:0] and GPE[31:24] not used Bh= GPP_K[11:0]mapped to GPE[11:0] and GPE[31:12] is not used

Bit Range	Default & Access	Field Name (ID): Description
7:4	0h RO	Reserved
3	0h RW	**GPIO Driver IRQ Route (GPIO_DRIVER_IRQ_ROUTE):** Specifies the APIC IRQ globally for all pads within the current community (GPI_IS with corresponding GPI_IE enable). 0 = IRQ14 1 = IRQ15
2	0h RO	Reserved
1	0h RW	**GPIO Dynamic Partition Clock Gating Enable (GPDPCGEN):** Specifies whether the GPIO Community should take part in partition clock gating 0 = Disable participation in dynamic partition clock gating 1 = Enable participation in dynamic partition clock gating
0	0h RW	**GPIO Dynamic Local Clock Gating Enable (GPDLCGEN):** Specifies whether the GPIO Community should perform local clock gating. 0 = Disable dynamic local clock gating 1 = Enable dynamic local clock gating

18.11.3 Pad Ownership (PAD_OWN_GPP_G_0)—Offset 20h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD	PAD_OWN_GPP_G_7			RSVD	PAD_OWN_GPP_G_6			RSVD	PAD_OWN_GPP_G_5			RSVD	PAD_OWN_GPP_G_4			RSVD	PAD_OWN_GPP_G_3			RSVD	PAD_OWN_GPP_G_2			RSVD	PAD_OWN_GPP_G_1			RSVD	PAD_OWN_GPP_G_0		

Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RW	**Pad Ownership (PAD_OWN_GPP_G_7):** Same description as bit 0, except that the bit field applies to GPP_G7.
27:26	0h RO	Reserved
25:24	0h RW	**Pad Ownership (PAD_OWN_GPP_G_6):** Same description as bit 0, except that the bit field applies to GPP_G6.
23:22	0h RO	Reserved
21:20	0h RW	**Pad Ownership (PAD_OWN_GPP_G_5):** Same description as bit 0, except that the bit field applies to GPP_G5.
19:18	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
17:16	0h RW	**Pad Ownership (PAD_OWN_GPP_G_4):** Same description as bit 0, except that the bit field applies to GPP_G4.
15:14	0h RO	Reserved
13:12	0h RW	**Pad Ownership (PAD_OWN_GPP_G_3):** Same description as bit 0, except that the bit field applies to GPP_G3.
11:10	0h RO	Reserved
9:8	0h RW	**Pad Ownership (PAD_OWN_GPP_G_2):** Same description as bit 0, except that the bit field applies to GPP_G2.
7:6	0h RO	Reserved
5:4	0h RW	**Pad Ownership (PAD_OWN_GPP_G_1):** Same description as bit 0, except that the bit field applies to GPP_G1.
3:2	0h RO	Reserved
1:0	0h RW	**Pad Ownership (PAD_OWN_GPP_G_0):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = ME GPIO Mode. ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad.

18.11.4 Pad Ownership (PAD_OWN_GPP_G_1)—Offset 24h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD	PAD_OWN_GPP_G_15	RSVD	PAD_OWN_GPP_G_14	RSVD	PAD_OWN_GPP_G_13	RSVD	PAD_OWN_GPP_G_12	RSVD	PAD_OWN_GPP_G_11	RSVD	PAD_OWN_GPP_G_10	RSVD	PAD_OWN_GPP_G_9	RSVD	PAD_OWN_GPP_G_8																

Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RW	**Pad Ownership (PAD_OWN_GPP_G_15):** Same description as bit 0, except that the bit field applies to GPP_G15.
27:26	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
25:24	0h RW	**Pad Ownership (PAD_OWN_GPP_G_14):** Same description as bit 0, except that the bit field applies to GPP_G14.
23:22	0h RO	Reserved
21:20	0h RW	**Pad Ownership (PAD_OWN_GPP_G_13):** Same description as bit 0, except that the bit field applies to GPP_G13.
19:18	0h RO	Reserved
17:16	0h RW	**Pad Ownership (PAD_OWN_GPP_G_12):** Same description as bit 0, except that the bit field applies to GPP_G12.
15:14	0h RO	Reserved
13:12	0h RW	**Pad Ownership (PAD_OWN_GPP_G_11):** Same description as bit 0, except that the bit field applies to GPP_G11.
11:10	0h RO	Reserved
9:8	0h RW	**Pad Ownership (PAD_OWN_GPP_G_10):** Same description as bit 0, except that the bit field applies to GPP_G10.
7:6	0h RO	Reserved
5:4	0h RW	**Pad Ownership (PAD_OWN_GPP_G_9):** Same description as bit 0, except that the bit field applies to GPP_G9.
3:2	0h RO	Reserved
1:0	0h RW	**Pad Ownership (PAD_OWN_GPP_G_8):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = ME GPIO Mode. ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad.

18.11.5 Pad Ownership (PAD_OWN_GPP_G_2)—Offset 28h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	28	24	20	16	12	8	4	0
0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0
RSVD	PAD_OWN_GPP_G_23	RSVD	PAD_OWN_GPP_G_22	RSVD	PAD_OWN_GPP_G_21	RSVD	PAD_OWN_GPP_G_20	RSVD	PAD_OWN_GPP_G_19	RSVD	PAD_OWN_GPP_G_18	RSVD	PAD_OWN_GPP_G_17	RSVD	PAD_OWN_GPP_G_16

Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RW	**Pad Ownership (PAD_OWN_GPP_G_23):** Same description as bit 0, except that the bit field applies to GPP_G23.
27:26	0h RO	Reserved
25:24	0h RW	**Pad Ownership (PAD_OWN_GPP_G_22):** Same description as bit 0, except that the bit field applies to GPP_G22.
23:22	0h RO	Reserved
21:20	0h RW	**Pad Ownership (PAD_OWN_GPP_G_21):** Same description as bit 0, except that the bit field applies to GPP_G21.
19:18	0h RO	Reserved
17:16	0h RW	**Pad Ownership (PAD_OWN_GPP_G_20):** Same description as bit 0, except that the bit field applies to GPP_G20.
15:14	0h RO	Reserved
13:12	0h RW	**Pad Ownership (PAD_OWN_GPP_G_19):** Same description as bit 0, except that the bit field applies to GPP_G19.
11:10	0h RO	Reserved
9:8	0h RW	**Pad Ownership (PAD_OWN_GPP_G_18):** Same description as bit 0, except that the bit field applies to GPP_G18.
7:6	0h RO	Reserved
5:4	0h RW	**Pad Ownership (PAD_OWN_GPP_G_17):** Same description as bit 0, except that the bit field applies to GPP_G17.
3:2	0h RO	Reserved
1:0	0h RW	**Pad Ownership (PAD_OWN_GPP_G_16):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = ME GPIO Mode. ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad.

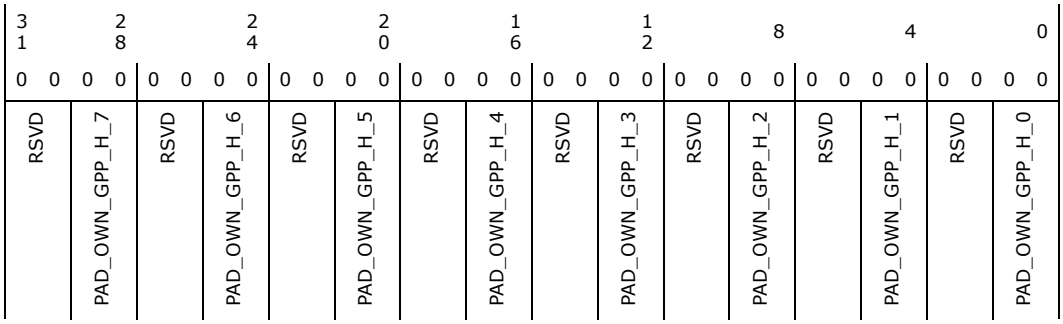
18.11.6 Pad Ownership (PAD_OWN_GPP_H_0)—Offset 2Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RW	**Pad Ownership (PAD_OWN_GPP_H_7):** Same description as bit 0, except that the bit field applies to GPP_H7.
27:26	0h RO	Reserved
25:24	0h RW	**Pad Ownership (PAD_OWN_GPP_H_6):** Same description as bit 0, except that the bit field applies to GPP_H6.
23:22	0h RO	Reserved
21:20	0h RW	**Pad Ownership (PAD_OWN_GPP_H_5):** Same description as bit 0, except that the bit field applies to GPP_H5.
19:18	0h RO	Reserved
17:16	0h RW	**Pad Ownership (PAD_OWN_GPP_H_4):** Same description as bit 0, except that the bit field applies to GPP_H4.
15:14	0h RO	Reserved
13:12	0h RW	**Pad Ownership (PAD_OWN_GPP_H_3):** Same description as bit 0, except that the bit field applies to GPP_H3.
11:10	0h RO	Reserved
9:8	0h RW	**Pad Ownership (PAD_OWN_GPP_H_2):** Same description as bit 0, except that the bit field applies to GPP_H2.
7:6	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
5:4	0h RW	**Pad Ownership (PAD_OWN_GPP_H_1):** Same description as bit 0, except that the bit field applies to GPP_H1.
3:2	0h RO	Reserved
1:0	0h RW	**Pad Ownership (PAD_OWN_GPP_H_0):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = ME GPIO Mode. ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad.

18.11.7 Pad Ownership (PAD_OWN_GPP_H_1)—Offset 30h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD		PAD_OWN_GPP_H_15		RSVD		PAD_OWN_GPP_H_14		RSVD		PAD_OWN_GPP_H_13		RSVD		PAD_OWN_GPP_H_12		RSVD		PAD_OWN_GPP_H_11		RSVD		PAD_OWN_GPP_H_10		RSVD		PAD_OWN_GPP_H_9		RSVD		PAD_OWN_GPP_H_8	

Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RW	**Pad Ownership (PAD_OWN_GPP_H_15):** Same description as bit 0, except that the bit field applies to GPP_H15.
27:26	0h RO	Reserved
25:24	0h RW	**Pad Ownership (PAD_OWN_GPP_H_14):** Same description as bit 0, except that the bit field applies to GPP_H14.
23:22	0h RO	Reserved
21:20	0h RW	**Pad Ownership (PAD_OWN_GPP_H_13):** Same description as bit 0, except that the bit field applies to GPP_H13.
19:18	0h RO	Reserved
17:16	0h RW	**Pad Ownership (PAD_OWN_GPP_H_12):** Same description as bit 0, except that the bit field applies to GPP_H12.
15:14	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
13:12	0h RW	**Pad Ownership (PAD_OWN_GPP_H_11):** Same description as bit 0, except that the bit field applies to GPP_H11.
11:10	0h RO	Reserved
9:8	0h RW	**Pad Ownership (PAD_OWN_GPP_H_10):** Same description as bit 0, except that the bit field applies to GPP_H10.
7:6	0h RO	Reserved
5:4	0h RW	**Pad Ownership (PAD_OWN_GPP_H_9):** Same description as bit 0, except that the bit field applies to GPP_H9.
3:2	0h RO	Reserved
1:0	0h RW	**Pad Ownership (PAD_OWN_GPP_H_8):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = ME GPIO Mode. ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad.

18.11.8 Pad Ownership (PAD_OWN_GPP_H_2)—Offset 34h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD		PAD_OWN_GPP_H_23		RSVD		PAD_OWN_GPP_H_22		RSVD		PAD_OWN_GPP_H_21		RSVD		PAD_OWN_GPP_H_20		RSVD		PAD_OWN_GPP_H_19		RSVD		PAD_OWN_GPP_H_18		RSVD		PAD_OWN_GPP_H_17		RSVD		PAD_OWN_GPP_H_16	

Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RW	**Pad Ownership (PAD_OWN_GPP_H_23):** Same description as bit 0, except that the bit field applies to GPP_H23.
27:26	0h RO	Reserved
25:24	0h RW	**Pad Ownership (PAD_OWN_GPP_H_22):** Same description as bit 0, except that the bit field applies to GPP_H22.
23:22	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
21:20	0h RW	**Pad Ownership (PAD_OWN_GPP_H_21):** Same description as bit 0, except that the bit field applies to GPP_H21.
19:18	0h RO	Reserved
17:16	0h RW	**Pad Ownership (PAD_OWN_GPP_H_20):** Same description as bit 0, except that the bit field applies to GPP_H20.
15:14	0h RO	Reserved
13:12	0h RW	**Pad Ownership (PAD_OWN_GPP_H_19):** Same description as bit 0, except that the bit field applies to GPP_H19.
11:10	0h RO	Reserved
9:8	0h RW	**Pad Ownership (PAD_OWN_GPP_H_18):** Same description as bit 0, except that the bit field applies to GPP_H18.
7:6	0h RO	Reserved
5:4	0h RW	**Pad Ownership (PAD_OWN_GPP_H_17):** Same description as bit 0, except that the bit field applies to GPP_H17.
3:2	0h RO	Reserved
1:0	0h RW	**Pad Ownership (PAD_OWN_GPP_H_16):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = ME GPIO Mode. ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad.

18.11.9 Pad Ownership (PAD_OWN_GPP_L_0)—Offset 38h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD	PAD_OWN_GPP_L_7			RSVD	PAD_OWN_GPP_L_6			RSVD	PAD_OWN_GPP_L_5			RSVD	PAD_OWN_GPP_L_4			RSVD	PAD_OWN_GPP_L_3			RSVD	PAD_OWN_GPP_L_2			RSVD	PAD_OWN_GPP_L_1			RSVD	PAD_OWN_GPP_L_0	

Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RW	**Pad Ownership (PAD_OWN_GPP_L_7):** Same description as bit 0, except that the bit field applies to GPP_L7.
27:26	0h RO	Reserved
25:24	0h RW	**Pad Ownership (PAD_OWN_GPP_L_6):** Same description as bit 0, except that the bit field applies to GPP_L6.
23:22	0h RO	Reserved
21:20	0h RW	**Pad Ownership (PAD_OWN_GPP_L_5):** Same description as bit 0, except that the bit field applies to GPP_L5.
19:18	0h RO	Reserved
17:16	0h RW	**Pad Ownership (PAD_OWN_GPP_L_4):** Same description as bit 0, except that the bit field applies to GPP_L4.
15:14	0h RO	Reserved
13:12	0h RW	**Pad Ownership (PAD_OWN_GPP_L_3):** Same description as bit 0, except that the bit field applies to GPP_L3.
11:10	0h RO	Reserved
9:8	0h RW	**Pad Ownership (PAD_OWN_GPP_L_2):** Same description as bit 0, except that the bit field applies to GPP_L2.
7:6	0h RO	Reserved
5:4	0h RW	**Pad Ownership (PAD_OWN_GPP_L_1):** Same description as bit 0, except that the bit field applies to GPP_L1.
3:2	0h RO	Reserved
1:0	0h RW	**Pad Ownership (PAD_OWN_GPP_L_0):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = ME GPIO Mode. ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad.

18.11.10 Pad Ownership (PAD_OWN_GPP_L_1)—Offset 3Ch

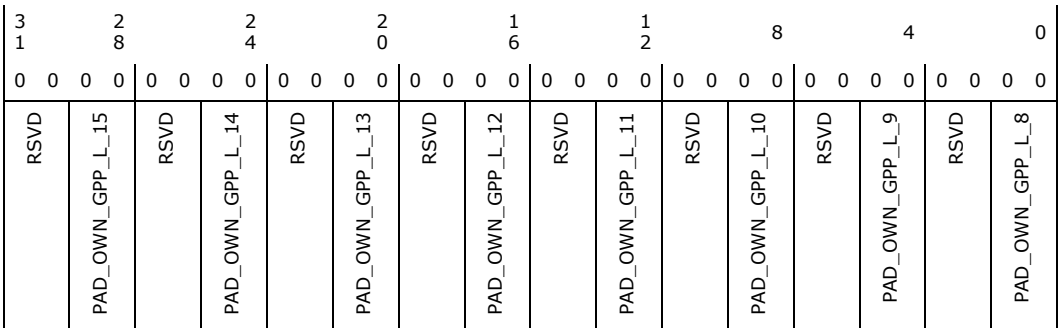
Same description as PAD_OWN_GPP_G_0, except that this register is for GPP_L[15:8].

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:30	0h RO	Reserved
29:28	0h RW	**Pad Ownership (PAD_OWN_GPP_L_15):** Same description as bit 0, except that the bit field applies to GPP_L15.
27:26	0h RO	Reserved
25:24	0h RW	**Pad Ownership (PAD_OWN_GPP_L_14):** Same description as bit 0, except that the bit field applies to GPP_L14.
23:22	0h RO	Reserved
21:20	0h RW	**Pad Ownership (PAD_OWN_GPP_L_13):** Same description as bit 0, except that the bit field applies to GPP_L13.
19:18	0h RO	Reserved
17:16	0h RW	**Pad Ownership (PAD_OWN_GPP_L_12):** Same description as bit 0, except that the bit field applies to GPP_L12.
15:14	0h RO	Reserved
13:12	0h RW	**Pad Ownership (PAD_OWN_GPP_L_11):** Same description as bit 0, except that the bit field applies to GPP_L11.
11:10	0h RO	Reserved
9:8	0h RW	**Pad Ownership (PAD_OWN_GPP_L_10):** Same description as bit 0, except that the bit field applies to GPP_L10.
7:6	0h RO	Reserved

Bit Range	Default & Access	Field Name (ID): Description
5:4	0h RW	**Pad Ownership (PAD_OWN_GPP_L_9):** Same description as bit 0, except that the bit field applies to GPP_L9.
3:2	0h RO	Reserved
1:0	0h RW	**Pad Ownership (PAD_OWN_GPP_L_8):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = ME GPIO Mode. ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad.

18.11.11 Pad Ownership (PAD_OWN_GPP_L_2)—Offset 40h

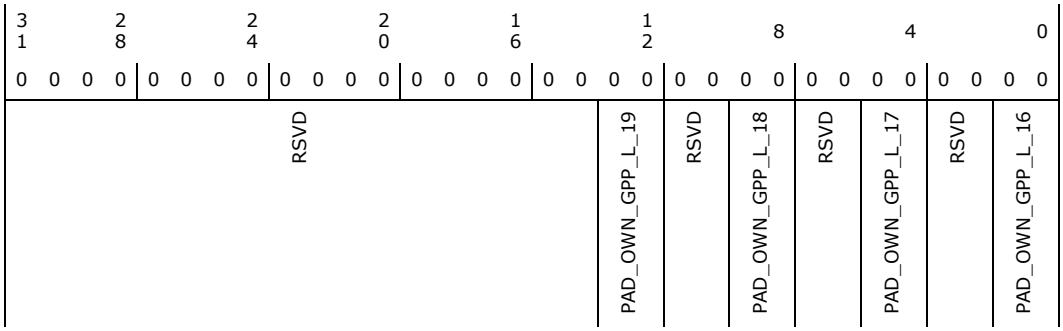
Same description as PAD_OWN_GPP_G_0, except that this register is for GPP_L[23:16].

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:14	00000h RO	Reserved.
13:12	0h RW	**Pad Ownership (PAD_OWN_GPP_L_19):** Same description as bit 0, except that the bit field applies to GPP_L19.
11:10	0h RO	Reserved.
9:8	0h RW	**Pad Ownership (PAD_OWN_GPP_L_18):** Same description as bit 0, except that the bit field applies to GPP_L18.
7:6	0h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
5:4	0h RW	**Pad Ownership (PAD_OWN_GPP_L_17):** Same description as bit 0, except that the bit field applies to GPP_L17.
3:2	0h RO	Reserved.
1:0	0h RW	**Pad Ownership (PAD_OWN_GPP_L_16):** 00 = Host GPIO ACPI Mode or GPIO Driver Mode. Host software (ACPI or GPIO Driver) has ownership of the pad. In Host GPIO Driver Mode (refer to HOSTSW_OWN), GPIO input event update is limited to GPI_STS update only. Otherwise in Host ACPI Mode, updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. 01 = ME GPIO Mode. ME has ownership of the pad. 10 = Reserved 11 = IE GPIO Mode. IE has ownership of the pad.

18.11.12 Pad Configuration Lock (PADCFGLOCK_GPP_G)—Offset 60h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSVD								PADCFGLOCK_GPP_G_23	PADCFGLOCK_GPP_G_22	PADCFGLOCK_GPP_G_21	PADCFGLOCK_GPP_G_20	PADCFGLOCK_GPP_G_19	PADCFGLOCK_GPP_G_18	PADCFGLOCK_GPP_G_17	PADCFGLOCK_GPP_G_16	PADCFGLOCK_GPP_G_15	PADCFGLOCK_GPP_G_14	PADCFGLOCK_GPP_G_13	PADCFGLOCK_GPP_G_12	PADCFGLOCK_GPP_G_11	PADCFGLOCK_GPP_G_10	PADCFGLOCK_GPP_G_9	PADCFGLOCK_GPP_G_8	PADCFGLOCK_GPP_G_7	PADCFGLOCK_GPP_G_6	PADCFGLOCK_GPP_G_5	PADCFGLOCK_GPP_G_4	PADCFGLOCK_GPP_G_3	PADCFGLOCK_GPP_G_2	PADCFGLOCK_GPP_G_1	PADCFGLOCK_GPP_G_0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_G_23):** Applied to GPP_G23. Same description as PADCFGLOCK_GPP_G_0
22	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_G_22):** Applied to GPP_G22. Same description as PADCFGLOCK_GPP_G_0
21	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_G_21):** Applied to GPP_G21. Same description as PADCFGLOCK_GPP_G_0
20	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_G_20):** Applied to GPP_G20. Same description as PADCFGLOCK_GPP_G_0
19	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_G_19):** Applied to GPP_G19. Same description as PADCFGLOCK_GPP_G_0
18	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_G_18):** Applied to GPP_G18. Same description as PADCFGLOCK_GPP_G_0
17	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_G_17):** Applied to GPP_G17. Same description as PADCFGLOCK_GPP_G_0

Bit Range	Default & Access	Field Name (ID): Description
16	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_G_16):** Applied to GPP_G16. Same description as PADCFGLOCK_GPP_G_0
15	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_G_15):** Applied to GPP_G15. Same description as PADCFGLOCK_GPP_G_0
14	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_G_14):** Applied to GPP_G14. Same description as PADCFGLOCK_GPP_G_0
13	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_G_13):** Applied to GPP_G13. Same description as PADCFGLOCK_GPP_G_0
12	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_G_12):** Applied to GPP_G12. Same description as PADCFGLOCK_GPP_G_0
11	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_G_11):** Applied to GPP_G11. Same description as PADCFGLOCK_GPP_G_0
10	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_G_10):** Applied to GPP_G10. Same description as PADCFGLOCK_GPP_G_0
9	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_G_9):** Applied to GPP_G9. Same description as PADCFGLOCK_GPP_G_0
8	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_G_8):** Applied to GPP_G8. Same description as PADCFGLOCK_GPP_G_0
7	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_G_7):** Applied to GPP_G7. Same description as PADCFGLOCK_GPP_G_0
6	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_G_6):** Applied to GPP_G6. Same description as PADCFGLOCK_GPP_G_0
5	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_G_5):** Applied to GPP_G5. Same description as PADCFGLOCK_GPP_G_0
4	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_G_4):** Applied to GPP_G4. Same description as PADCFGLOCK_GPP_G_0
3	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_G_3):** Applied to GPP_G3. Same description as PADCFGLOCK_GPP_G_0
2	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_G_2):** Applied to GPP_G2. Same description as PADCFGLOCK_GPP_G_0
1	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_G_1):** Applied to GPP_G1. Same description as PADCFGLOCK_GPP_G_0
0	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_G_0):** Pad Configuration Lock locks specific register fields in the GPP specific registers from being configured. The registers affected become Read-Only and software writes to these registers have no effect. 0 = Unlock 1 = Lock the following register fields as read-only (RO): - Pad Configuration registers (exclude GPIOTXState) - GPI_NMI_EN Register (if implemented) - GPI_SMI_EN Register (if implemented) - GPI_GPE_EN Register (if implemented) When PadCfgLock is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

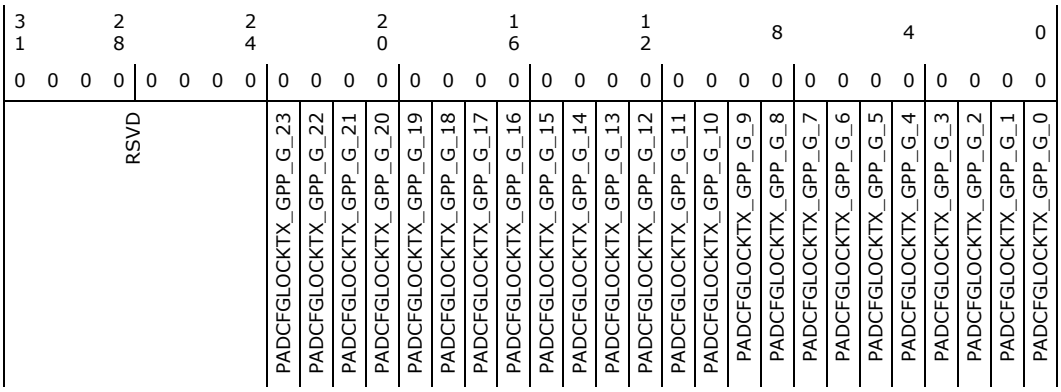
18.11.13 Pad Configuration Lock (PADCFGLOCKTX_GPP_G)—Offset 64h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_G_23):** Applied to GPP_G23. Same description as PADCFGLOCKTX_GPP_G_0
22	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_G_22):** Applied to GPP_G22. Same description as PADCFGLOCKTX_GPP_G_0
21	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_G_21):** Applied to GPP_G21. Same description as PADCFGLOCKTX_GPP_G_0
20	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_G_20):** Applied to GPP_G20. Same description as PADCFGLOCKTX_GPP_G_0
19	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_G_19):** Applied to GPP_G19. Same description as PADCFGLOCKTX_GPP_G_0
18	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_G_18):** Applied to GPP_G18. Same description as PADCFGLOCKTX_GPP_G_0
17	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_G_17):** Applied to GPP_G17. Same description as PADCFGLOCKTX_GPP_G_0
16	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_G_16):** Applied to GPP_G16. Same description as PADCFGLOCKTX_GPP_G_0
15	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_G_15):** Applied to GPP_G15. Same description as PADCFGLOCKTX_GPP_G_0
14	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_G_14):** Applied to GPP_G14. Same description as PADCFGLOCKTX_GPP_G_0
13	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_G_13):** Applied to GPP_G13. Same description as PADCFGLOCKTX_GPP_G_0
12	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_G_12):** Applied to GPP_G12. Same description as PADCFGLOCKTX_GPP_G_0

Bit Range	Default & Access	Field Name (ID): Description
11	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_G_11):** Applied to GPP_G11. Same description as PADCFGLOCKTX_GPP_G_0
10	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_G_10):** Applied to GPP_G10. Same description as PADCFGLOCKTX_GPP_G_0
9	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_G_9):** Applied to GPP_G9. Same description as PADCFGLOCKTX_GPP_G_0
8	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_G_8):** Applied to GPP_G8. Same description as PADCFGLOCKTX_GPP_G_0
7	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_G_7):** Applied to GPP_G7. Same description as PADCFGLOCKTX_GPP_G_0
6	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_G_6):** Applied to GPP_G6. Same description as PADCFGLOCKTX_GPP_G_0
5	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_G_5):** Applied to GPP_G5. Same description as PADCFGLOCKTX_GPP_G_0
4	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_G_4):** Applied to GPP_G4. Same description as PADCFGLOCKTX_GPP_G_0
3	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_G_3):** Applied to GPP_G3. Same description as PADCFGLOCKTX_GPP_G_0
2	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_G_2):** Applied to GPP_G2. Same description as PADCFGLOCKTX_GPP_G_0
1	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_G_1):** Applied to GPP_G1. Same description as PADCFGLOCKTX_GPP_G_0
0	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_G_0):** PadCfgLockTx locks the GPIOTxState bit from being configured. The GPIOTxState register becomes Read-Only and software writes to the register have no effect. 0 = Unlock 1 = Locks the Pad Configuration GPIOTXState field as read-only (RO) When PadCfgLockTx is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

18.11.14 Pad Configuration Lock (PADCFGLOCK_GPP_H)—Offset 68h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD								PADCFGLOCK_GPP_H_23	PADCFGLOCK_GPP_H_22	PADCFGLOCK_GPP_H_21	PADCFGLOCK_GPP_H_20	PADCFGLOCK_GPP_H_19	PADCFGLOCK_GPP_H_18	PADCFGLOCK_GPP_H_17	PADCFGLOCK_GPP_H_16	PADCFGLOCK_GPP_H_15	PADCFGLOCK_GPP_H_14	PADCFGLOCK_GPP_H_13	PADCFGLOCK_GPP_H_12	PADCFGLOCK_GPP_H_11	PADCFGLOCK_GPP_H_10	PADCFGLOCK_GPP_H_9	PADCFGLOCK_GPP_H_8	PADCFGLOCK_GPP_H_7	PADCFGLOCK_GPP_H_6	PADCFGLOCK_GPP_H_5	PADCFGLOCK_GPP_H_4	PADCFGLOCK_GPP_H_3	PADCFGLOCK_GPP_H_2	PADCFGLOCK_GPP_H_1	PADCFGLOCK_GPP_H_0

Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_H_23):** Applied to GPP_H23. Same description as PADCFGLOCK_GPP_H_0
22	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_H_22):** Applied to GPP_H22. Same description as PADCFGLOCK_GPP_H_0
21	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_H_21):** Applied to GPP_H21. Same description as PADCFGLOCK_GPP_H_0
20	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_H_20):** Applied to GPP_H20. Same description as PADCFGLOCK_GPP_H_0
19	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_H_19):** Applied to GPP_H19. Same description as PADCFGLOCK_GPP_H_0
18	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_H_18):** Applied to GPP_H18. Same description as PADCFGLOCK_GPP_H_0
17	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_H_17):** Applied to GPP_H17. Same description as PADCFGLOCK_GPP_H_0
16	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_H_16):** Applied to GPP_H16. Same description as PADCFGLOCK_GPP_H_0
15	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_H_15):** Applied to GPP_H15. Same description as PADCFGLOCK_GPP_H_0
14	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_H_14):** Applied to GPP_H14. Same description as PADCFGLOCK_GPP_H_0
13	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_H_13):** Applied to GPP_H13. Same description as PADCFGLOCK_GPP_H_0
12	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_H_12):** Applied to GPP_H12. Same description as PADCFGLOCK_GPP_H_0
11	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_H_11):** Applied to GPP_H11. Same description as PADCFGLOCK_GPP_H_0
10	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_H_10):** Applied to GPP_H10. Same description as PADCFGLOCK_GPP_H_0
9	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_H_9):** Applied to GPP_H9. Same description as PADCFGLOCK_GPP_H_0
8	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_H_8):** Applied to GPP_H8. Same description as PADCFGLOCK_GPP_H_0
7	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_H_7):** Applied to GPP_H7. Same description as PADCFGLOCK_GPP_H_0
6	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_H_6):** Applied to GPP_H6. Same description as PADCFGLOCK_GPP_H_0
5	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_H_5):** Applied to GPP_H5. Same description as PADCFGLOCK_GPP_H_0
4	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_H_4):** Applied to GPP_H4. Same description as PADCFGLOCK_GPP_H_0
3	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_H_3):** Applied to GPP_H3. Same description as PADCFGLOCK_GPP_H_0

Bit Range	Default & Access	Field Name (ID): Description
2	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_H_2):** Applied to GPP_H2. Same description as PADCFGLOCK_GPP_H_0
1	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_H_1):** Applied to GPP_H1. Same description as PADCFGLOCK_GPP_H_0
0	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_H_0):** Pad Configuration Lock locks specific register fields in the GPP specific registers from being configured. The registers affected become Read-Only and software writes to these registers have no effect. 0 = Unlock 1 = Lock the following register fields as read-only (RO): - Pad Configuration registers (exclude GPIOTXState) - GPI_NMI_EN Register (if implemented) - GPI_SMI_EN Register (if implemented) - GPI_GPE_EN Register (if implemented) When PadCfgLock is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

18.11.15 Pad Configuration Lock (PADCFGLOCKTX_GPP_H)—Offset 6Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD								PADCFGLOCKTX_GPP_H_23	PADCFGLOCKTX_GPP_H_22	PADCFGLOCKTX_GPP_H_21	PADCFGLOCKTX_GPP_H_20	PADCFGLOCKTX_GPP_H_19	PADCFGLOCKTX_GPP_H_18	PADCFGLOCKTX_GPP_H_17	PADCFGLOCKTX_GPP_H_16	PADCFGLOCKTX_GPP_H_15	PADCFGLOCKTX_GPP_H_14	PADCFGLOCKTX_GPP_H_13	PADCFGLOCKTX_GPP_H_12	PADCFGLOCKTX_GPP_H_11	PADCFGLOCKTX_GPP_H_10	PADCFGLOCKTX_GPP_H_9	PADCFGLOCKTX_GPP_H_8	PADCFGLOCKTX_GPP_H_7	PADCFGLOCKTX_GPP_H_6	PADCFGLOCKTX_GPP_H_5	PADCFGLOCKTX_GPP_H_4	PADCFGLOCKTX_GPP_H_3	PADCFGLOCKTX_GPP_H_2	PADCFGLOCKTX_GPP_H_1	PADCFGLOCKTX_GPP_H_0

Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_H_23):** Applied to GPP_H23. Same description as PADCFGLOCKTX_GPP_H_0
22	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_H_22):** Applied to GPP_H22. Same description as PADCFGLOCKTX_GPP_H_0
21	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_H_21):** Applied to GPP_H21. Same description as PADCFGLOCKTX_GPP_H_0
20	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_H_20):** Applied to GPP_H20. Same description as PADCFGLOCKTX_GPP_H_0
19	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_H_19):** Applied to GPP_H19. Same description as PADCFGLOCKTX_GPP_H_0

Bit Range	Default & Access	Field Name (ID): Description
18	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_H_18):** Applied to GPP_H18. Same description as PADCFGLOCKTX_GPP_H_0
17	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_H_17):** Applied to GPP_H17. Same description as PADCFGLOCKTX_GPP_H_0
16	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_H_16):** Applied to GPP_H16. Same description as PADCFGLOCKTX_GPP_H_0
15	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_H_15):** Applied to GPP_H15. Same description as PADCFGLOCKTX_GPP_H_0
14	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_H_14):** Applied to GPP_H14. Same description as PADCFGLOCKTX_GPP_H_0
13	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_H_13):** Applied to GPP_H13. Same description as PADCFGLOCKTX_GPP_H_0
12	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_H_12):** Applied to GPP_H12. Same description as PADCFGLOCKTX_GPP_H_0
11	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_H_11):** Applied to GPP_H11. Same description as PADCFGLOCKTX_GPP_H_0
10	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_H_10):** Applied to GPP_H10. Same description as PADCFGLOCKTX_GPP_H_0
9	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_H_9):** Applied to GPP_H9. Same description as PADCFGLOCKTX_GPP_H_0
8	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_H_8):** Applied to GPP_H8. Same description as PADCFGLOCKTX_GPP_H_0
7	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_H_7):** Applied to GPP_H7. Same description as PADCFGLOCKTX_GPP_H_0
6	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_H_6):** Applied to GPP_H6. Same description as PADCFGLOCKTX_GPP_H_0
5	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_H_5):** Applied to GPP_H5. Same description as PADCFGLOCKTX_GPP_H_0
4	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_H_4):** Applied to GPP_H4. Same description as PADCFGLOCKTX_GPP_H_0
3	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_H_3):** Applied to GPP_H3. Same description as PADCFGLOCKTX_GPP_H_0
2	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_H_2):** Applied to GPP_H2. Same description as PADCFGLOCKTX_GPP_H_0
1	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_H_1):** Applied to GPP_H1. Same description as PADCFGLOCKTX_GPP_H_0
0	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_H_0):** PadCfgLockTx locks the GPIOTxState bit from being configured. The GPIOTxState register becomes Read-Only and software writes to the register have no effect. 0 = Unlock 1 = Locks the Pad Configuration GPIOTXState field as read-only (RO) When PadCfgLockTx is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

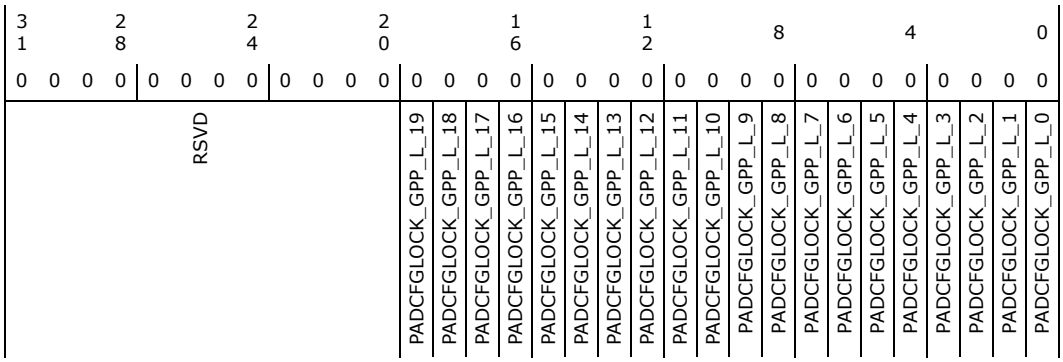
18.11.16 Pad Configuration Lock (PADCFGLOCK_GPP_L)—Offset 70h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:20	0h RO	Reserved
19	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_L_19):** Applied to GPP_L19. Same description as PADCFGLOCK_GPP_L_0
18	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_L_18):** Applied to GPP_L18. Same description as PADCFGLOCK_GPP_L_0
17	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_L_17):** Applied to GPP_L17. Same description as PADCFGLOCK_GPP_L_0
16	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_L_16):** Applied to GPP_L16. Same description as PADCFGLOCK_GPP_L_0
15	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_L_15):** Applied to GPP_L15. Same description as PADCFGLOCK_GPP_L_0
14	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_L_14):** Applied to GPP_L14. Same description as PADCFGLOCK_GPP_L_0
13	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_L_13):** Applied to GPP_L13. Same description as PADCFGLOCK_GPP_L_0
12	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_L_12):** Applied to GPP_L12. Same description as PADCFGLOCK_GPP_L_0
11	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_L_11):** Applied to GPP_L11. Same description as PADCFGLOCK_GPP_L_0
10	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_L_10):** Applied to GPP_L10. Same description as PADCFGLOCK_GPP_L_0
9	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_L_9):** Applied to GPP_G9. Same description as PADCFGLOCK_GPP_L_0
8	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_L_8):** Applied to GPP_L8. Same description as PADCFGLOCK_GPP_L_0
7	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_L_7):** Applied to GPP_L7. Same description as PADCFGLOCK_GPP_L_0

Bit Range	Default & Access	Field Name (ID): Description
6	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_L_6):** Applied to GPP_L6. Same description as PADCFGLOCK_GPP_L_0
5	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_L_5):** Applied to GPP_L5. Same description as PADCFGLOCK_GPP_L_0
4	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_L_4):** Applied to GPP_L4. Same description as PADCFGLOCK_GPP_L_0
3	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_L_3):** Applied to GPP_L3. Same description as PADCFGLOCK_GPP_L_0
2	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_L_2):** Applied to GPP_L2. Same description as PADCFGLOCK_GPP_L_0
1	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_L_1):** Applied to GPP_L1. Same description as PADCFGLOCK_GPP_L_0
0	0h RW	**Pad Config Lock (PADCFGLOCK_GPP_L_0):** Pad Configuration Lock locks specific register fields in the GPP specific registers from being configured. The registers affected become Read-Only and software writes to these registers have no effect. 0 = Unlock 1 = Lock the following register fields as read-only (RO): - Pad Configuration registers (exclude GPIOTXState) - GPI_NMI_EN Register (if implemented) - GPI_SMI_EN Register (if implemented) - GPI_GPE_EN Register (if implemented) When PadCfgLock is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

18.11.17 Pad Configuration Lock (PADCFGLOCKTX_GPP_L)—Offset 74h

Same description as PADCFGLOCKTX_GPP_G register, except that this register applies to GPP_L[23:0].

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD								PADCFGLOCKTX_GPP_L_19	PADCFGLOCKTX_GPP_L_18	PADCFGLOCKTX_GPP_L_17	PADCFGLOCKTX_GPP_L_16	PADCFGLOCKTX_GPP_L_15	PADCFGLOCKTX_GPP_L_14	PADCFGLOCKTX_GPP_L_13	PADCFGLOCKTX_GPP_L_12	PADCFGLOCKTX_GPP_L_11	PADCFGLOCKTX_GPP_L_10	PADCFGLOCKTX_GPP_L_9	PADCFGLOCKTX_GPP_L_8	PADCFGLOCKTX_GPP_L_7	PADCFGLOCKTX_GPP_L_6	PADCFGLOCKTX_GPP_L_5	PADCFGLOCKTX_GPP_L_4	PADCFGLOCKTX_GPP_L_3	PADCFGLOCKTX_GPP_L_2	PADCFGLOCKTX_GPP_L_1	PADCFGLOCKTX_GPP_L_0		

Bit Range	Default & Access	Field Name (ID): Description
31:20	0h RO	Reserved
19	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_L_19):** Applied to GPP_L19. Same description as PADCFGLOCKTX_GPP_L_0
18	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_L_18):** Applied to GPP_L18. Same description as PADCFGLOCKTX_GPP_L_0
17	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_L_17):** Applied to GPP_L17. Same description as PADCFGLOCKTX_GPP_L_0
16	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_L_16):** Applied to GPP_L16. Same description as PADCFGLOCKTX_GPP_L_0
15	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_L_15):** Applied to GPP_L15. Same description as PADCFGLOCKTX_GPP_L_0
14	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_L_14):** Applied to GPP_L14. Same description as PADCFGLOCKTX_GPP_L_0
13	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_L_13):** Applied to GPP_L13. Same description as PADCFGLOCKTX_GPP_L_0
12	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_L_12):** Applied to GPP_L12. Same description as PADCFGLOCKTX_GPP_L_0
11	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_L_11):** Applied to GPP_L11. Same description as PADCFGLOCKTX_GPP_L_0
10	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_L_10):** Applied to GPP_L10. Same description as PADCFGLOCKTX_GPP_L_0
9	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_L_9):** Applied to GPP_L9. Same description as PADCFGLOCKTX_GPP_L_0
8	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_L_8):** Applied to GPP_L8. Same description as PADCFGLOCKTX_GPP_L_0
7	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_L_7):** Applied to GPP_L7. Same description as PADCFGLOCKTX_GPP_L_0
6	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_L_6):** Applied to GPP_L6. Same description as PADCFGLOCKTX_GPP_L_0
5	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_L_5):** Applied to GPP_L5. Same description as PADCFGLOCKTX_GPP_L_0
4	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_L_4):** Applied to GPP_L4. Same description as PADCFGLOCKTX_GPP_L_0
3	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_L_3):** Applied to GPP_L3. Same description as PADCFGLOCKTX_GPP_L_0
2	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_L_2):** Applied to GPP_L2. Same description as PADCFGLOCKTX_GPP_L_0
1	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_L_1):** Applied to GPP_L1. Same description as PADCFGLOCKTX_GPP_L_0
0	0h RW	**Pad Config Lock TXState (PADCFGLOCKTX_GPP_L_0):** PadCfgLockTx locks the GPIOTxState bit from being configured. The GPIOTxState register becomes Read-Only and software writes to the register have no effect. 0 = Unlock 1 = Locks the Pad Configuration GPIOTXState field as read-only (RO) When PadCfgLockTx is written from a '1' to a '0' (unlock), a synchronous SMI# is generated if enabled. This ensures that only SMM code can change the above GPIO registers after the lock down.

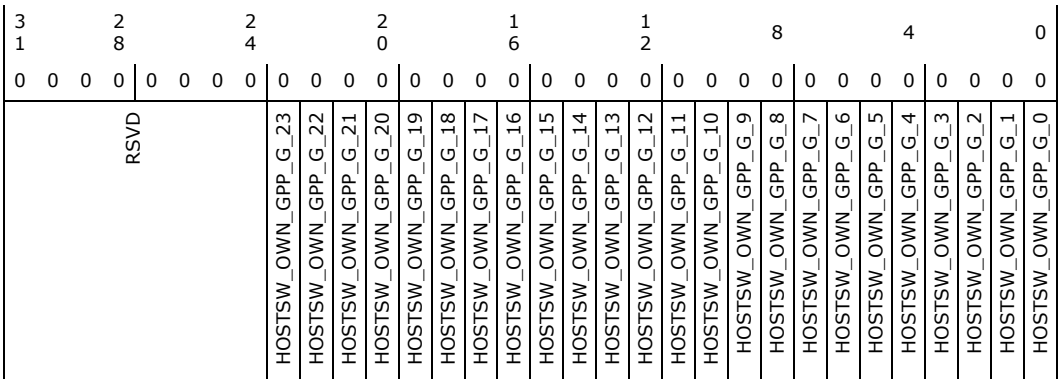
18.11.18 Host Software Pad Ownership (HOSTSW_OWN_GPP_G)—Offset 80h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_G_23):** Applied to GPP_G23. Same description as bit 0.
22	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_G_22):** Applied to GPP_G22. Same description as bit 0.
21	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_G_21):** Applied to GPP_G21. Same description as bit 0.
20	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_G_20):** Applied to GPP_G20. Same description as bit 0.
19	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_G_19):** Applied to GPP_G19. Same description as bit 0.
18	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_G_18):** Applied to GPP_G18. Same description as bit 0.
17	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_G_17):** Applied to GPP_G17. Same description as bit 0.
16	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_G_16):** Applied to GPP_G16. Same description as bit 0.
15	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_G_15):** Applied to GPP_G15. Same description as bit 0.
14	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_G_14):** Applied to GPP_G14. Same description as bit 0.
13	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_G_13):** Applied to GPP_G13. Same description as bit 0.
12	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_G_12):** Applied to GPP_G12. Same description as bit 0.
11	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_G_11):** Applied to GPP_G11. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
10	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_G_10):** Applied to GPP_G10. Same description as bit 0.
9	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_G_9):** Applied to GPP_G9. Same description as bit 0.
8	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_G_8):** Applied to GPP_G8. Same description as bit 0.
7	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_G_7):** Applied to GPP_G7. Same description as bit 0.
6	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_G_6):** Applied to GPP_G6. Same description as bit 0.
5	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_G_5):** Applied to GPP_G5. Same description as bit 0.
4	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_G_4):** Applied to GPP_G4. Same description as bit 0.
3	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_G_3):** Applied to GPP_G3. Same description as bit 0.
2	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_G_2):** Applied to GPP_G2. Same description as bit 0.
1	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_G_1):** Applied to GPP_G1. Same description as bit 0.
0	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_G_0):** This register determines the appropriate host status bit update when a pad is under host ownership. 0 = ACPI Mode. GPIO input event updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. GPI_STS update is masked. 1 = GPIO Driver Mode. GPIO input event updates are limited to GPI_STS. GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS updates are masked.

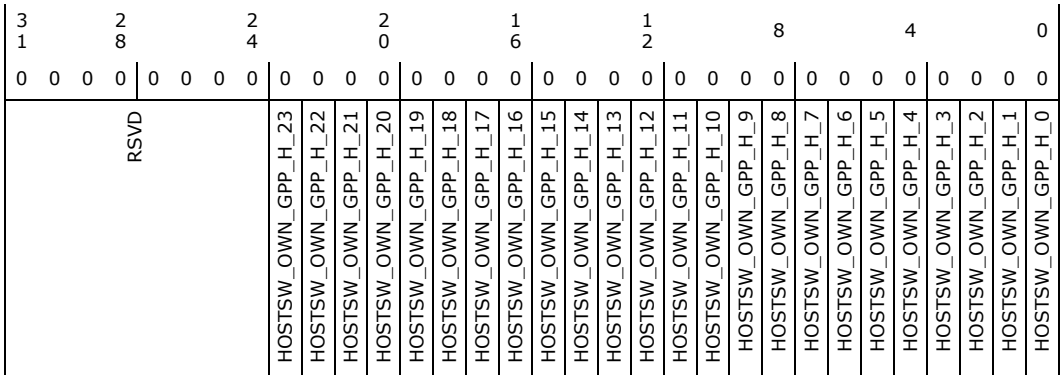
18.11.19 Host Software Pad Ownership (HOSTSW_OWN_GPP_H)—Offset 84h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_H_23):** Applied to GPP_H23. Same description as bit 0.
22	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_H_22):** Applied to GPP_H22. Same description as bit 0.
21	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_H_21):** Applied to GPP_H21. Same description as bit 0.
20	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_H_20):** Applied to GPP_H20. Same description as bit 0.
19	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_H_19):** Applied to GPP_H19. Same description as bit 0.
18	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_H_18):** Applied to GPP_H18. Same description as bit 0.
17	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_H_17):** Applied to GPP_H17. Same description as bit 0.
16	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_H_16):** Applied to GPP_H16. Same description as bit 0.
15	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_H_15):** Applied to GPP_H15. Same description as bit 0.
14	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_H_14):** Applied to GPP_H14. Same description as bit 0.
13	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_H_13):** Applied to GPP_H13. Same description as bit 0.
12	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_H_12):** Applied to GPP_H12. Same description as bit 0.
11	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_H_11):** Applied to GPP_H11. Same description as bit 0.
10	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_H_10):** Applied to GPP_H10. Same description as bit 0.
9	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_H_9):** Applied to GPP_H9. Same description as bit 0.
8	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_H_8):** Applied to GPP_H8. Same description as bit 0.
7	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_H_7):** Applied to GPP_H7. Same description as bit 0.
6	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_H_6):** Applied to GPP_H6. Same description as bit 0.
5	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_H_5):** Applied to GPP_H5. Same description as bit 0.
4	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_H_4):** Applied to GPP_H4. Same description as bit 0.
3	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_H_3):** Applied to GPP_H3. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
2	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_H_2):** Applied to GPP_H2. Same description as bit 0.
1	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_H_1):** Applied to GPP_H1. Same description as bit 0.
0	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_H_0):** This register determines the appropriate host status bit update when a pad is under host ownership. 0 = ACPI Mode. GPIO input event updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. GPI_STS update is masked. 1 = GPIO Driver Mode. GPIO input event updates are limited to GPI_STS. GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS updates are masked.

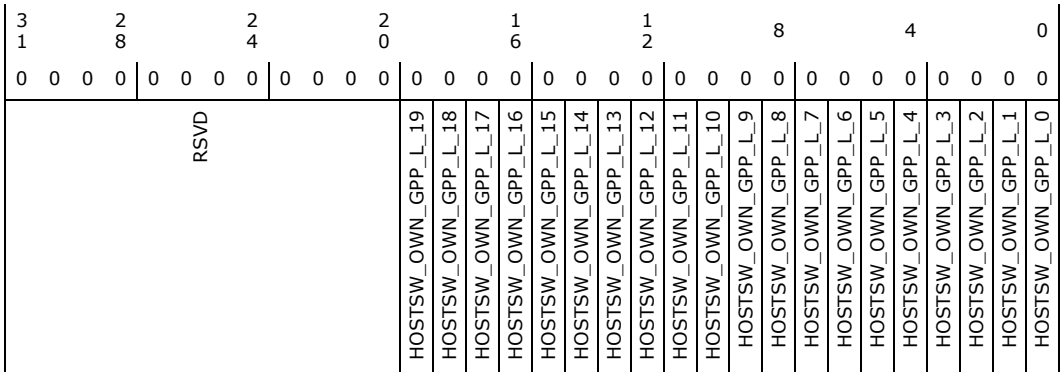
18.11.20 Host Software Pad Ownership (HOSTSW_OWN_GPP_L)—Offset 88h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:20	0h RO	Reserved
19	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_L_19):** Applied to GPP_L19. Same description as bit 0.
18	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_L_18):** Applied to GPP_L18. Same description as bit 0.
17	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_L_17):** Applied to GPP_L17. Same description as bit 0.
16	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_L_16):** Applied to GPP_L16. Same description as bit 0.
15	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_L_15):** Applied to GPP_L15. Same description as bit 0.
14	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_L_14):** Applied to GPP_L14. Same description as bit 0.
13	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_L_13):** Applied to GPP_L13. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
12	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_L_12):** Applied to GPP_L12. Same description as bit 0.
11	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_L_11):** Applied to GPP_L11. Same description as bit 0.
10	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_L_10):** Applied to GPP_L10. Same description as bit 0.
9	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_L_9):** Applied to GPP_L9. Same description as bit 0.
8	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_L_8):** Applied to GPP_L8. Same description as bit 0.
7	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_L_7):** Applied to GPP_L7. Same description as bit 0.
6	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_L_6):** Applied to GPP_L6. Same description as bit 0.
5	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_L_5):** Applied to GPP_L5. Same description as bit 0.
4	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_L_4):** Applied to GPP_L4. Same description as bit 0.
3	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_L_3):** Applied to GPP_L3. Same description as bit 0.
2	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_L_2):** Applied to GPP_L2. Same description as bit 0.
1	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_L_1):** Applied to GPP_L1. Same description as bit 0.
0	0h RW	**HostSW_Own (HOSTSW_OWN_GPP_L_0):** This register determines the appropriate host status bit update when a pad is under host ownership. 0 = ACPI Mode. GPIO input event updates are limited to GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS. GPI_STS update is masked. 1 = GPIO Driver Mode. GPIO input event updates are limited to GPI_STS. GPI_GPE_STS, GPI_NMI_STS and/or GPI_SMI_STS updates are masked.

18.11.21 GPI Interrupt Status (GPI_IS_GPP_G)—Offset 100h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	28	24	20	16	12	8	4	0
0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0
RSVD		GPI_INT_STS_GPP_G_23	GPI_INT_STS_GPP_G_22	GPI_INT_STS_GPP_G_21	GPI_INT_STS_GPP_G_20	GPI_INT_STS_GPP_G_19	GPI_INT_STS_GPP_G_18	GPI_INT_STS_GPP_G_17	GPI_INT_STS_GPP_G_16	GPI_INT_STS_GPP_G_15	GPI_INT_STS_GPP_G_14	GPI_INT_STS_GPP_G_13	GPI_INT_STS_GPP_G_12	GPI_INT_STS_GPP_G_11	GPI_INT_STS_GPP_G_10	GPI_INT_STS_GPP_G_9	GPI_INT_STS_GPP_G_8	GPI_INT_STS_GPP_G_7	GPI_INT_STS_GPP_G_6	GPI_INT_STS_GPP_G_5	GPI_INT_STS_GPP_G_4	GPI_INT_STS_GPP_G_3	GPI_INT_STS_GPP_G_2	GPI_INT_STS_GPP_G_1	GPI_INT_STS_GPP_G_0

Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_G_23):** Applied to GPP_G23. Same description as bit 0.
22	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_G_22):** Applied to GPP_G22. Same description as bit 0.
21	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_G_21):** Applied to GPP_G21. Same description as bit 0.
20	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_G_20):** Applied to GPP_G20. Same description as bit 0.
19	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_G_19):** Applied to GPP_G19. Same description as bit 0.
18	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_G_18):** Applied to GPP_G18. Same description as bit 0.
17	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_G_17):** Applied to GPP_G17. Same description as bit 0.
16	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_G_16):** Applied to GPP_G16. Same description as bit 0.
15	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_G_15):** Applied to GPP_G15. Same description as bit 0.
14	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_G_14):** Applied to GPP_G14. Same description as bit 0.
13	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_G_13):** Applied to GPP_G13. Same description as bit 0.
12	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_G_12):** Applied to GPP_G12. Same description as bit 0.
11	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_G_11):** Applied to GPP_G11. Same description as bit 0.
10	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_G_10):** Applied to GPP_G10. Same description as bit 0.
9	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_G_9):** Applied to GPP_G9. Same description as bit 0.
8	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_G_8):** Applied to GPP_G8. Same description as bit 0.
7	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_G_7):** Applied to GPP_G7. Same description as bit 0.
6	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_G_6):** Applied to GPP_G6. Same description as bit 0.
5	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_G_5):** Applied to GPP_G5. Same description as bit 0.
4	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_G_4):** Applied to GPP_G4. Same description as bit 0.
3	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_G_3):** Applied to GPP_G3. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
2	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_G_2):** Applied to GPP_G2. Same description as bit 0.
1	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_G_1):** Applied to GPP_G1. Same description as bit 0.
0	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_G_0):** GPI Interrupt Status (GPI_INT_STS) This bit is set to '1' by hardware when either an edge or a level event is detected (See RxEdCfg RxInv) on pad and all the following conditions are true: - The corresponding pad is used in GPIO input mode - HOSTSW_OWN = 1 (i.e. Host GPIO Driver Mode). Writing a value of 1 will clear the bit while writing a value of 0 has no effect. 0 = No interrupt 1 = Interrupt asserts The state of GPI_INT_EN[x] does not prevent the setting of GPI_INT_STS[x].

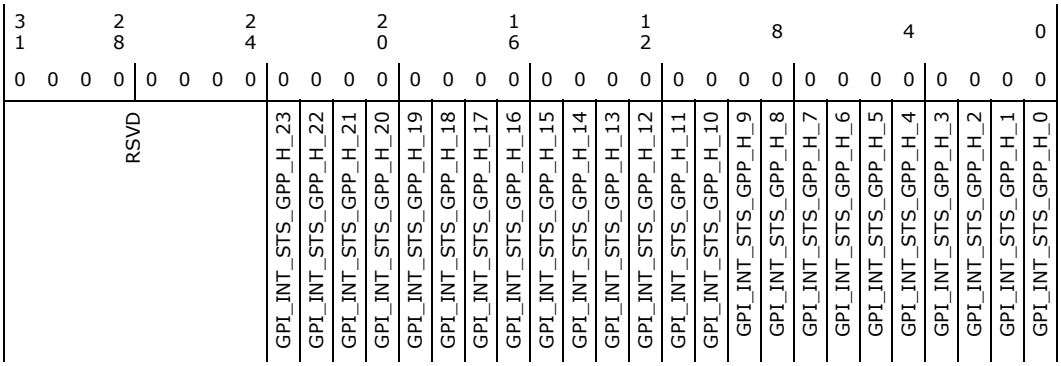
18.11.22 GPI Interrupt Status (GPI_IS_GPP_H)—Offset 104h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_H_23):** Applied to GPP_H23. Same description as bit 0.
22	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_H_22):** Applied to GPP_H22. Same description as bit 0.
21	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_H_21):** Applied to GPP_H21. Same description as bit 0.
20	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_H_20):** Applied to GPP_H20. Same description as bit 0.
19	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_H_19):** Applied to GPP_H19. Same description as bit 0.
18	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_H_18):** Applied to GPP_H18. Same description as bit 0.
17	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_H_17):** Applied to GPP_H17. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
16	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_H_16):** Applied to GPP_H16. Same description as bit 0.
15	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_H_15):** Applied to GPP_H15. Same description as bit 0.
14	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_H_14):** Applied to GPP_H14. Same description as bit 0.
13	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_H_13):** Applied to GPP_H13. Same description as bit 0.
12	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_H_12):** Applied to GPP_H12. Same description as bit 0.
11	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_H_11):** Applied to GPP_H11. Same description as bit 0.
10	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_H_10):** Applied to GPP_H10. Same description as bit 0.
9	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_H_9):** Applied to GPP_H9. Same description as bit 0.
8	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_H_8):** Applied to GPP_H8. Same description as bit 0.
7	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_H_7):** Applied to GPP_H7. Same description as bit 0.
6	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_H_6):** Applied to GPP_H6. Same description as bit 0.
5	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_H_5):** Applied to GPP_H5. Same description as bit 0.
4	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_H_4):** Applied to GPP_H4. Same description as bit 0.
3	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_H_3):** Applied to GPP_H3. Same description as bit 0.
2	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_H_2):** Applied to GPP_H2. Same description as bit 0.
1	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_H_1):** Applied to GPP_H1. Same description as bit 0.
0	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_H_0):** GPI Interrupt Status (GPI_INT_STS) This bit is set to '1' by hardware when either an edge or a level event is detected (See RxEdCfg RxInv) on pad and all the following conditions are true: - The corresponding pad is used in GPIO input mode - HOSTSW_OWN = 1 (i.e. Host GPIO Driver Mode). Writing a value of 1 will clear the bit while writing a value of 0 has no effect. 0 = No interrupt 1 = Interrupt asserts The state of GPI_INT_EN[x] does not prevent the setting of GPI_INT_STS[x].

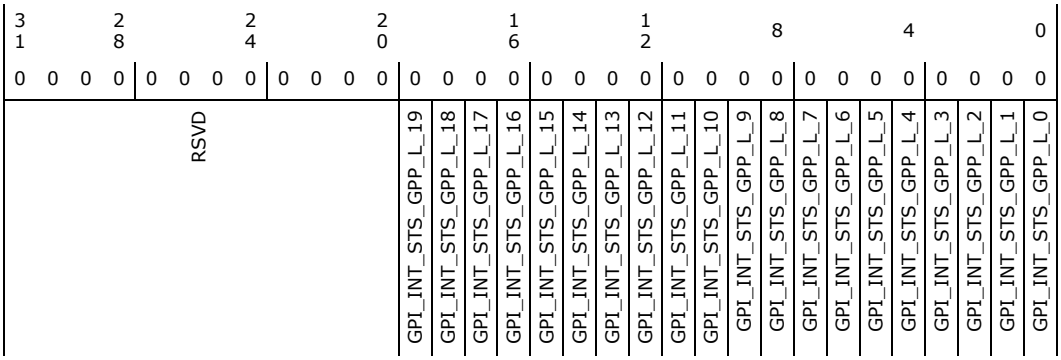
18.11.23 GPI Interrupt Status (GPI_IS_GPP_L)—Offset 108h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:20	0h RO	Reserved
19	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_L_19):** Applied to GPP_L19. Same description as bit 0.
18	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_L_18):** Applied to GPP_L18. Same description as bit 0.
17	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_L_17):** Applied to GPP_L17. Same description as bit 0.
16	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_L_16):** Applied to GPP_L16. Same description as bit 0.
15	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_L_15):** Applied to GPP_L15. Same description as bit 0.
14	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_L_14):** Applied to GPP_L14. Same description as bit 0.
13	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_L_13):** Applied to GPP_L13. Same description as bit 0.
12	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_L_12):** Applied to GPP_L12. Same description as bit 0.
11	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_L_11):** Applied to GPP_L11. Same description as bit 0.
10	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_L_10):** Applied to GPP_L10. Same description as bit 0.
9	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_L_9):** Applied to GPP_L9. Same description as bit 0.
8	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_L_8):** Applied to GPP_L8. Same description as bit 0.
7	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_L_7):** Applied to GPP_L7. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
6	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_L_6):** Applied to GPP_L6. Same description as bit 0.
5	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_L_5):** Applied to GPP_L5. Same description as bit 0.
4	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_L_4):** Applied to GPP_L4. Same description as bit 0.
3	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_L_3):** Applied to GPP_L3. Same description as bit 0.
2	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_L_2):** Applied to GPP_L2. Same description as bit 0.
1	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_L_1):** Applied to GPP_L1. Same description as bit 0.
0	0h RW1C	**GPI Interrupt Status (GPI_INT_STS_GPP_L_0):** GPI Interrupt Status (GPI_INT_STS) This bit is set to '1' by hardware when either an edge or a level event is detected (See RxEdCfg RxInv) on pad and all the following conditions are true: - The corresponding pad is used in GPIO input mode - HOSTSW_OWN = 1 (i.e. Host GPIO Driver Mode). Writing a value of 1 will clear the bit while writing a value of 0 has no effect. 0 = No interrupt 1 = Interrupt asserts The state of GPI_INT_EN[x] does not prevent the setting of GPI_INT_STS[x].

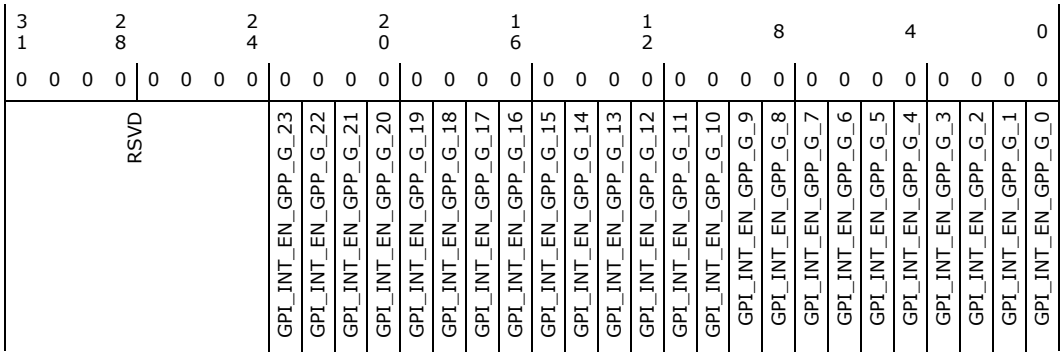
18.11.24 GPI Interrupt Enable (GPI_IE_GPP_G)—Offset 110h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_G_23):** Applied to GPP_G23. Same description as bit 0.
22	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_G_22):** Applied to GPP_G22. Same description as bit 0.
21	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_G_21):** Applied to GPP_G21. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
20	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_G_20):** Applied to GPP_G20. Same description as bit 0.
19	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_G_19):** Applied to GPP_G19. Same description as bit 0.
18	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_G_18):** Applied to GPP_G18. Same description as bit 0.
17	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_G_17):** Applied to GPP_G17. Same description as bit 0.
16	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_G_16):** Applied to GPP_G16. Same description as bit 0.
15	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_G_15):** Applied to GPP_G15. Same description as bit 0.
14	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_G_14):** Applied to GPP_G14. Same description as bit 0.
13	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_G_13):** Applied to GPP_G13. Same description as bit 0.
12	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_G_12):** Applied to GPP_G12. Same description as bit 0.
11	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_G_11):** Applied to GPP_G11. Same description as bit 0.
10	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_G_10):** Applied to GPP_G10. Same description as bit 0.
9	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_G_9):** Applied to GPP_G9. Same description as bit 0.
8	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_G_8):** Applied to GPP_G8. Same description as bit 0.
7	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_G_7):** Applied to GPP_G7. Same description as bit 0.
6	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_G_6):** Applied to GPP_G6. Same description as bit 0.
5	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_G_5):** Applied to GPP_G5. Same description as bit 0.
4	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_G_4):** Applied to GPP_G3. Same description as bit 0.
3	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_G_3):** Applied to GPP_G3. Same description as bit 0.
2	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_G_2):** Applied to GPP_G2. Same description as bit 0.
1	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_G_1):** Applied to GPP_G1. Same description as bit 0.
0	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_G_0):** This bit is used to enable/ disable the generation of APIC interrupt when the corresponding GPI_INT_STS bit is set. 0 = disable interrupt generation 1 = enable interrupt generation

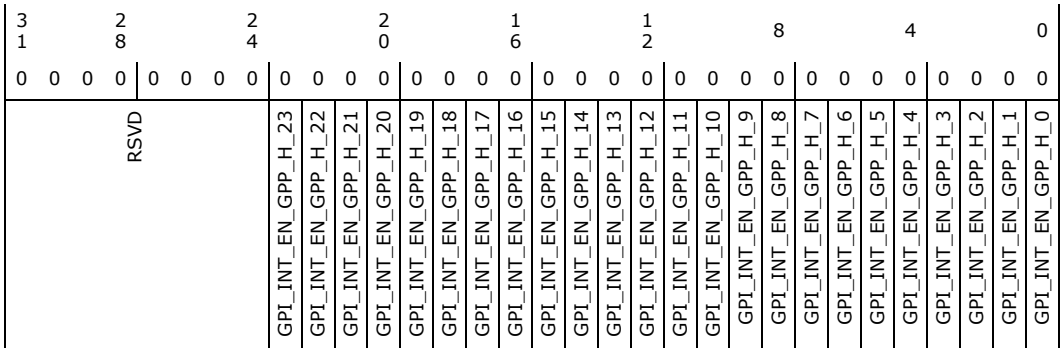
18.11.25 GPI Interrupt Enable (GPI_IE_GPP_H)—Offset 114h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved
23	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_H_23):** Applied to GPP_H23. Same description as bit 0.
22	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_H_22):** Applied to GPP_H22. Same description as bit 0.
21	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_H_21):** Applied to GPP_H21. Same description as bit 0.
20	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_H_20):** Applied to GPP_H20. Same description as bit 0.
19	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_H_19):** Applied to GPP_H19. Same description as bit 0.
18	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_H_18):** Applied to GPP_H18. Same description as bit 0.
17	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_H_17):** Applied to GPP_H17. Same description as bit 0.
16	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_H_16):** Applied to GPP_H16. Same description as bit 0.
15	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_H_15):** Applied to GPP_H15. Same description as bit 0.
14	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_H_14):** Applied to GPP_H14. Same description as bit 0.
13	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_H_13):** Applied to GPP_H13. Same description as bit 0.
12	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_H_12):** Applied to GPP_H12. Same description as bit 0.
11	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_H_11):** Applied to GPP_H11. Same description as bit 0.
10	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_H_10):** Applied to GPP_H10. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
9	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_H_9):** Applied to GPP_H9. Same description as bit 0.
8	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_H_8):** Applied to GPP_H8. Same description as bit 0.
7	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_H_7):** Applied to GPP_H7. Same description as bit 0.
6	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_H_6):** Applied to GPP_H6. Same description as bit 0.
5	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_H_5):** Applied to GPP_H5. Same description as bit 0.
4	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_H_4):** Applied to GPP_H3. Same description as bit 0.
3	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_H_3):** Applied to GPP_H3. Same description as bit 0.
2	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_H_2):** Applied to GPP_H2. Same description as bit 0.
1	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_H_1):** Applied to GPP_H1. Same description as bit 0.
0	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_H_0):** This bit is used to enable/ disable the generation of APIC interrupt when the corresponding GPI_INT_STS bit is set. 0 = disable interrupt generation 1 = enable interrupt generation

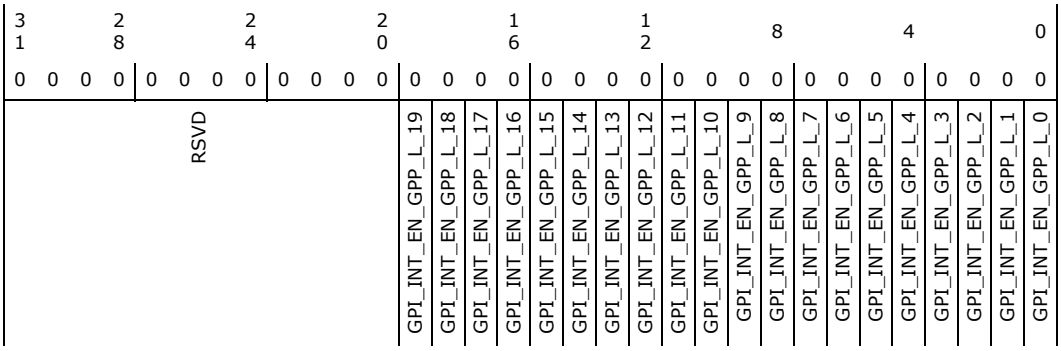
18.11.26 GPI Interrupt Enable (GPI_IE_GPP_L)—Offset 118h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:20	0h RO	Reserved.
19	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_L_19):** Applied to GPP_L19. Same description as bit 0.
18	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_L_18):** Applied to GPP_L18. Same description as bit 0.
17	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_L_17):** Applied to GPP_L17. Same description as bit 0.
16	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_L_16):** Applied to GPP_L16. Same description as bit 0.
15	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_L_15):** Applied to GPP_L15. Same description as bit 0.
14	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_L_14):** Applied to GPP_L14. Same description as bit 0.
13	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_L_13):** Applied to GPP_L13. Same description as bit 0.
12	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_L_12):** Applied to GPP_L12. Same description as bit 0.
11	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_L_11):** Applied to GPP_L11. Same description as bit 0.
10	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_L_10):** Applied to GPP_L10. Same description as bit 0.
9	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_L_9):** Applied to GPP_L9. Same description as bit 0.
8	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_L_8):** Applied to GPP_L8. Same description as bit 0.
7	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_L_7):** Applied to GPP_L7. Same description as bit 0.
6	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_L_6):** Applied to GPP_L6. Same description as bit 0.
5	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_L_5):** Applied to GPP_L5. Same description as bit 0.
4	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_L_4):** Applied to GPP_L3. Same description as bit 0.
3	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_L_3):** Applied to GPP_L3. Same description as bit 0.
2	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_L_2):** Applied to GPP_L2. Same description as bit 0.
1	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_L_1):** Applied to GPP_L1. Same description as bit 0.
0	0h RW	**GPI Interrupt Enable (GPI_INT_EN_GPP_L_0):** This bit is used to enable/ disable the generation of APIC interrupt when the corresponding GPI_INT_STS bit is set. 0 = disable interrupt generation 1 = enable interrupt generation

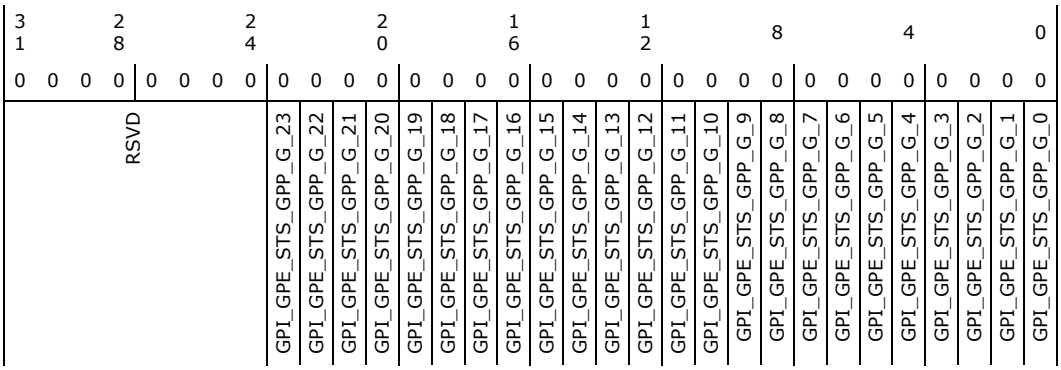
18.11.27 GPI General Purpose Events Status (GPI_GPE_STS_GPP_G)—Offset 120h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_G_23):** Applied to GPP_G23. Same description as bit 0.
22	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_G_22):** Applied to GPP_G22. Same description as bit 0.
21	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_G_21):** Applied to GPP_G21. Same description as bit 0.
20	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_G_20):** Applied to GPP_G20. Same description as bit 0.
19	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_G_19):** Applied to GPP_G19. Same description as bit 0.
18	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_G_18):** Applied to GPP_G18. Same description as bit 0.
17	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_G_17):** Applied to GPP_G17. Same description as bit 0.
16	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_G_16):** Applied to GPP_G16. Same description as bit 0.
15	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_G_15):** Applied to GPP_G15. Same description as bit 0.
14	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_G_14):** Applied to GPP_G14. Same description as bit 0.
13	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_G_13):** Applied to GPP_G13. Same description as bit 0.
12	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_G_12):** Applied to GPP_G12. Same description as bit 0.
11	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_G_11):** Applied to GPP_G11. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
10	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_G_10):** Applied to GPP_G10. Same description as bit 0.
9	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_G_9):** Applied to GPP_G9. Same description as bit 0.
8	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_G_8):** Applied to GPP_G8. Same description as bit 0.
7	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_G_7):** Applied to GPP_G7. Same description as bit 0.
6	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_G_6):** Applied to GPP_G6. Same description as bit 0.
5	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_G_5):** Applied to GPP_G5. Same description as bit 0.
4	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_G_4):** Applied to GPP_G4. Same description as bit 0.
3	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_G_3):** Applied to GPP_G3. Same description as bit 0.
2	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_G_2):** Applied to GPP_G2. Same description as bit 0.
1	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_G_1):** Applied to GPP_G1. Same description as bit 0.
0	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_G_0):** These bits are set any time the corresponding GPIO pad is set up as an input, under host ownership and the corresponding GPIO signal is high (or low if the corresponding RXINV bit is set). If the corresponding enable bit is set in the GPI_GPE_EN register, then when the GPI_GPE_STS bit is set: - If the system is in an S3-S5 state, the event will also wake the system. - If the system is in an S0 state (or upon waking back to an S0 state), an SCI will be caused, depending on the GPIRoutSCI bit for the corresponding pad. These bits are sticky bits and are cleared by writing a 1 back to this bit position. The state of GPI_GPE_EN does not prevent the setting of GPI_GPE_STS.

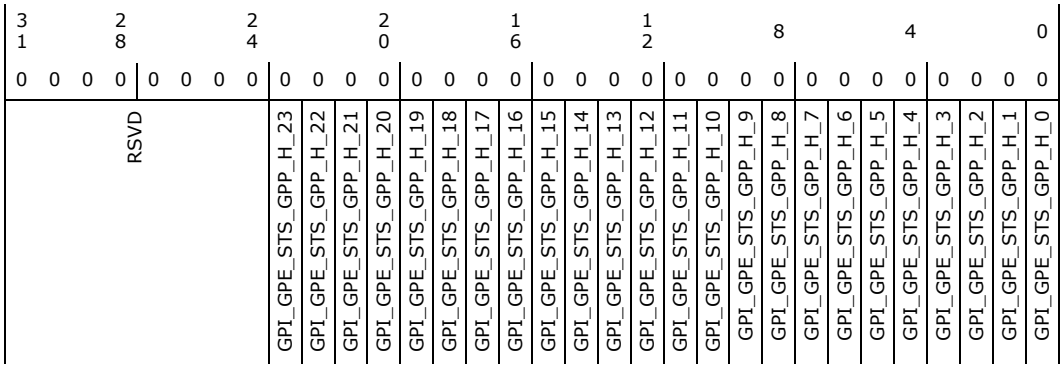
18.11.28 GPI General Purpose Events Status (GPI_GPE_STS_GPP_H)—Offset 124h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_H_23):** Applied to GPP_H23. Same description as bit 0.
22	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_H_22):** Applied to GPP_H22. Same description as bit 0.
21	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_H_21):** Applied to GPP_H21. Same description as bit 0.
20	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_H_20):** Applied to GPP_H20. Same description as bit 0.
19	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_H_19):** Applied to GPP_H19. Same description as bit 0.
18	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_H_18):** Applied to GPP_H18. Same description as bit 0.
17	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_H_17):** Applied to GPP_H17. Same description as bit 0.
16	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_H_16):** Applied to GPP_H16. Same description as bit 0.
15	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_H_15):** Applied to GPP_H15. Same description as bit 0.
14	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_H_14):** Applied to GPP_H14. Same description as bit 0.
13	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_H_13):** Applied to GPP_H13. Same description as bit 0.
12	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_H_12):** Applied to GPP_H12. Same description as bit 0.
11	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_H_11):** Applied to GPP_H11. Same description as bit 0.
10	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_H_10):** Applied to GPP_H10. Same description as bit 0.
9	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_H_9):** Applied to GPP_H9. Same description as bit 0.
8	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_H_8):** Applied to GPP_H8. Same description as bit 0.
7	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_H_7):** Applied to GPP_H7. Same description as bit 0.
6	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_H_6):** Applied to GPP_H6. Same description as bit 0.
5	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_H_5):** Applied to GPP_H5. Same description as bit 0.
4	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_H_4):** Applied to GPP_H4. Same description as bit 0.
3	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_H_3):** Applied to GPP_H3. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
2	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_H_2):** Applied to GPP_H2. Same description as bit 0.
1	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_H_1):** Applied to GPP_H1. Same description as bit 0.
0	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_H_0):** These bits are set any time the corresponding GPIO pad is set up as an input, under host ownership and the corresponding GPIO signal is high (or low if the corresponding RXINV bit is set). If the corresponding enable bit is set in the GPI_GPE_EN register, then when the GPI_GPE_STS bit is set: - If the system is in an S3-S5 state, the event will also wake the system. - If the system is in an S0 state (or upon waking back to an S0 state), an SCI will be caused, depending on the GPIRoutSCI bit for the corresponding pad. These bits are sticky bits and are cleared by writing a 1 back to this bit position. The state of GPI_GPE_EN does not prevent the setting of GPI_GPE_STS.

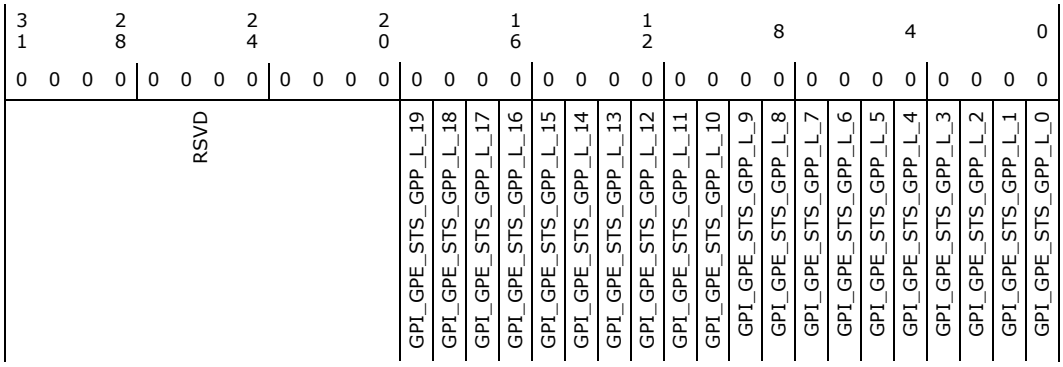
18.11.29 GPI General Purpose Events Status (GPI_GPE_STS_GPP_LGPP_L)—Offset 128h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:20	0h RO	Reserved.
19	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_L_19):** Applied to GPP_L19. Same description as bit 0.
18	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_L_18):** Applied to GPP_L18. Same description as bit 0.
17	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_L_17):** Applied to GPP_L17. Same description as bit 0.
16	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_L_16):** Applied to GPP_L16. Same description as bit 0.
15	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_L_15):** Applied to GPP_L15. Same description as bit 0.
14	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_L_14):** Applied to GPP_L14. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
13	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_L_13):** Applied to GPP_L13. Same description as bit 0.
12	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_L_12):** Applied to GPP_L12. Same description as bit 0.
11	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_L_11):** Applied to GPP_L11. Same description as bit 0.
10	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_L_10):** Applied to GPP_L10. Same description as bit 0.
9	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_L_9):** Applied to GPP_L9. Same description as bit 0.
8	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_L_8):** Applied to GPP_L8. Same description as bit 0.
7	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_L_7):** Applied to GPP_L7. Same description as bit 0.
6	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_L_6):** Applied to GPP_L6. Same description as bit 0.
5	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_L_5):** Applied to GPP_L5. Same description as bit 0.
4	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_L_4):** Applied to GPP_L4. Same description as bit 0.
3	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_L_3):** Applied to GPP_L3. Same description as bit 0.
2	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_L_2):** Applied to GPP_L2. Same description as bit 0.
1	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_L_1):** Applied to GPP_L1. Same description as bit 0.
0	0h RW1C	**GPI General Purpose Events Status (GPI_GPE_STS_GPP_L_0):** These bits are set any time the corresponding GPIO pad is set up as an input, under host ownership and the corresponding GPIO signal is high (or low if the corresponding RXINV bit is set). If the corresponding enable bit is set in the GPI_GPE_EN register, then when the GPI_GPE_STS bit is set: - If the system is in an S3-S5 state, the event will also wake the system. - If the system is in an S0 state (or upon waking back to an S0 state), an SCI will be caused, depending on the GPIRoutSCI bit for the corresponding pad. These bits are sticky bits and are cleared by writing a 1 back to this bit position. The state of GPI_GPE_EN does not prevent the setting of GPI_GPE_STS.

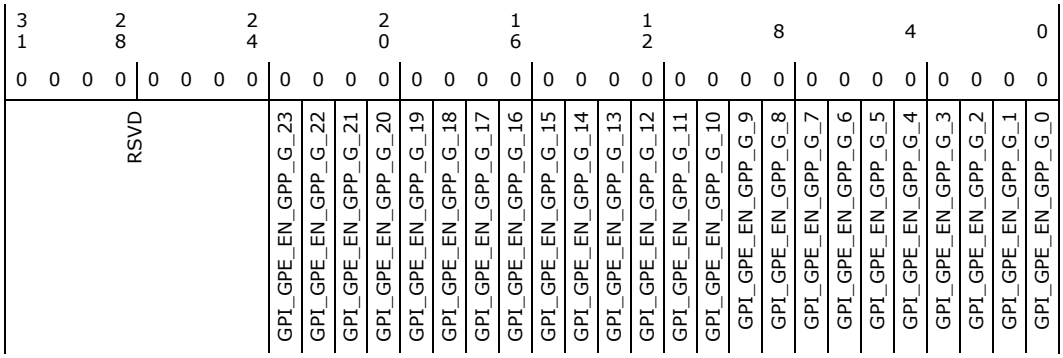
18.11.30 GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G)—Offset 130h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G_23):** Applied to GPP_G23. Same description as bit 0.
22	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G_22):** Applied to GPP_G22. Same description as bit 0.
21	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G_21):** Applied to GPP_G21. Same description as bit 0.
20	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G_20):** Applied to GPP_G20. Same description as bit 0.
19	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G_19):** Applied to GPP_G19. Same description as bit 0.
18	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G_18):** Applied to GPP_G18. Same description as bit 0.
17	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G_17):** Applied to GPP_G17. Same description as bit 0.
16	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G_16):** Applied to GPP_G16. Same description as bit 0.
15	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G_15):** Applied to GPP_G15. Same description as bit 0.
14	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G_14):** Applied to GPP_G14. Same description as bit 0.
13	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G_13):** Applied to GPP_G13. Same description as bit 0.
12	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G_12):** Applied to GPP_G12. Same description as bit 0.
11	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G_11):** Applied to GPP_G11. Same description as bit 0.
10	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G_10):** Applied to GPP_G10. Same description as bit 0.
9	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G_9):** Applied to GPP_G9. Same description as bit 0.
8	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G_8):** Applied to GPP_G8. Same description as bit 0.
7	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G_7):** Applied to GPP_G7. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
6	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G_6):** Applied to GPP_G6. Same description as bit 0.
5	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G_5):** Applied to GPP_G5. Same description as bit 0.
4	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G_4):** Applied to GPP_G4. Same description as bit 0.
3	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G_3):** Applied to GPP_G3. Same description as bit 0.
2	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G_2):** Applied to GPP_G2. Same description as bit 0.
1	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G_1):** Applied to GPP_G1. Same description as bit 0.
0	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_G_0):** This bit is used to enable/disable the generation of GPE to cause SCI and/or wake when the corresponding GPI_GPE_STS bit is set. 0 = disable GPE generation 1 = enable GPE generation ***Note:*** The pad must also be routed for GPE functionality in order for GPE to be generated, i.e. the corresponding GPIRoutSCI must be set to '1'.

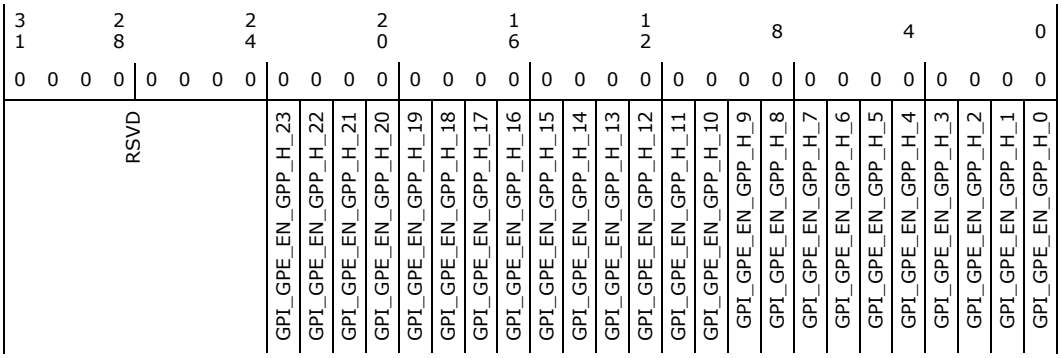
18.11.31 GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H)—Offset 134h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H_23):** Applied to GPP_H23. Same description as bit 0.
22	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H_22):** Applied to GPP_H22. Same description as bit 0.
21	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H_21):** Applied to GPP_H21. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
20	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H_20):** Applied to GPP_H20. Same description as bit 0.
19	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H_19):** Applied to GPP_H19. Same description as bit 0.
18	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H_18):** Applied to GPP_H18. Same description as bit 0.
17	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H_17):** Applied to GPP_H17. Same description as bit 0.
16	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H_16):** Applied to GPP_H16. Same description as bit 0.
15	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H_15):** Applied to GPP_H15. Same description as bit 0.
14	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H_14):** Applied to GPP_H14. Same description as bit 0.
13	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H_13):** Applied to GPP_H13. Same description as bit 0.
12	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H_12):** Applied to GPP_H12. Same description as bit 0.
11	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H_11):** Applied to GPP_H11. Same description as bit 0.
10	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H_10):** Applied to GPP_H10. Same description as bit 0.
9	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H_9):** Applied to GPP_H9. Same description as bit 0.
8	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H_8):** Applied to GPP_H8. Same description as bit 0.
7	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H_7):** Applied to GPP_H7. Same description as bit 0.
6	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H_6):** Applied to GPP_HGPP_H6. Same description as bit 0.
5	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H_5):** Applied to GPP_H5. Same description as bit 0.
4	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H_4):** Applied to GPP_H4. Same description as bit 0.
3	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H_3):** Applied to GPP_H3. Same description as bit 0.
2	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H_2):** Applied to GPP_H2. Same description as bit 0.
1	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H_1):** Applied to GPP_H1. Same description as bit 0.
0	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_H_0):** This bit is used to enable/disable the generation of GPE to cause SCI and/or wake when the corresponding GPI_GPE_STS bit is set. 0 = disable GPE generation 1 = enable GPE generation ***Note:*** The pad must also be routed for GPE functionality in order for GPE to be generated, i.e. the corresponding GPIRoutSCI must be set to '1'.

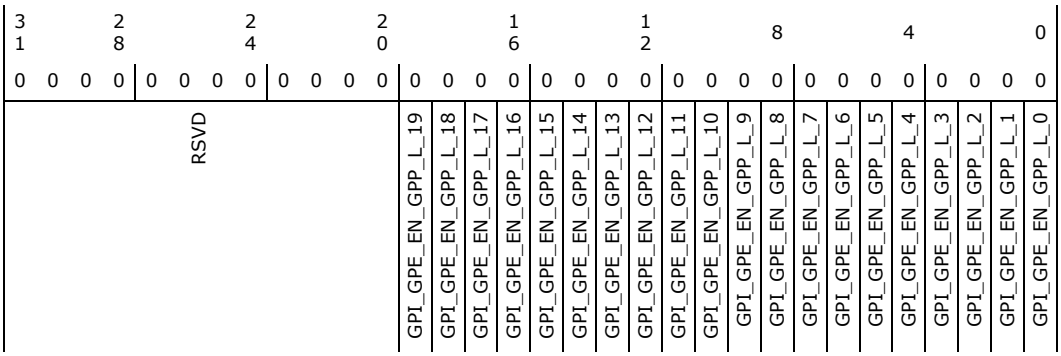
18.11.32 GPI General Purpose Events Enable (GPI_GPE_EN_GPP_L)—Offset 138h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default & Access	Field Name (ID): Description
31:24	0h RO	Reserved.
19	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_L_19):** Applied to GPP_L19. Same description as bit 0.
18	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_L_18):** Applied to GPP_L18. Same description as bit 0.
17	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_L_17):** Applied to GPP_L17. Same description as bit 0.
16	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_L_16):** Applied to GPP_L16. Same description as bit 0.
15	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_L_15):** Applied to GPP_L15. Same description as bit 0.
14	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_L_14):** Applied to GPP_L14. Same description as bit 0.
13	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_L_13):** Applied to GPP_L13. Same description as bit 0.
12	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_L_12):** Applied to GPP_L12. Same description as bit 0.
11	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_L_11):** Applied to GPP_L11. Same description as bit 0.
10	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_L_10):** Applied to GPP_L10. Same description as bit 0.
9	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_L_9):** Applied to GPP_L9. Same description as bit 0.
8	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_L_8):** Applied to GPP_L8. Same description as bit 0.
7	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_L_7):** Applied to GPP_L7. Same description as bit 0.

Bit Range	Default & Access	Field Name (ID): Description
6	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_L_6):** Applied to GPP_LGPP_L6. Same description as bit 0.
5	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_L_5):** Applied to GPP_L5. Same description as bit 0.
4	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_L_4):** Applied to GPP_L4. Same description as bit 0.
3	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_L_3):** Applied to GPP_L3. Same description as bit 0.
2	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_L_2):** Applied to GPP_L2. Same description as bit 0.
1	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_L_1):** Applied to GPP_L1. Same description as bit 0.
0	0h RW	**GPI General Purpose Events Enable (GPI_GPE_EN_GPP_L_0):** This bit is used to enable/disable the generation of GPE to cause SCI and/or wake when the corresponding GPI_GPE_STS bit is set. 0 = disable GPE generation 1 = enable GPE generation ***Note:*** The pad must also be routed for GPE functionality in order for GPE to be generated, i.e. the corresponding GPIRoutSCI must be set to '1'.

18.11.33 Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_0)—Offset 400h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG		RXPADSTSEL	RXRAW1	RSVD		RXEVCFG	PreGfRXSel	RXINV	RSVD		GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD				PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD							GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)

Bit Range	Default & Access	Field Name (ID): Description
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

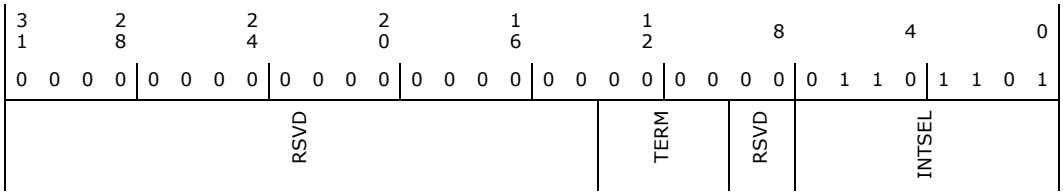
18.11.34 Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_0)—Offset 404h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 6Dh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	6Dh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

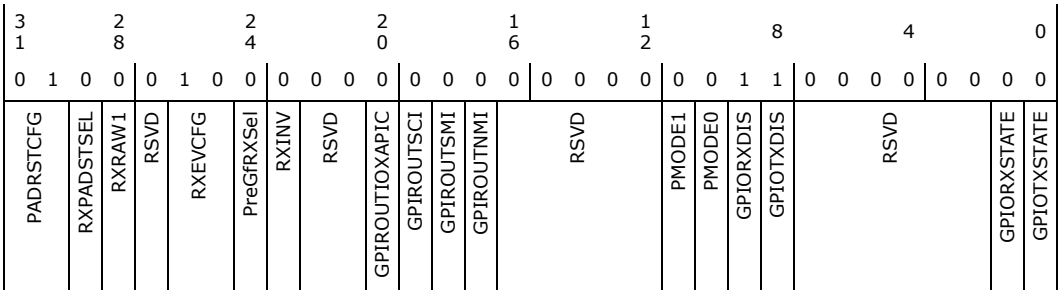
18.11.35 Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_1)—Offset 408h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

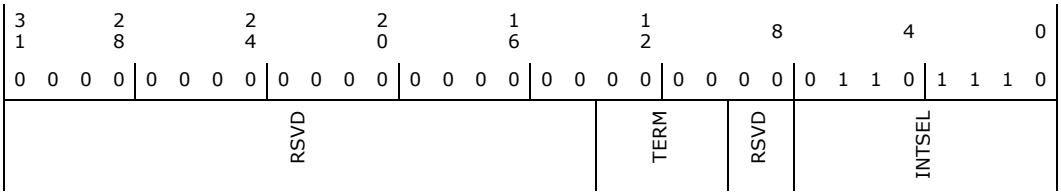
18.11.36 Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_1)—Offset 40Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 6Eh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	6Eh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.37 Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_2)—Offset 410h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG		RXPADSTSEL	RXRAW1	RSVD		RXEVCFG		PreGfRXSel	RXINV	RSVD		GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD				PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

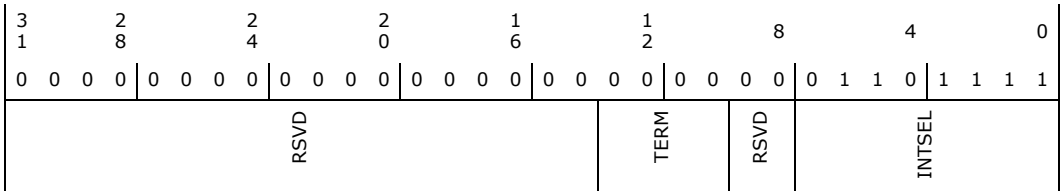
18.11.38 Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_2)—Offset 414h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 6Fh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	6Fh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.39 Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_3)—Offset 418h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG		RXPADSTSEL		RXRAW1	RSVD		RXEVCFG		PreGfRXSel	RXINV		RSVD		GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD				PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD				GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

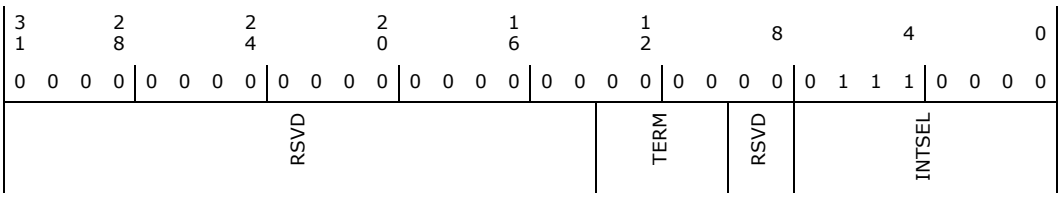
18.11.40 Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_3)—Offset 41Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 70h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	70h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.41 Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_4)—Offset 420h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG		RXPADSTSEL		RXRAW1		RSVD		RXEVCFG		PreGfRXSel		RXINV		RSVD		GPIROUTIOXAPIC		GPIROUTSCI		GPIROUTSMI		GPIROUTNMI		RSVD		PMODE1		PMODE0		GPIORXDIS		GPIOTXDIS		RSVD		GPIORXSTATE		GPIOTXSTATE	

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

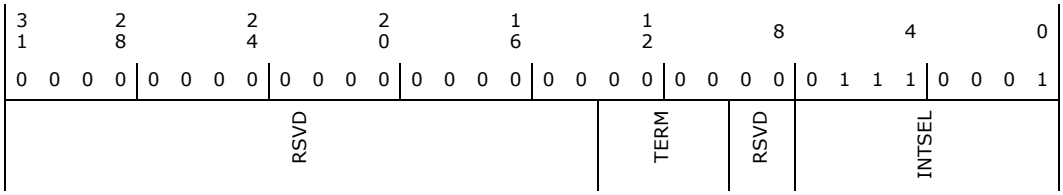
18.11.42 Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_4)—Offset 424h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 71h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	71h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

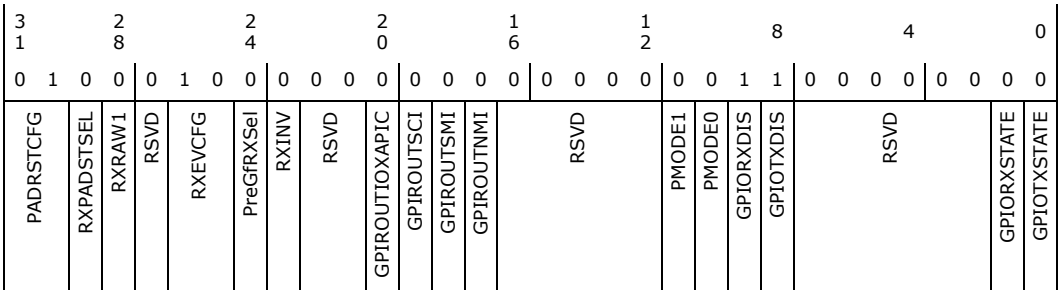
18.11.43 Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_5)—Offset 428h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

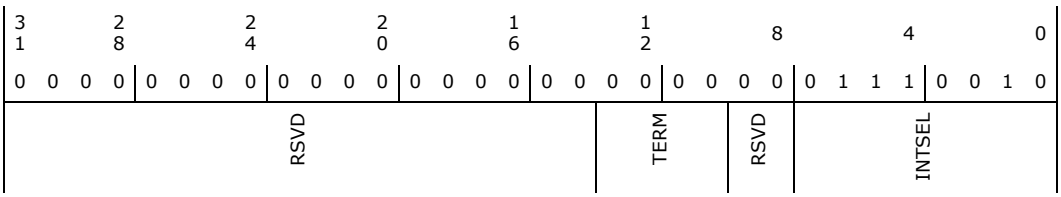
18.11.44 Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_5)—Offset 42Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 72h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	72h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.45 Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_6)—Offset 430h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG		RXPADSTSEL		RXRAW1		RSVD		RXEVCFG		PreGfRXSel		RXINV		RSVD		GPIROUTIOXAPIC		GPIROUTSCI		GPIROUTSMI		GPIROUTNMI		RSVD		PMODE1		PMODE0		GPIORXDIS		GPIOTXDIS		RSVD		GPIORXSTATE		GPIOTXSTATE	

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

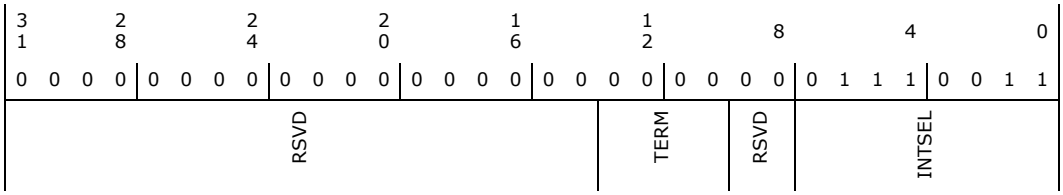
18.11.46 Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_6)—Offset 434h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 73h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	73h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

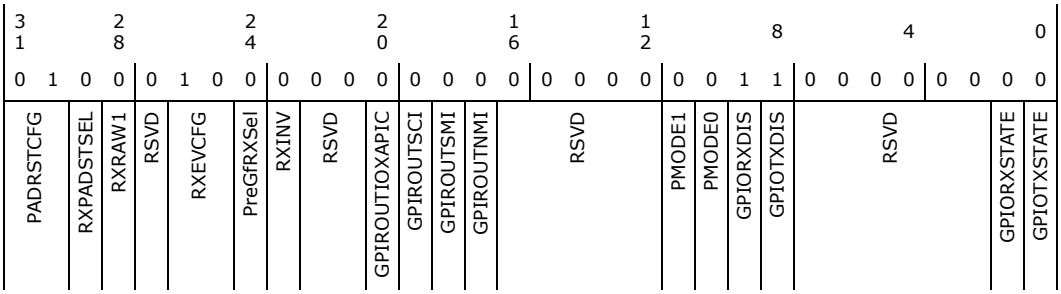
18.11.47 Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_7)—Offset 438h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

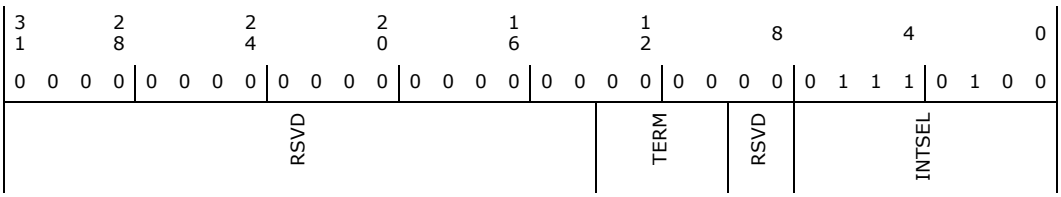
18.11.48 Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_7)—Offset 43Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 74h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	74h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.49 Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_8)—Offset 440h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

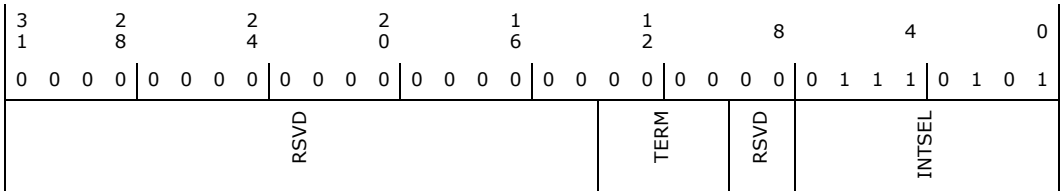
18.11.50 Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_8)—Offset 444h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 75h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	75h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

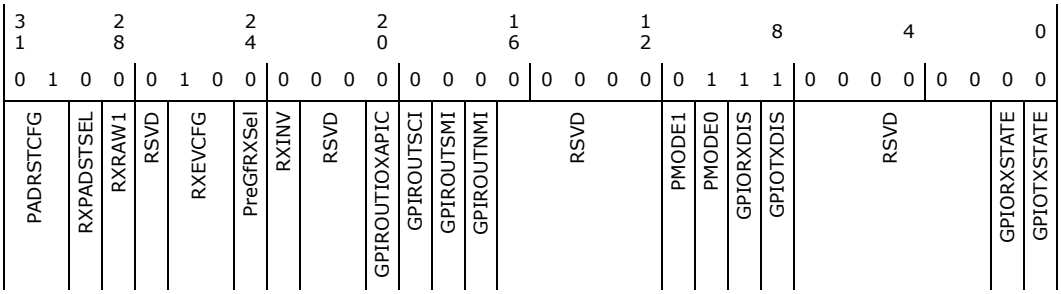
18.11.51 Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_9)—Offset 448h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

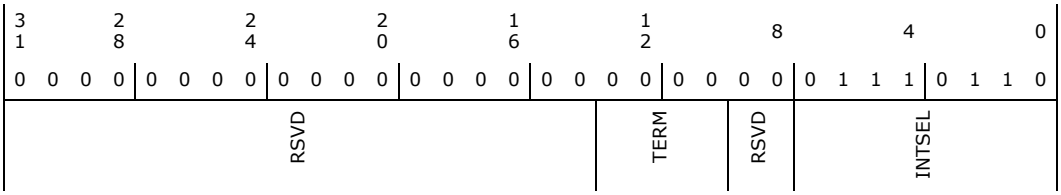
18.11.52 Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_9)—Offset 44Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 76h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	76h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.53 Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_10)—Offset 450h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

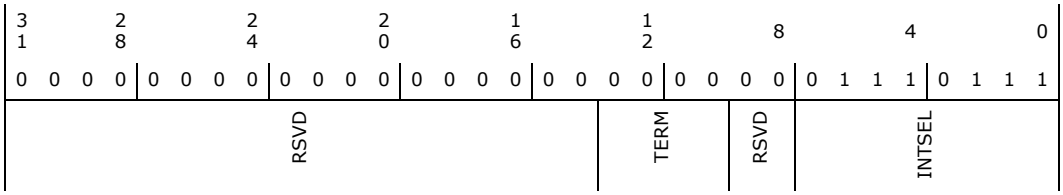
18.11.54 Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_10)—Offset 454h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 77h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	77h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

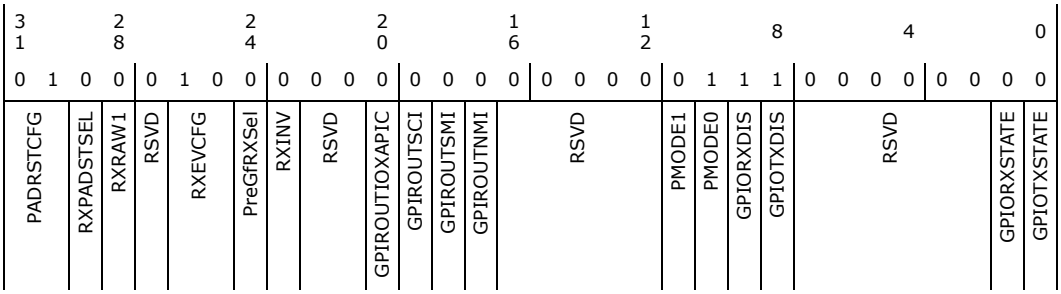
18.11.55 Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_11)—Offset 458h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

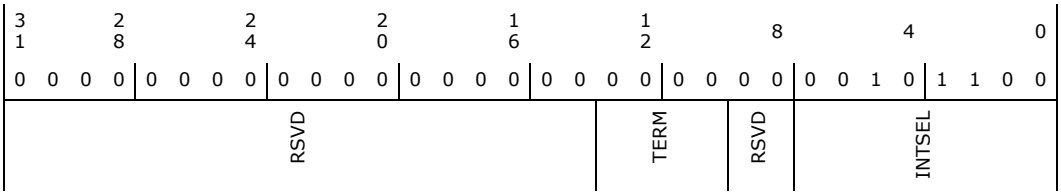
18.11.56 Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_11)—Offset 45Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 2Ch



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	2Ch RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.57 Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_12)—Offset 460h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

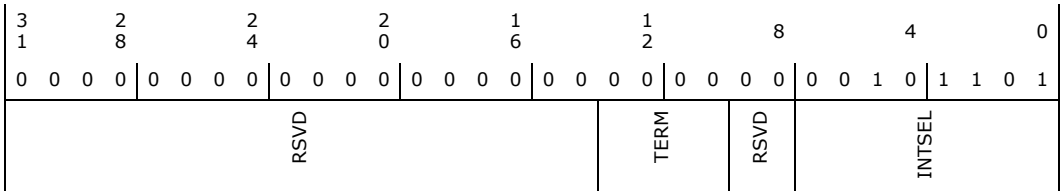
18.11.58 Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_12)—Offset 464h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 2D



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	2Dh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

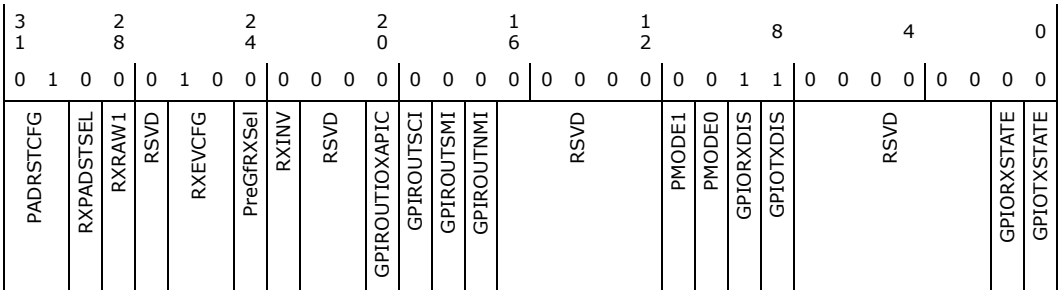
18.11.59 Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_13)—Offset 468h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

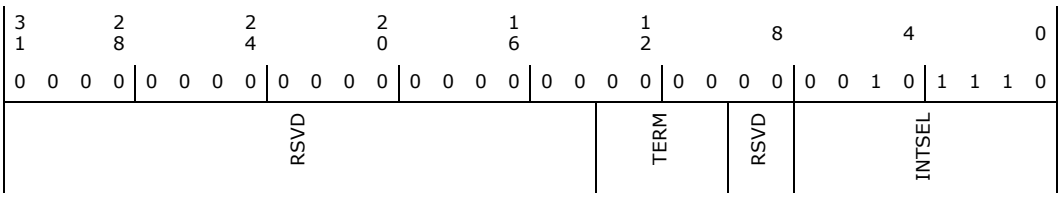
18.11.60 Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_13)—Offset 46Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 2Eh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	2Eh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.61 Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_14)—Offset 470h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

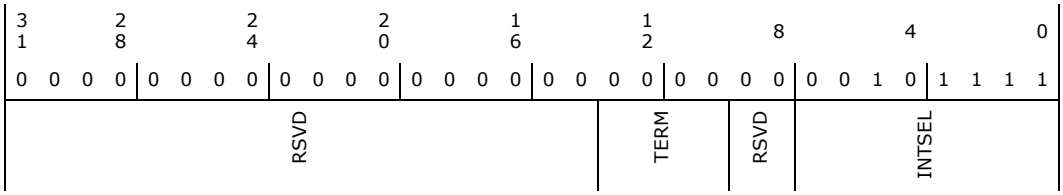
18.11.62 Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_14)—Offset 474h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 2Fh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	2Fh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.63 Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_15)—Offset 478h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG		RXPADSTSEL	RXRAW1	RSVD		RXEVCFG	PreGfRXSel	RXINV	RSVD		GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD							PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD					GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

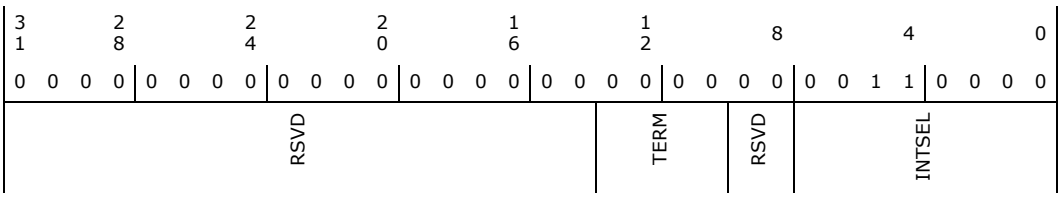
18.11.64 Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_15)—Offset 47Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 30h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	30h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.65 Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_16)—Offset 480h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG		RXPADSTSEL		RXRAW1		RSVD		RXEVCFG		PreGfRXSel		RXINV		RSVD		GPIROUTIOXAPIC		GPIROUTSCI		GPIROUTSMI		GPIROUTNMI		RSVD		PMODE1		PMODE0		GPIORXDIS		GPIOTXDIS		RSVD		GPIORXSTATE		GPIOTXSTATE	

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

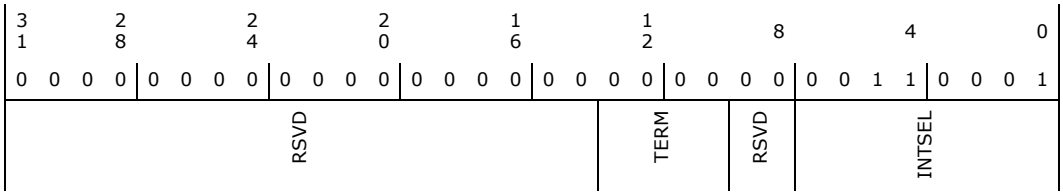
18.11.66 Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_16)—Offset 484h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 31h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	31h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

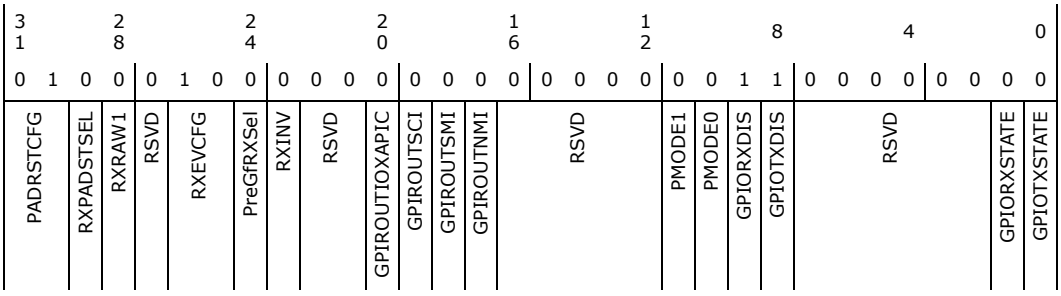
18.11.67 Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_17)—Offset 488h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

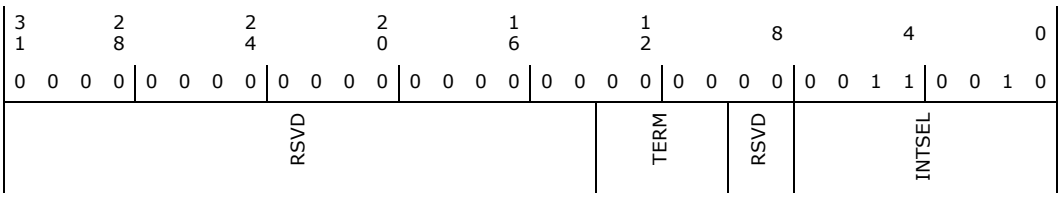
18.11.68 Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_17)—Offset 48Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 32h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	32h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.69 Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_18)—Offset 490h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG		RXPADSTSEL		RXRAW1		RSVD		RXEVCFG		PreGfRXSel		RXINV		RSVD		GPIROUTIOXAPIC		GPIROUTSCI		GPIROUTSMI		GPIROUTNMI		RSVD		PMODE1		PMODE0		GPIORXDIS		GPIOTXDIS		RSVD		GPIORXSTATE		GPIOTXSTATE	

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

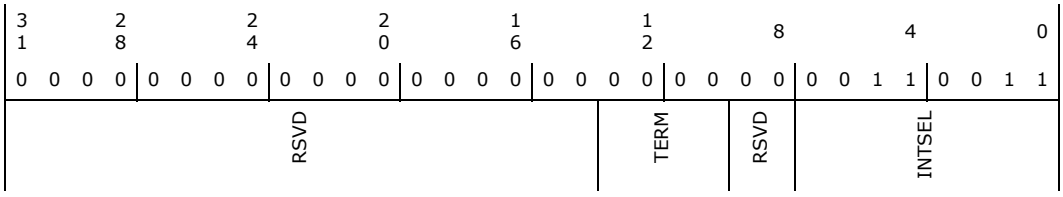
18.11.70 Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_18)—Offset 494h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 33h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	33h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

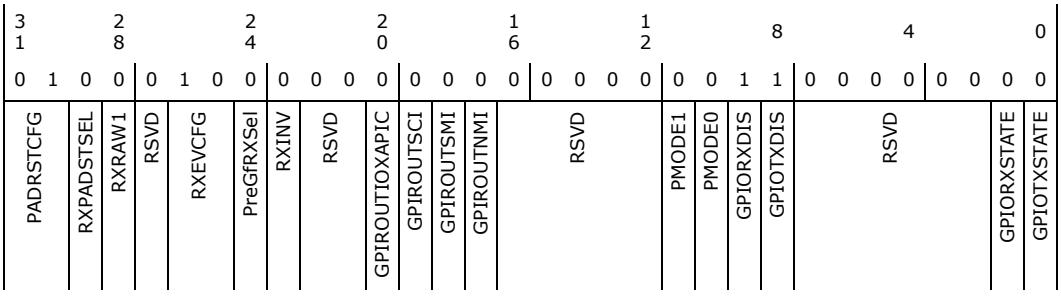
18.11.71 Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_19)—Offset 498h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

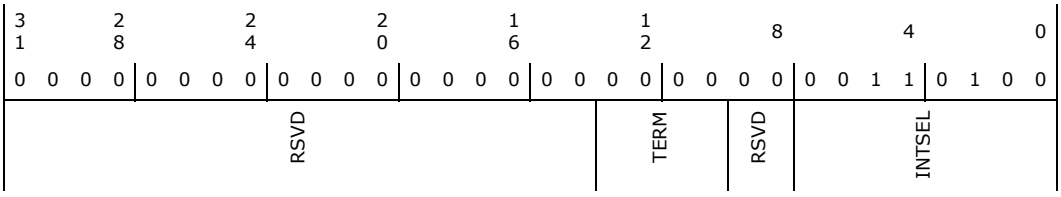
18.11.72 Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_19)—Offset 49Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 34h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	34h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.73 Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_20)—Offset 4A0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

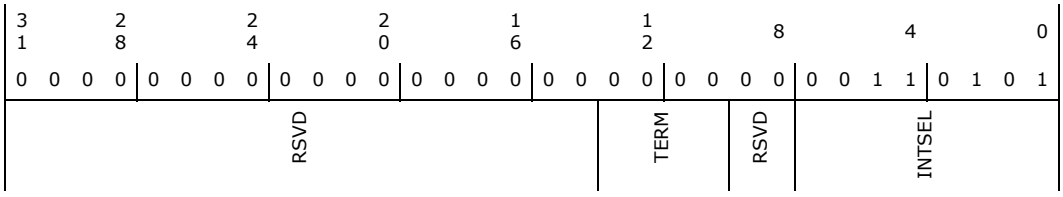
18.11.74 Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_20)—Offset 4A4h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 35h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	35h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

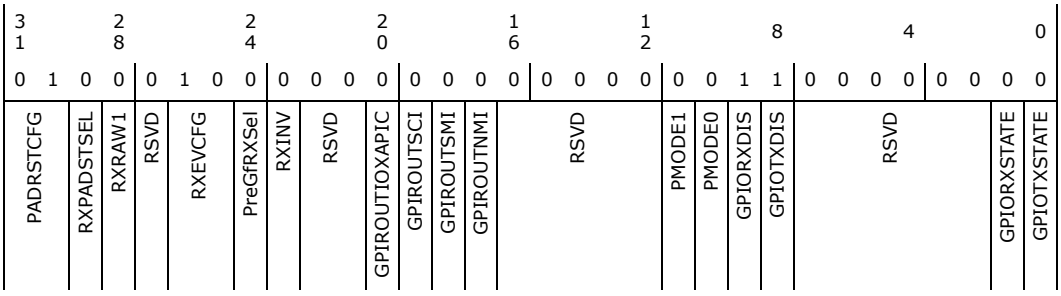
18.11.75 Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_21)—Offset 4A8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

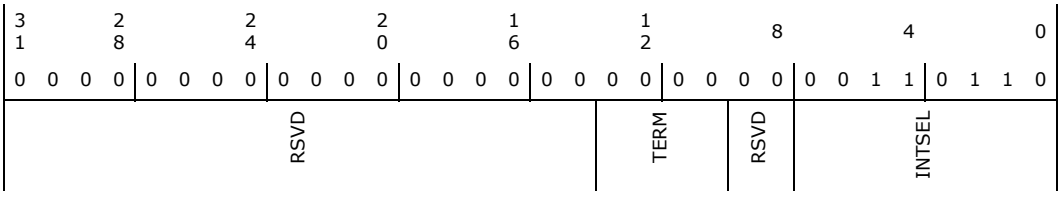
18.11.76 Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_21)—Offset 4ACh

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 36h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	36h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.77 Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_22)—Offset 4B0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG		RXPADSTSEL		RXRAW1		RSVD		RXEVCFG		PreGfRXSel		RXINV		RSVD		GPIROUTIOXAPIC		GPIROUTSCI		GPIROUTSMI		GPIROUTNMI		RSVD		PMODE1		PMODE0		GPIORXDIS		GPIOTXDIS		RSVD		GPIORXSTATE		GPIOTXSTATE	

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

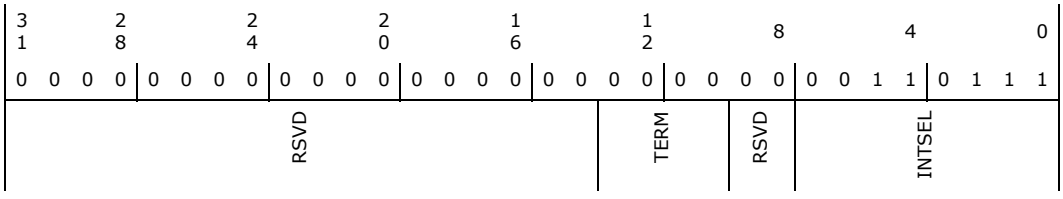
18.11.78 Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_22)—Offset 4B4h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 37h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	37h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.79 Pad Configuration DW0 (PAD_CFG_DW0_GPP_G_23)—Offset 4B8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG		RXPADSTSEL	RXRAW1	RSVD		RXEVCFG	PreGfRXSel	RXINV	RSVD		GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD							PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD					GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

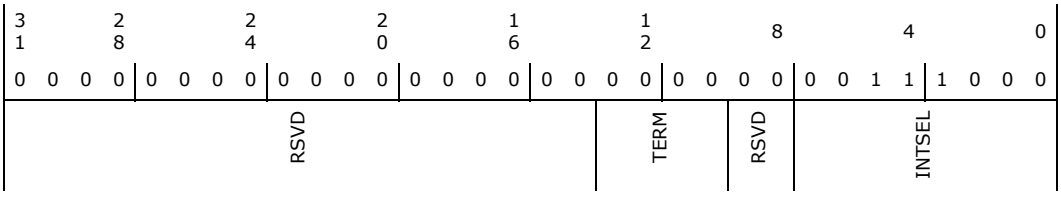
18.11.80 Pad Configuration DW1 (PAD_CFG_DW1_GPP_G_23)—Offset 4BCh

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 38h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	38h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.81 Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_0)—Offset 4C0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

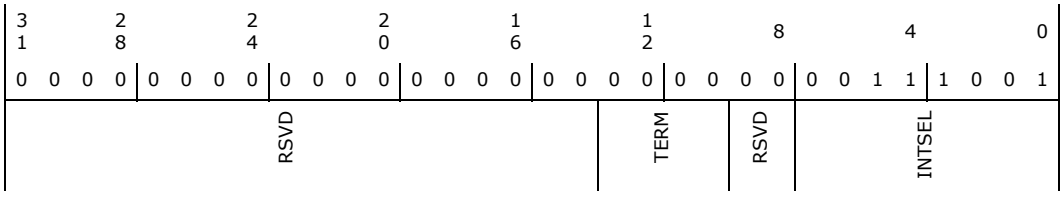
18.11.82 Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_0)—Offset 4C4h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 39h



Bit Range	Default & Access	Field Name (ID): Description
31:14	00000h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
13:10	0h RO	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	39h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

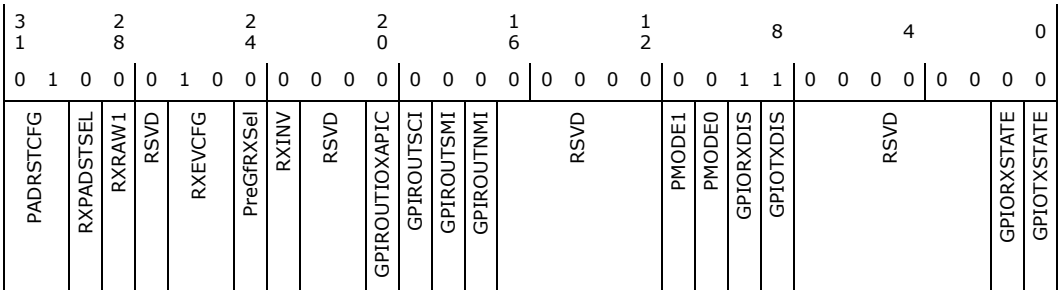
18.11.83 Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_1)—Offset 4C8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)

Bit Range	Default & Access	Field Name (ID): Description
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

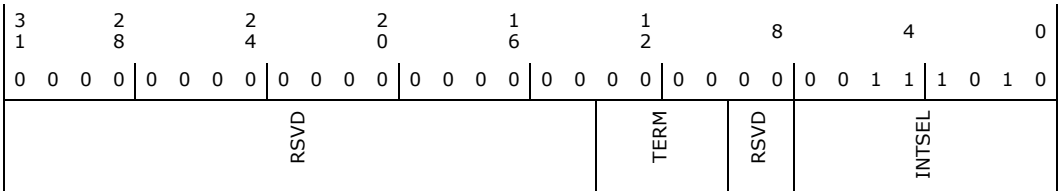
18.11.84 Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_1)—Offset 4CCh

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 3Ah



Bit Range	Default & Access	Field Name (ID): Description
31:14	00000h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
13:10	0h RO	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	3Ah RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

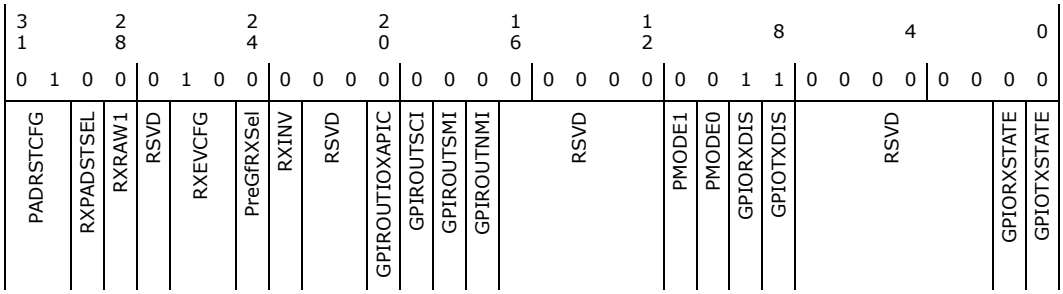
18.11.85 Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_2)—Offset 4D0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)

Bit Range	Default & Access	Field Name (ID): Description
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

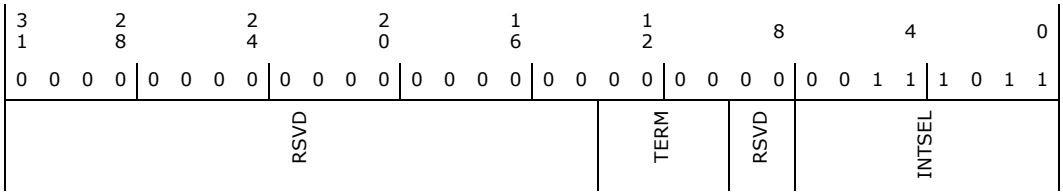
18.11.86 Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_2)—Offset 4D4h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 3Bh



Bit Range	Default & Access	Field Name (ID): Description
31:14	00000h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
13:10	0h RO	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	3Bh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

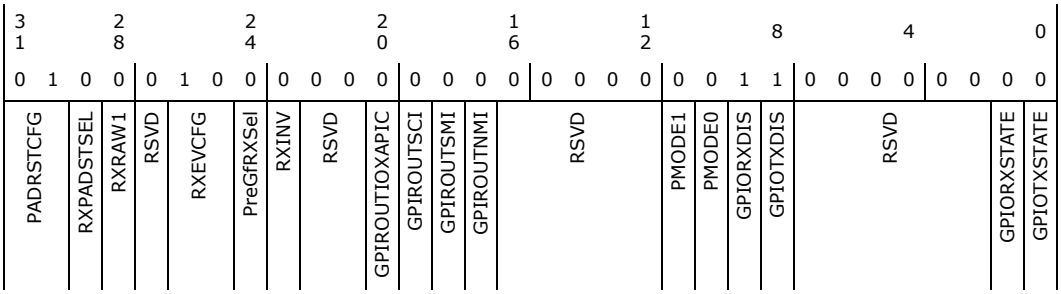
18.11.87 Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_3)—Offset 4D8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)

Bit Range	Default & Access	Field Name (ID): Description
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

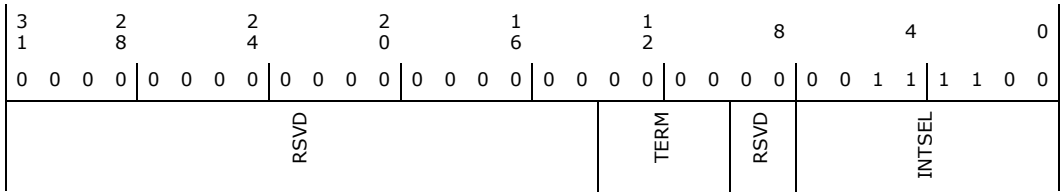
18.11.88 Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_3)—Offset 4DCh

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 3Ch



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
13:10	0h RO	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	3Ch RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

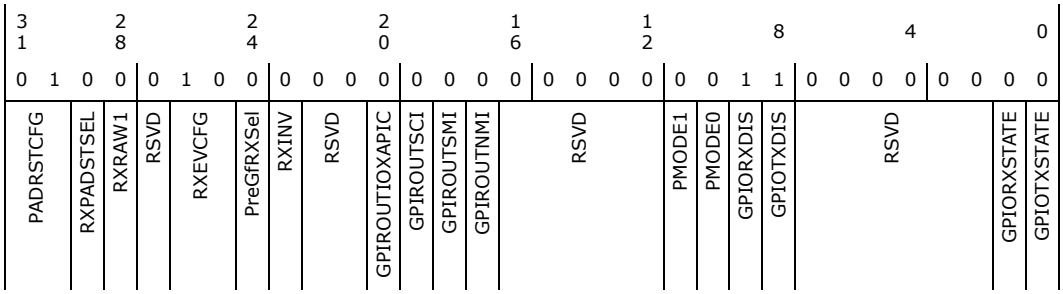
18.11.89 Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_4)—Offset 4E0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)

Bit Range	Default & Access	Field Name (ID): Description
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

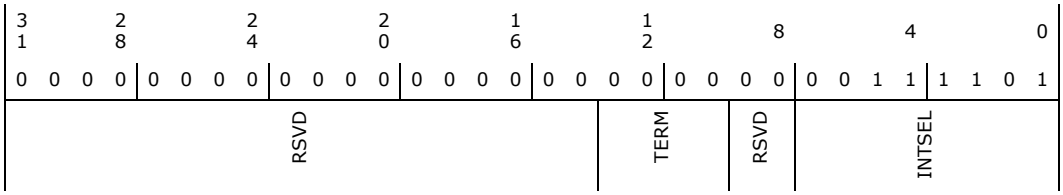
18.11.90 Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_4)—Offset 4E4h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 3Dh



Bit Range	Default & Access	Field Name (ID): Description
31:14	00000h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
13:10	0h RO	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	3Dh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

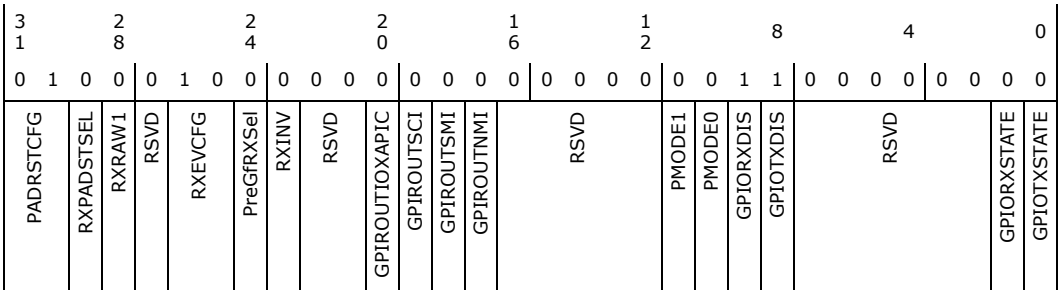
18.11.91 Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_5)—Offset 4E8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)

Bit Range	Default & Access	Field Name (ID): Description
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	Reserved.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

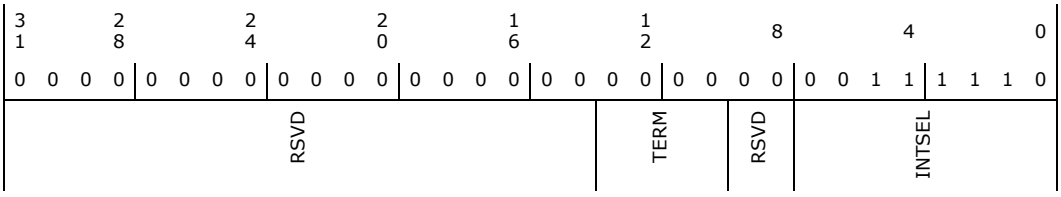
18.11.92 Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_5)—Offset 4ECh

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 3Eh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
13:10	0h RO	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	3Eh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

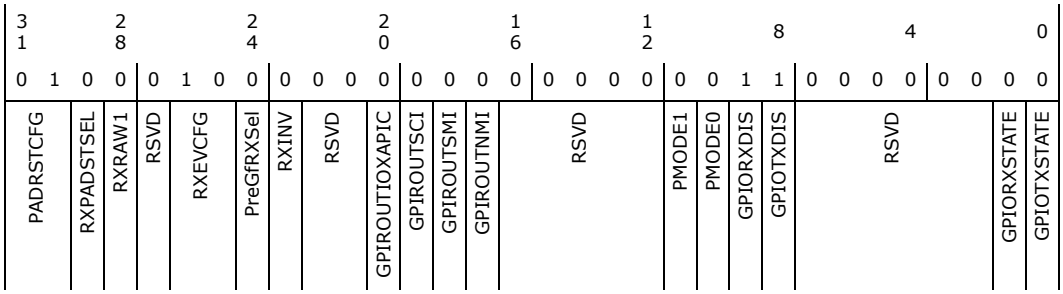
18.11.93 Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_6)—Offset 4F0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)

Bit Range	Default & Access	Field Name (ID): Description
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

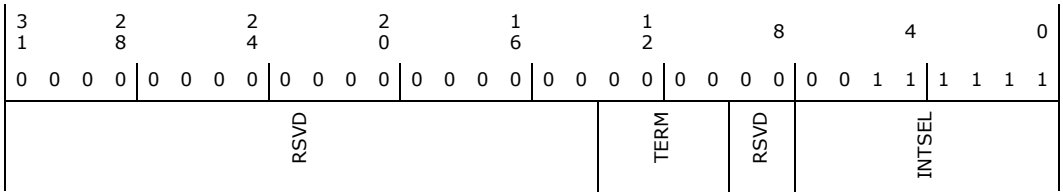
18.11.94 Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_6)—Offset 4F4h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 3Fh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
13:10	0h RO	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	3Fh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

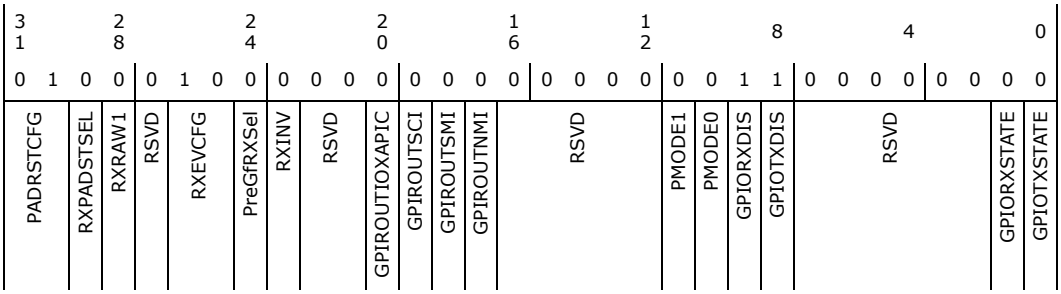
18.11.95 Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_7)—Offset 4F8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)

Bit Range	Default & Access	Field Name (ID): Description
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

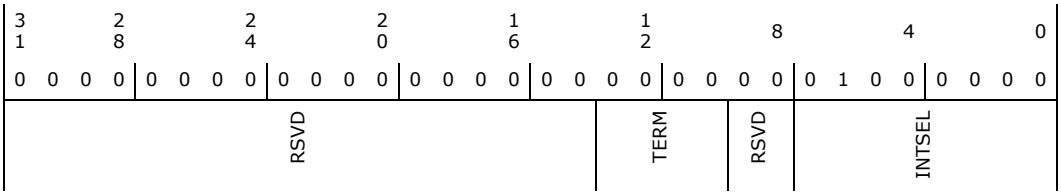
18.11.96 Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_7)—Offset 4FCh

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 40h



Bit Range	Default & Access	Field Name (ID): Description
31:14	00000h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
13:10	0h RO	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	40h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

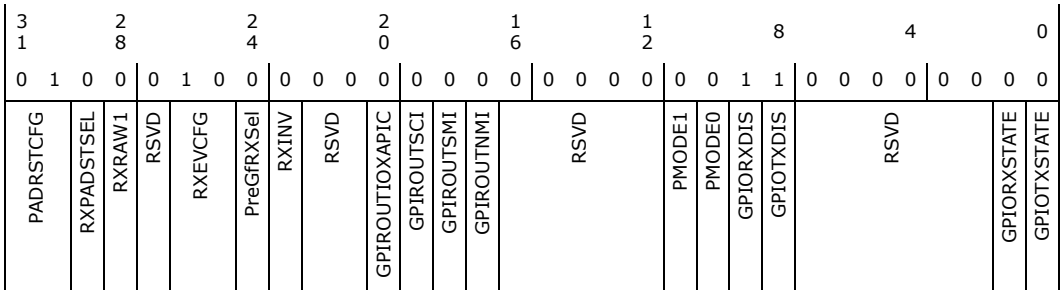
18.11.97 Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_8)—Offset 500h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)

Bit Range	Default & Access	Field Name (ID): Description
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

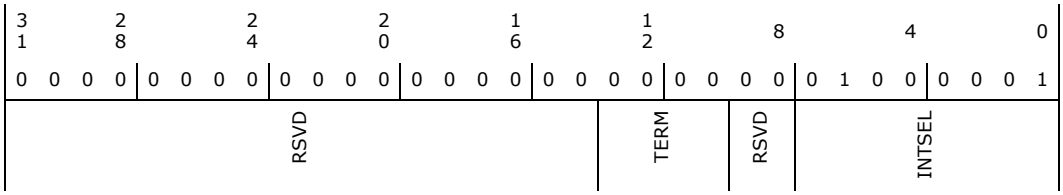
18.11.98 Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_8)—Offset 504h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 41h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
13:10	0h RO	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	41h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

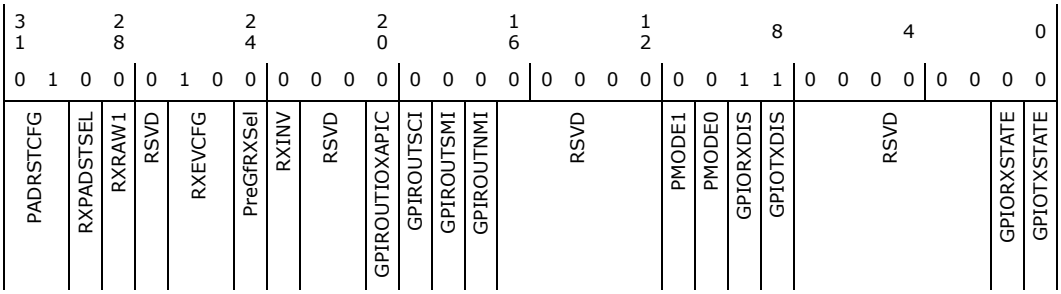
18.11.99 Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_9)—Offset 508h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)

Bit Range	Default & Access	Field Name (ID): Description
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

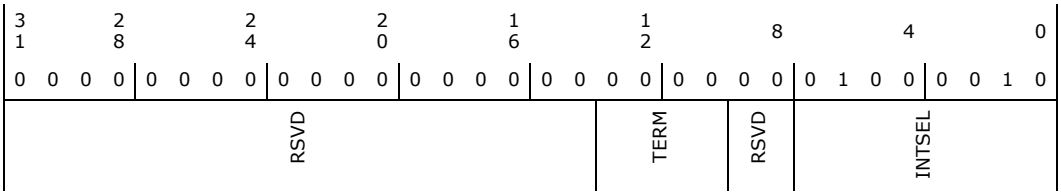
18.11.100Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_9)—Offset 50Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 42h



Bit Range	Default & Access	Field Name (ID): Description
31:14	00000h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
13:10	0h RO	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	42h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

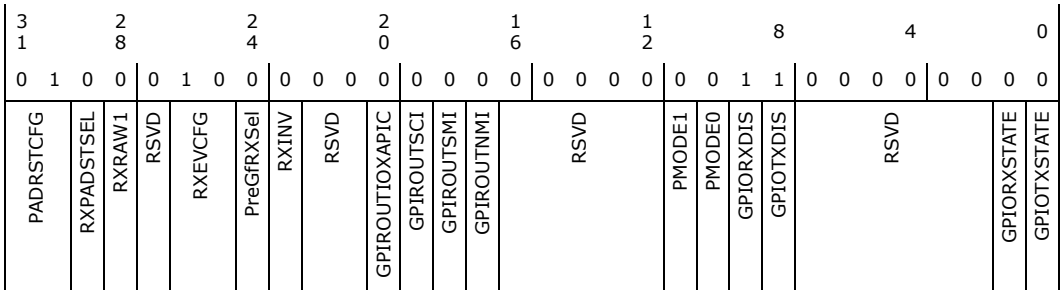
18.11.101 Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_10)—Offset 510h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)

Bit Range	Default & Access	Field Name (ID): Description
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

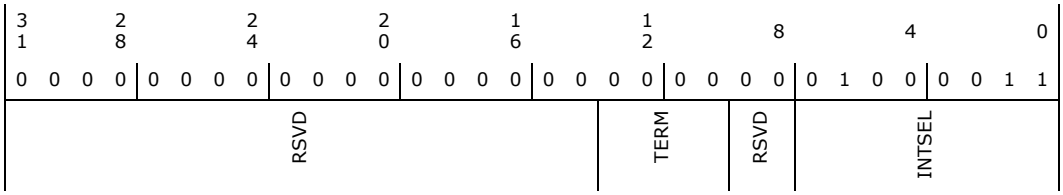
18.11.102 Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_10)—Offset 514h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 43h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	43h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

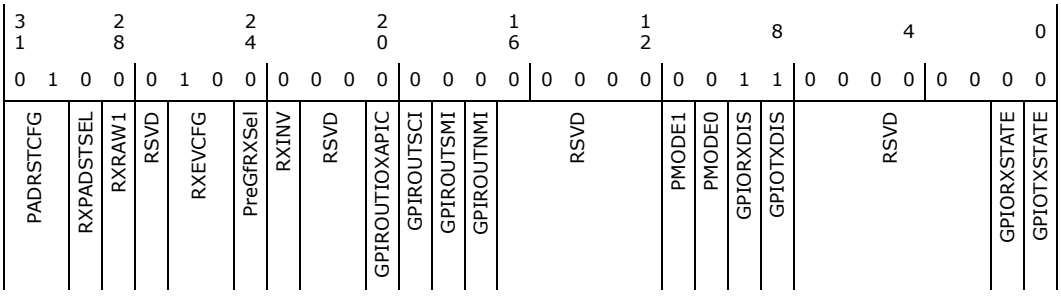
18.11.103Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_11)—Offset 518h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

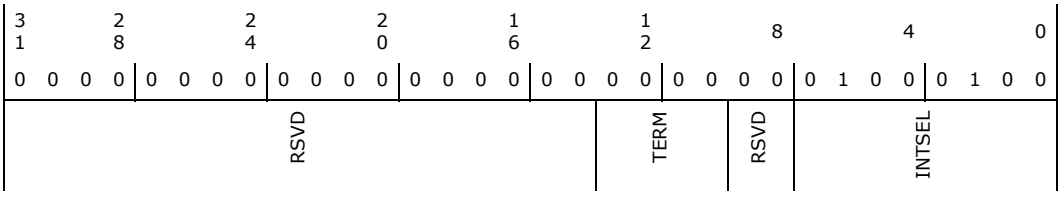
18.11.104Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_11)—Offset 51Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	44h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.105 Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_12)—Offset 520h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000200h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

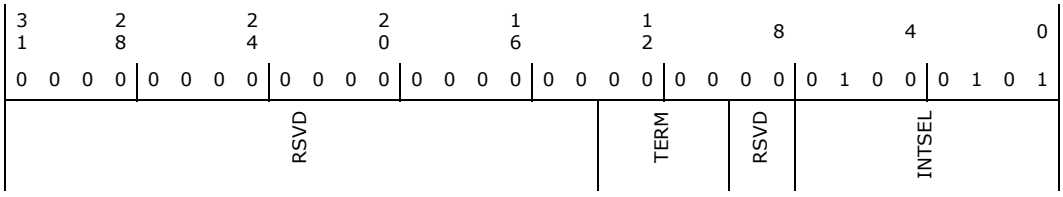
18.11.106 Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_12)—Offset 524h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 45h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	45h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

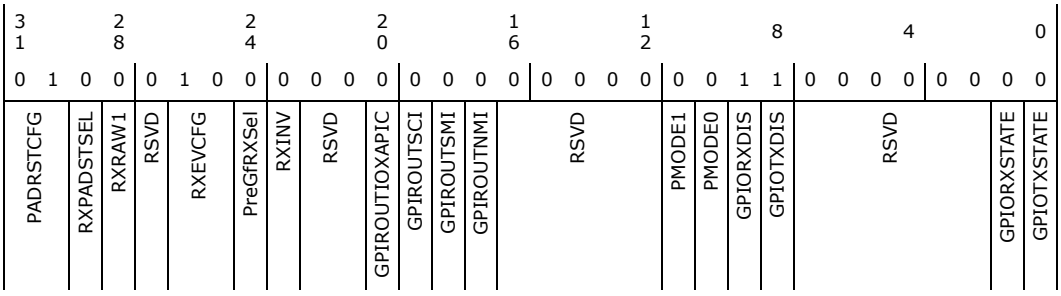
18.11.107 Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_13)—Offset 528h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

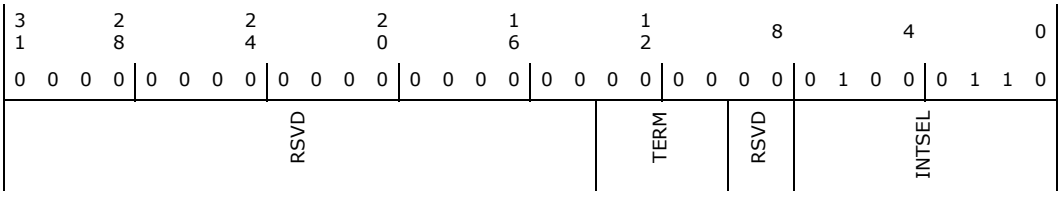
18.11.108 Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_13)—Offset 52Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 46h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	46h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.109 Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_14)—Offset 530h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

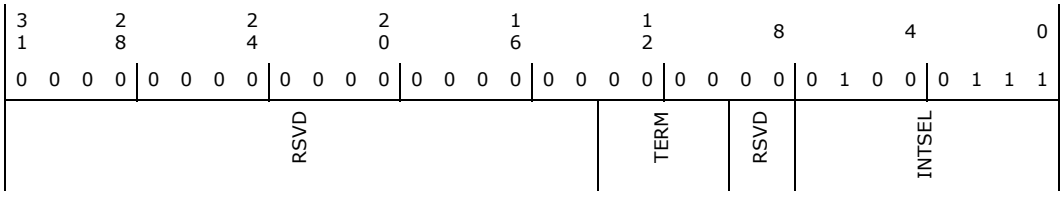
18.11.110 Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_14)—Offset 534h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 47h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	47h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

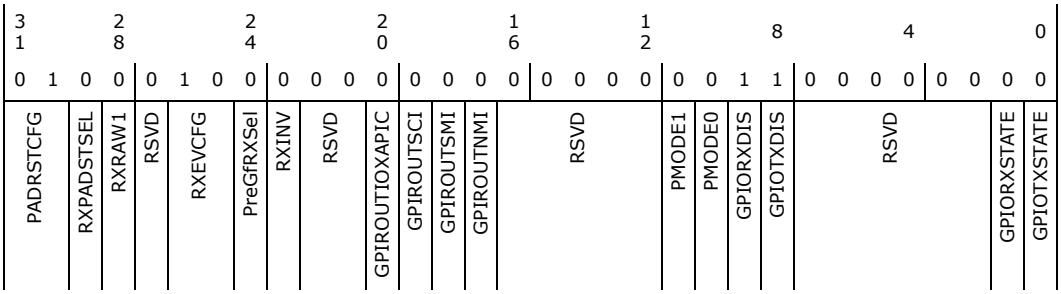
18.11.111Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_15)—Offset 538h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

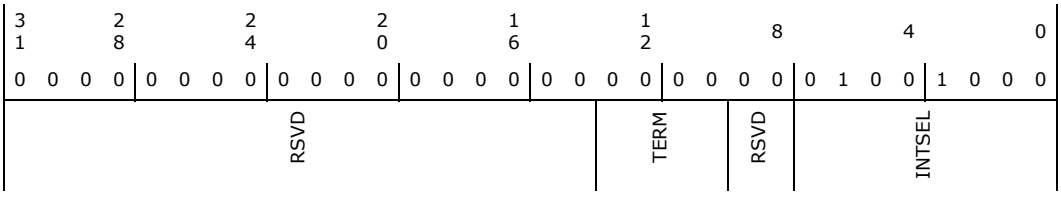
18.11.112Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_15)—Offset 53Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 48h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	48h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.113Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_16)—Offset 540h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

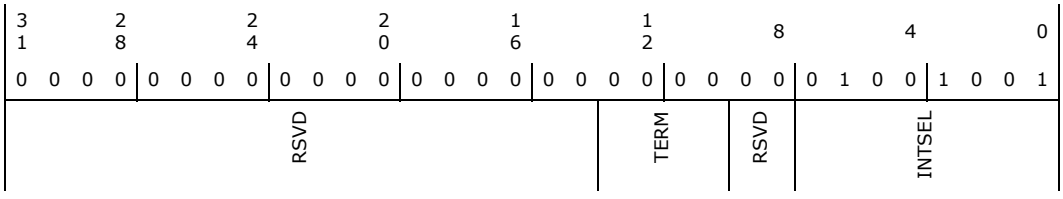
18.11.114 Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_16)—Offset 544h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 49h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	49h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

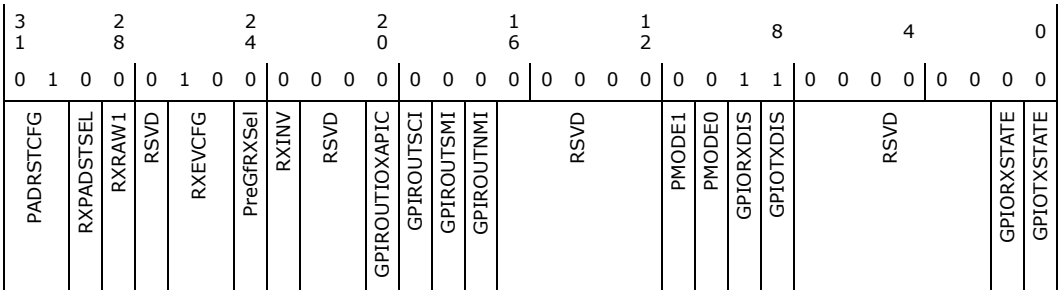
18.11.115Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_17)—Offset 548h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

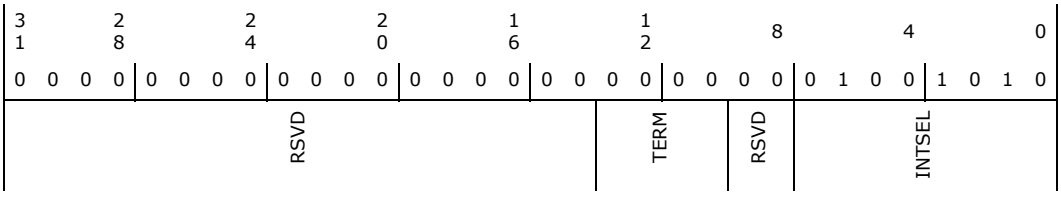
18.11.116 Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_17)—Offset 54Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 4Ah



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	4Ah RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

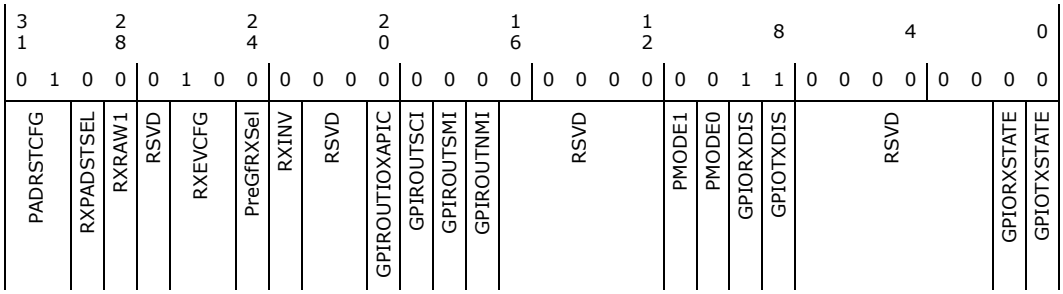
18.11.117Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_18)—Offset 550h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

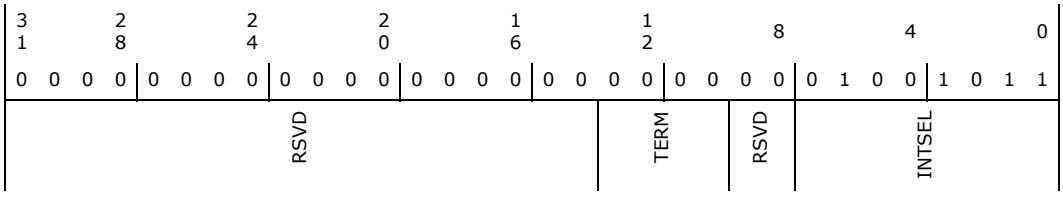
18.11.118 Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_18)—Offset 554h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 4Bh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	4Bh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

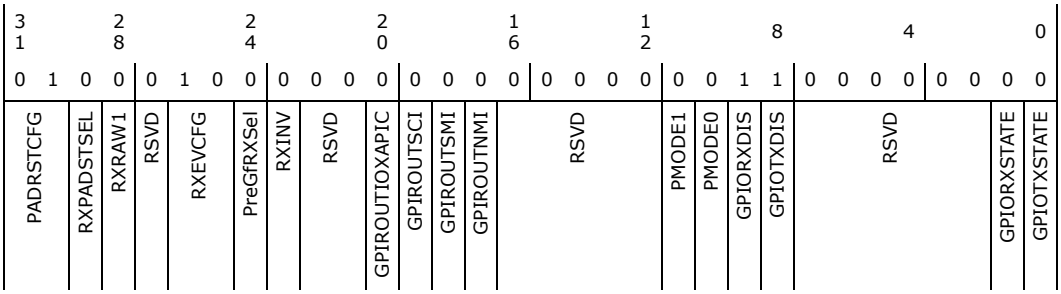
18.11.119Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_19)—Offset 558h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

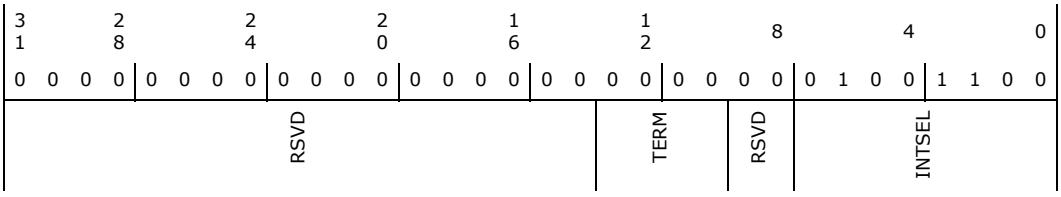
18.11.120Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_19)—Offset 55Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 4Ch



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
13:10	0h RO	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	4Ch RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

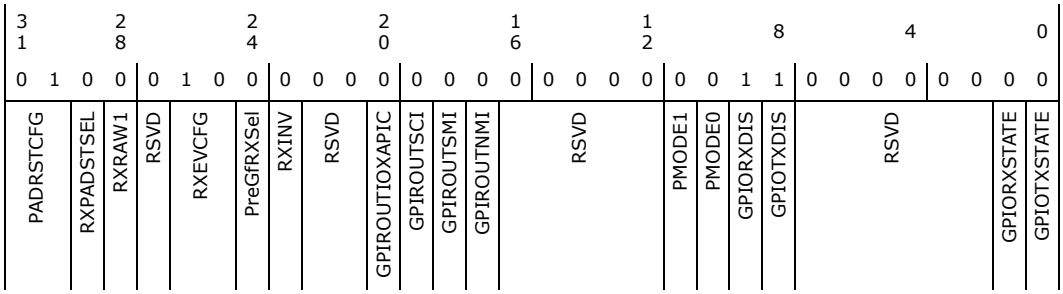
18.11.121 Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_20)—Offset 560h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)

Bit Range	Default & Access	Field Name (ID): Description
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

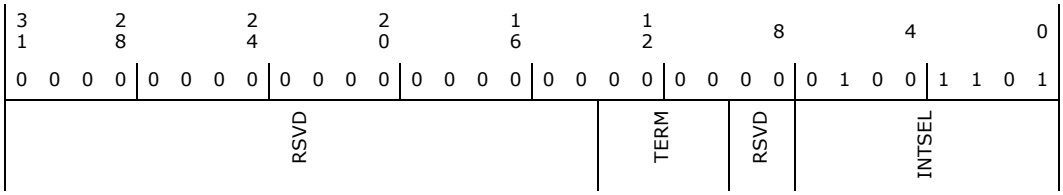
18.11.122Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_20)—Offset 564h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 4Dh



Bit Range	Default & Access	Field Name (ID): Description
31:14	00000h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
13:10	0h RO	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	4Dh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

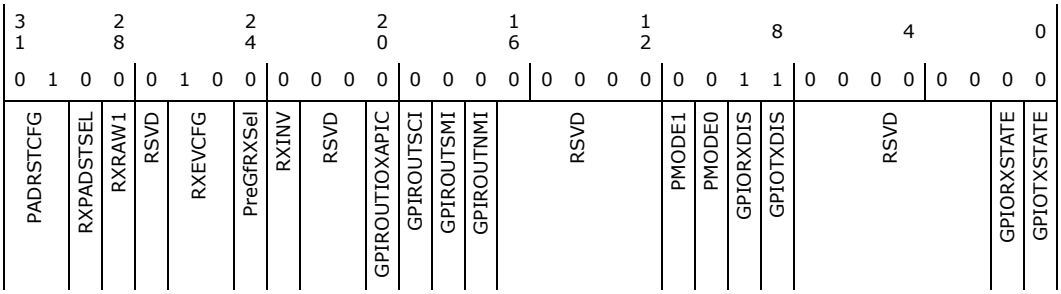
18.11.123Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_21)—Offset 568h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)

Bit Range	Default & Access	Field Name (ID): Description
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

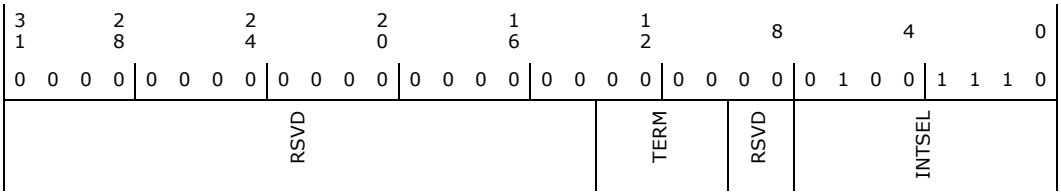
18.11.124 Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_21)—Offset 56Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 4Eh



Bit Range	Default & Access	Field Name (ID): Description
31:14	00000h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
13:10	0h RO	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	4Eh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

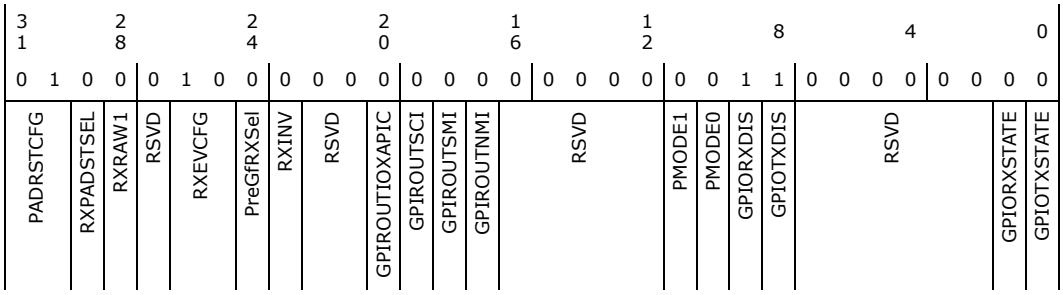
18.11.125 Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_22)—Offset 570h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

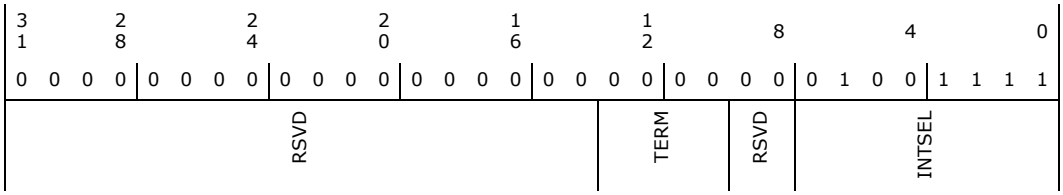
18.11.126 Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_22)—Offset 574h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 4Fh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RO	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	4Fh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

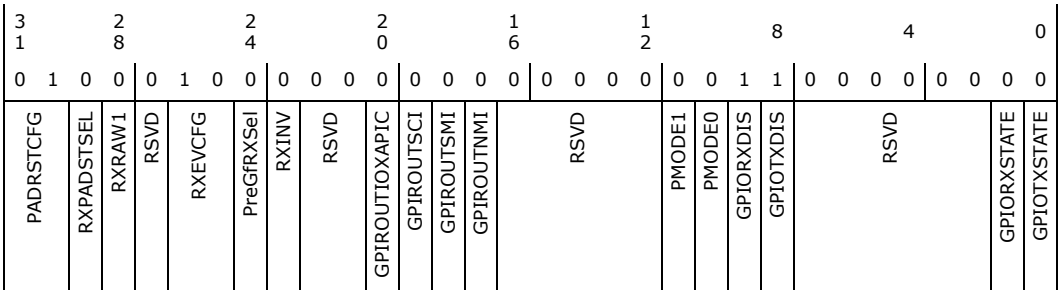
18.11.127Pad Configuration DW0 (PAD_CFG_DW0_GPP_H_23)—Offset 578h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

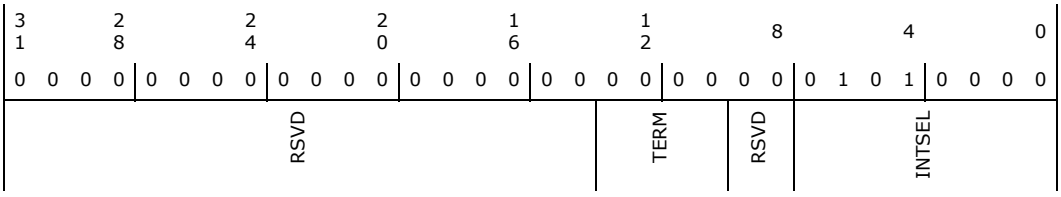
18.11.128Pad Configuration DW1 (PAD_CFG_DW1_GPP_H_23)—Offset 57Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 50h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	50h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

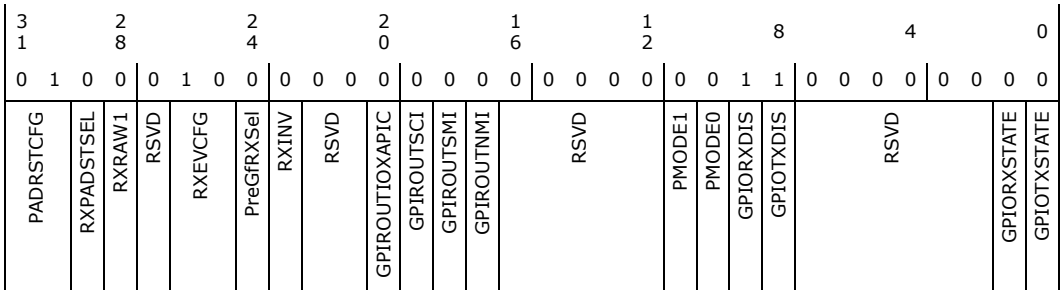
18.11.129 Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_0)—Offset 580h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000300h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)

Bit Range	Default & Access	Field Name (ID): Description
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.

Bit Range	Default & Access	Field Name (ID): Description
11	0h RO	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	0h RO	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

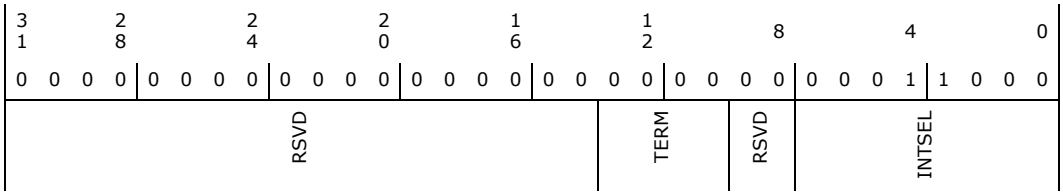
18.11.130 Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_0)—Offset 584h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 18h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	18h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.131 Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_1)—Offset 588h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000700h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
PADRSTCFG		RXPADSTSEL	RXRAW1	RSVD		RXEVCFG	PreGfRXSel	RXINV	RSVD		GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD						PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD					GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. When the input buffer is disabled, the internal pad state is always driven to '0'.
8	1h RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

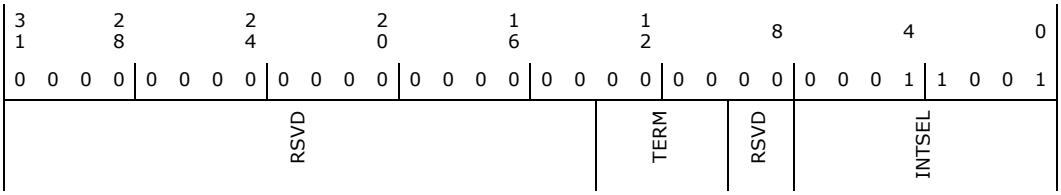
18.11.132Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_1)—Offset 58Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 19h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	19h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.133Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_2)—Offset 590h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000600h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

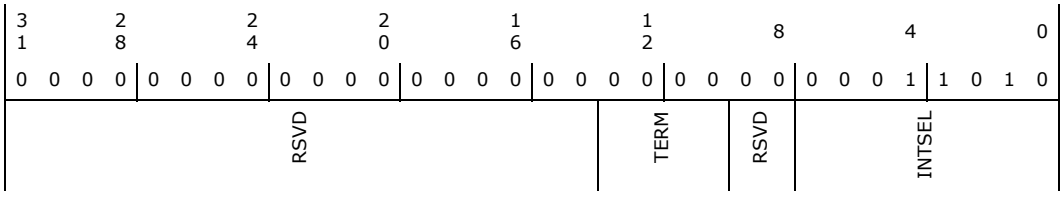
18.11.134 Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_2)—Offset 594h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Ah



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	1Ah RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

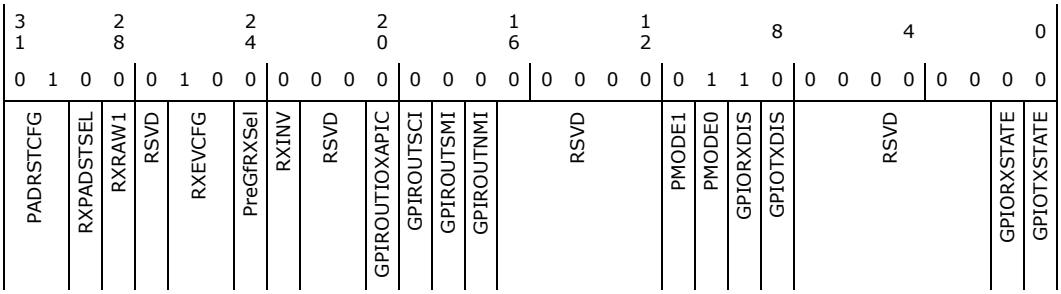
18.11.135Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_3)—Offset 598h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000600h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

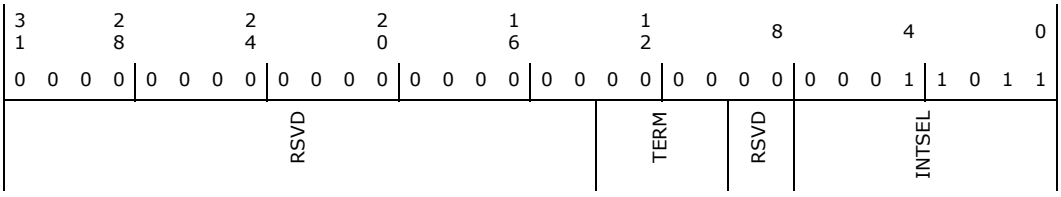
18.11.136 Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_3)—Offset 59Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Bh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	1Bh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.137Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_4)—Offset 5A0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000600h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

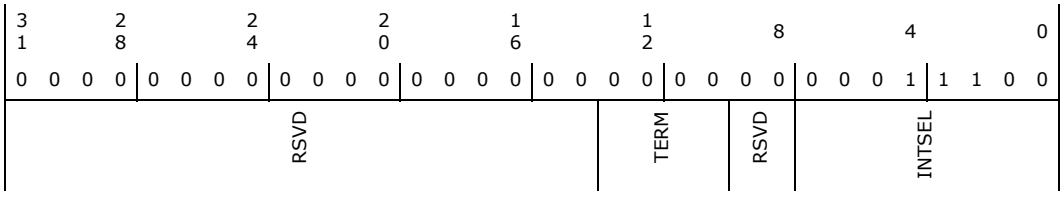
18.11.138 Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_4)—Offset 5A4h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Ch



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	1Ch RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.139 Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_5)—Offset 5A8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000600h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
PADRSTCFG		RXPADSTSEL	RXRAW1	RSVD		RXEVCFG	PreGfRXSel	RXINV	RSVD		GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD						PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD					GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). ***Note:*** This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

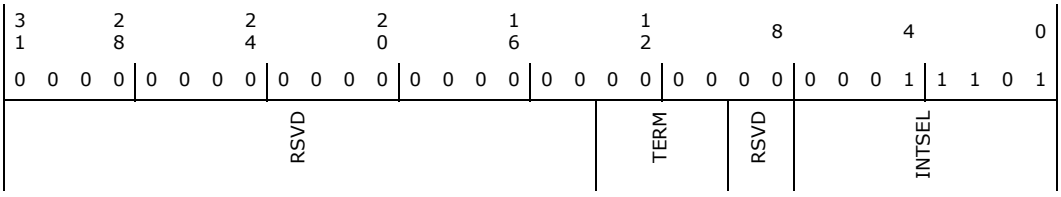
18.11.140 Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_5)—Offset 5ACh

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Dh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	1Dh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.141 Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_6)—Offset 5B0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000600h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

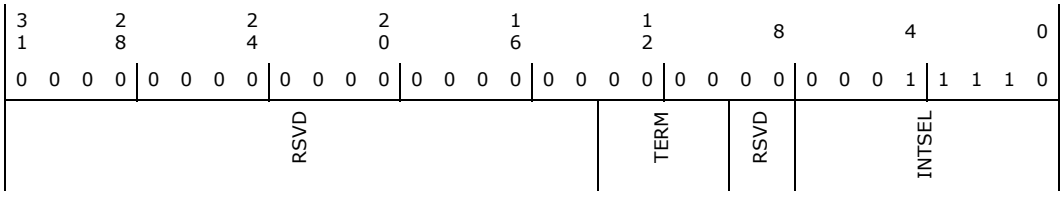
18.11.142 Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_6)—Offset 5B4h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Eh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	1Eh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

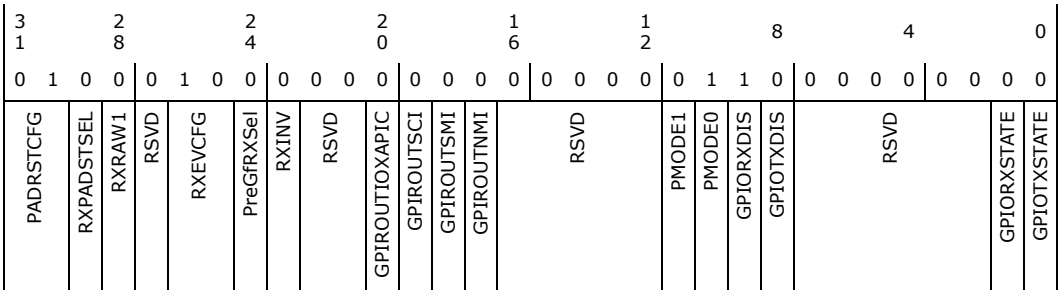
18.11.143Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_7)—Offset 5B8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000600h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

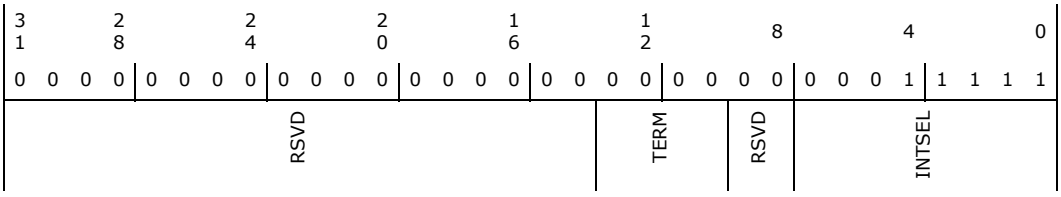
18.11.144Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_7)—Offset 5BCh

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1Fh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	1Fh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.145 Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_8)—Offset 5C0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000600h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

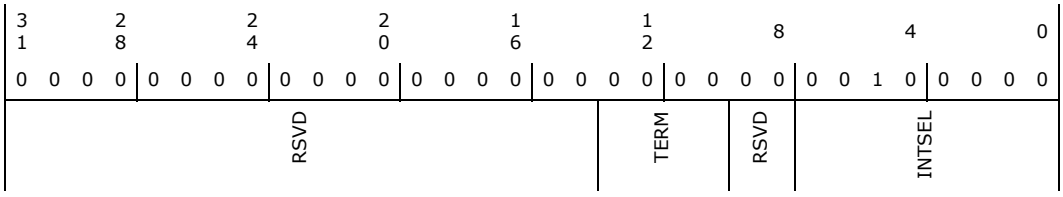
18.11.146 Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_8)—Offset 5C4h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 20h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	20h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

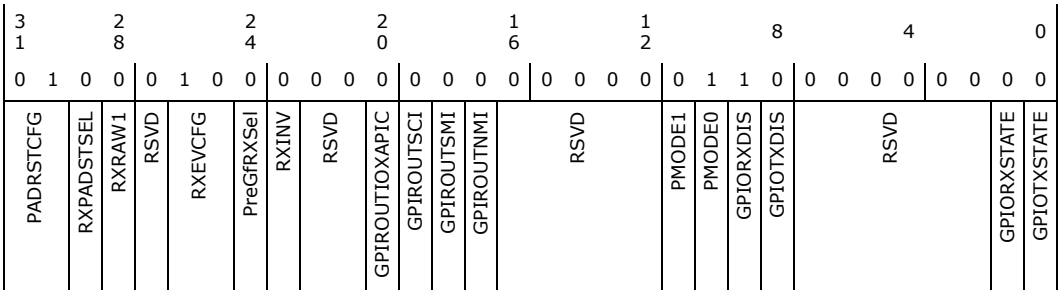
18.11.147Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_9)—Offset 5C8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000600h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

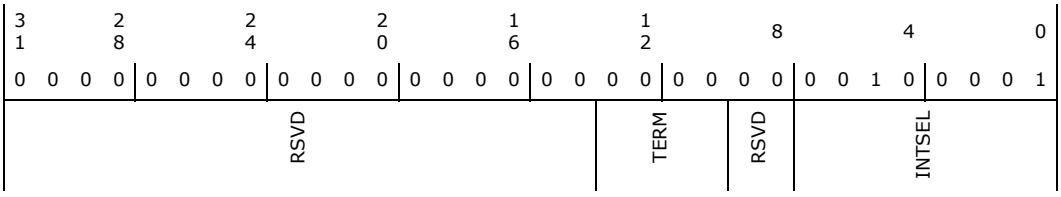
18.11.148Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_9)—Offset 5CCh

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 21h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	21h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.149 Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_10)—Offset 5D0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000600h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

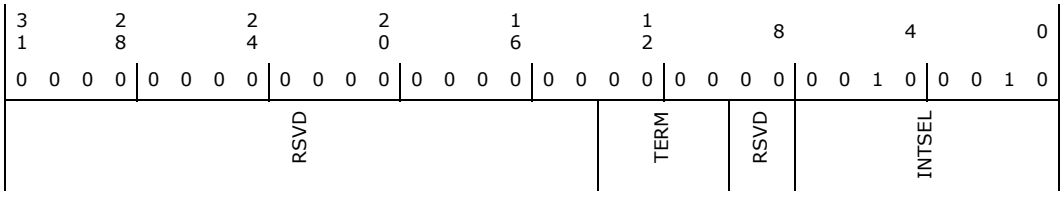
18.11.150 Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_10)—Offset 5D4h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 22h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	22h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

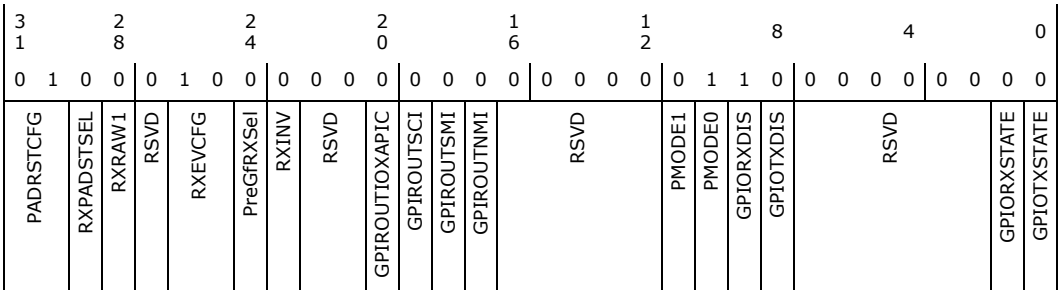
18.11.151 Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_11)—Offset 5D8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000600h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

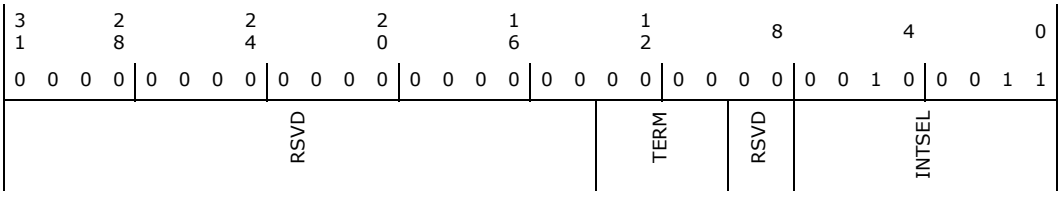
18.11.152 Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_11)—Offset 5DCh

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 23h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	23h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.153Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_12)—Offset 5E0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000600h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

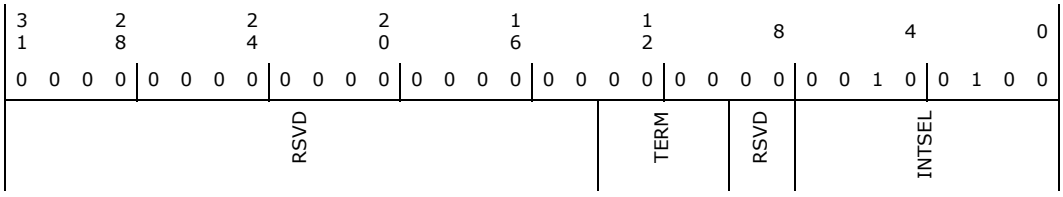
18.11.154 Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_12)—Offset 5E4h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 24h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	24h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

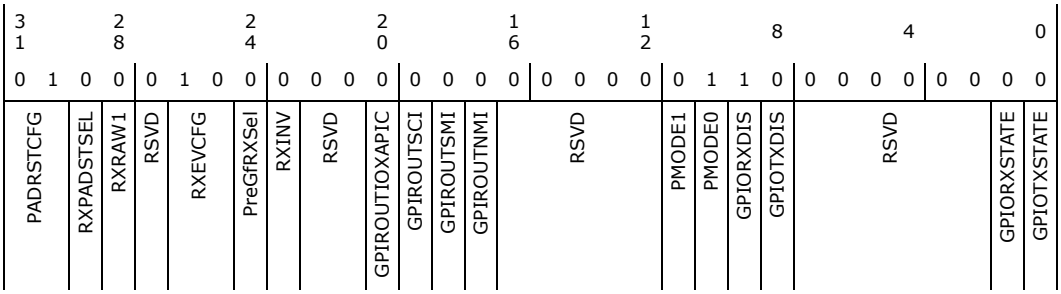
18.11.155Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_13)—Offset 5E8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000600h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

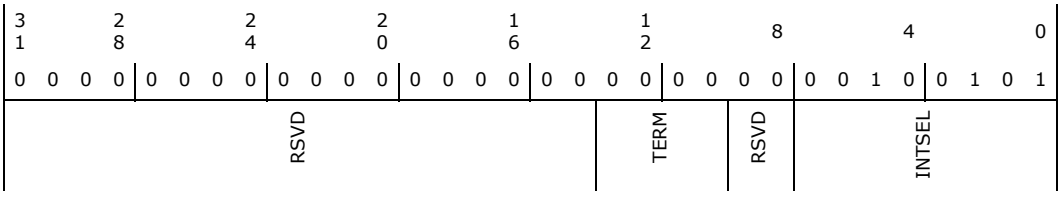
18.11.156 Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_13)—Offset 5ECh

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 25h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	25h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.157Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_14)—Offset 5F0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000600h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

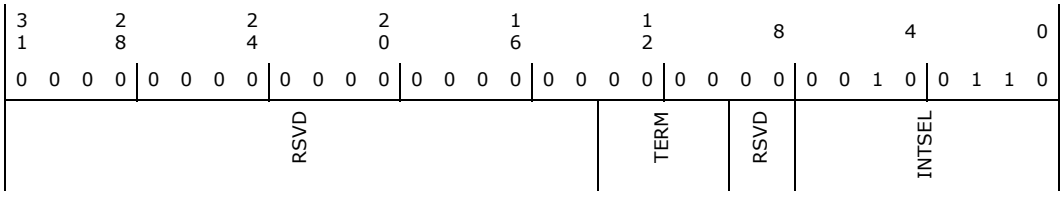
18.11.158 Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_14)—Offset 5F4h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 26h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	26h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

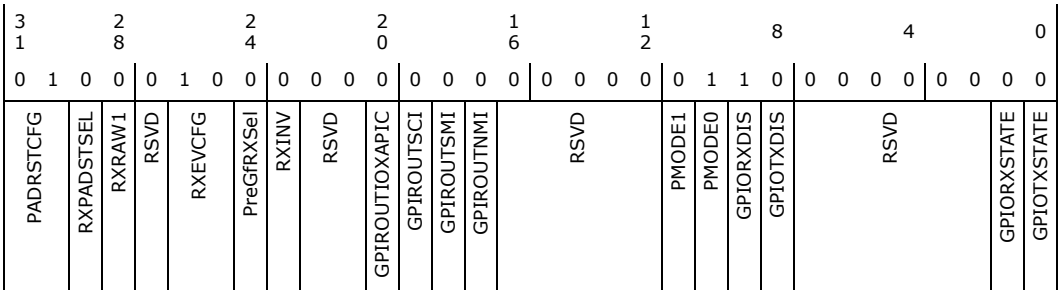
18.11.159 Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_15)—Offset 5F8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000600h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

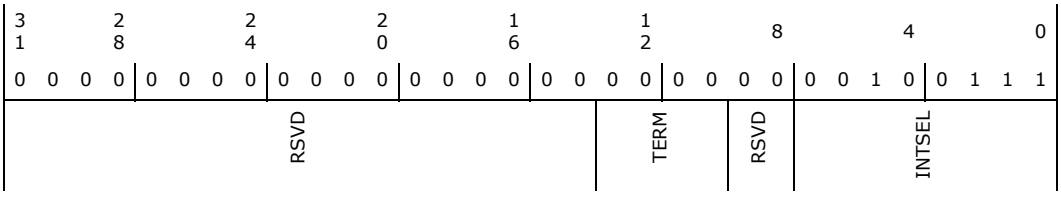
18.11.160 Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_15)—Offset 5FCh

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 27h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	27h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.161 Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_16)—Offset 600h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000600h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
PADRSTCFG		RXPADSTSEL		RXRAW1		RSVD		RXEVCFG		PreGfRXSel		RXINV		RSVD		GPIROUTIOXAPIC		GPIROUTSCI		GPIROUTSMI		GPIROUTNMI		RSVD		PMODE1		PMODE0		GPIORXDIS		GPIOTXDIS		RSVD		GPIORXSTATE		GPIOTXSTATE	

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

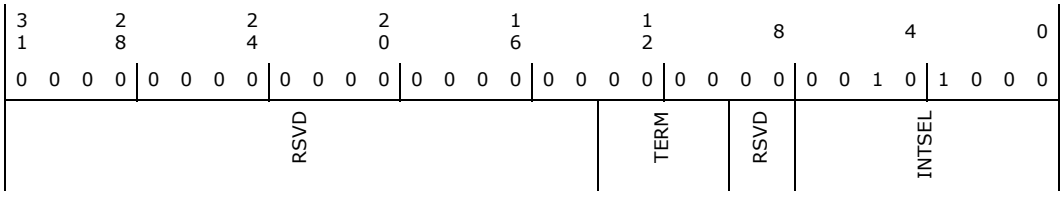
18.11.162 Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_16)—Offset 604h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 28h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	28h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

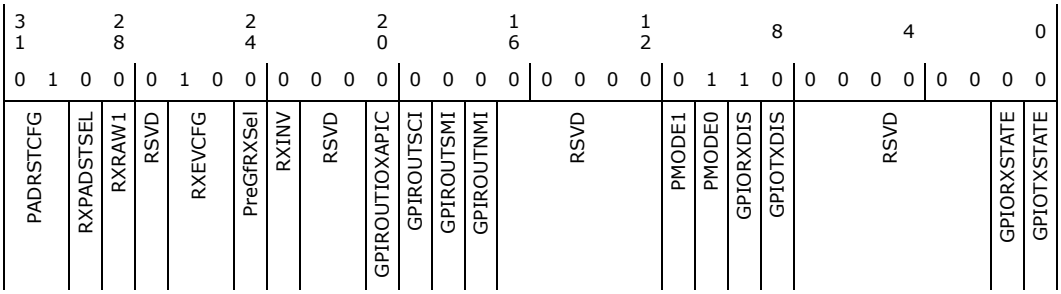
18.11.163Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_17)—Offset 608h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000600h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

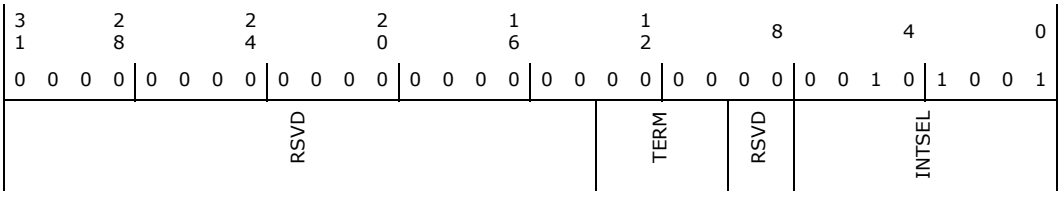
18.11.164 Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_17)—Offset 60Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 29h



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	29h RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

18.11.165 Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_18)—Offset 610h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000600h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
PADRSTCFG	RXPADSTSEL	RXRAW1	RSVD	RXEVCFG	PreGfRXSel	RXINV	RSVD	GPIROUTIOXAPIC	GPIROUTSCI	GPIROUTSMI	GPIROUTNMI	RSVD								PMODE1	PMODE0	GPIORXDIS	GPIOTXDIS	RSVD						GPIORXSTATE	GPIOTXSTATE

Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit 1. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

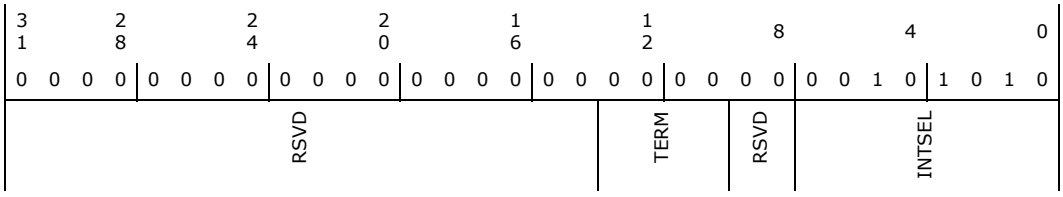
18.11.166 Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_18)—Offset 614h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 2Ah



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	2Ah RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

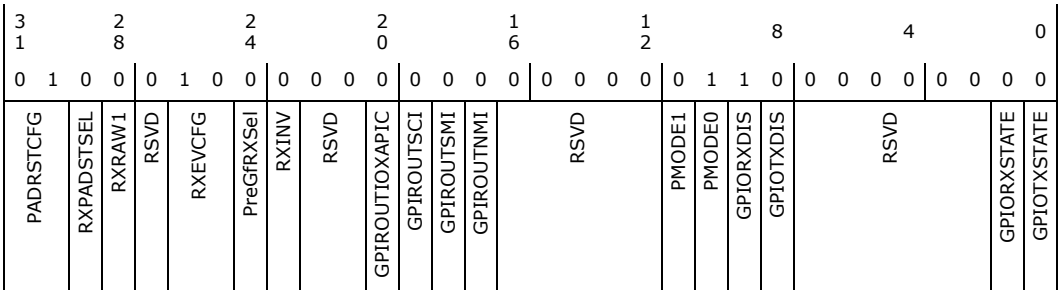
18.11.167Pad Configuration DW0 (PAD_CFG_DW0_GPP_L_19)—Offset 618h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 44000600h



Bit Range	Default & Access	Field Name (ID): Description
31:30	1h RW	**Pad Reset Config (PADRSTCFG):** This register controls which reset is used to reset GPIO pad register fields in PAD_CFG_DW0 and PAD_CFG_DW1 registers. This register can be used for Sx isolation of the associated signal if needed. 00 = RSMRST# 01 = Host deep reset. This reset occurs when a host reset (with or without power cycle) is initiated or a global reset is initiated, except that the reset does not assert when in S3/S4/S5. 10 = PLTRST# 11 = Reserved

Bit Range	Default & Access	Field Name (ID): Description
29	0h RW	**RX Pad State Select (RXPADSTSEL):** Determines from which node the RX pad state for native function should be taken from. This field only affects the pad state value being fanned out to native function(s) and is not meaningful if the pad is in GPIO mode (i.e. Pad Mode = 0). 0 = Raw RX pad state directly from RX buffer 1 = Internal RX pad state (subject to RXINV and PreGfRXSel settings)
28	0h RW	**RX Raw Override to '1' (RXRAW1):** This bit determines if the selected pad state is being overridden to '1'. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The override takes place at the internal pad state directly from buffer and before the RXINV. 0 = No Override 1 = RX drive 1 internally
27	0h RO	Reserved.
26:25	2h RW	**RX Level/Edge Configuration (RXEVCFG):** Determines if the internal RX pad state (synchronized, filtered vs non-filtered version as determined by PreGfRXSel, and is further subject to RXInv) should be passed on to the next logic stage as is, as a pulse, or level signal. This field does not affect the received pad state (to GPIORXState or native functions) but how the interrupt or wake triggering events should be delivered to the GPIO Community Controller. 0h = Level 1h = Edge 2h = Drive '0' 3h = Reserved (implement as setting 0h)
24	0h RO	**Pre Glitch Filter Stage RX Pad State Select (PreGfRXSel):** Non-filtered only.
23	0h RW	**RX Invert (RXINV):** This bit determines if the selected pad state should go through the polarity inversion stage. This field is only applicable when the RX buffer is configured as an input in either GPIO Mode or native function mode. The polarity inversion takes place at the mux node of raw vs filtered or non-filtered RX pad state, as determined by PreGfRXsel and RXPadStSel This bit does not affect GPIORXState. During host ownership GPIO Mode, when this bit is set to '1', then the RX pad state is inverted as it is sent to the GPIO-to-IOxAPIC, GPE/SCI, SMI, NMI logic or GPI_IS[n] that is using it. This is used to allow active-low and active-high inputs to cause IRQ, SMI#, SCI or NMI. 0 = No inversion 1 = Inversion
22:21	0h RO	Reserved.
20	0h RW	**GPIO Input Route IOxAPIC (GPIROUTIOXAPIC):** Determines if the pad can be routed to cause peripheral IRQ when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause peripheral IRQ 1 = Routing can cause peripheral IRQ ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the peripheral IRQ indication to the intended recipient(s).
19	0h RW	**GPIO Input Route SCI (GPIROUTSCI):** Determines if the pad can be routed to cause SCI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SCI. 1 = Routing can cause SCI ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the GPE indication to the intended recipient(s).
18	0h RW	**GPIO Input Route SMI (GPIROUTSMI):** Determines if the pad can be routed to cause SMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause SMI. 1 = Routing can cause SMI. ***Note:*** This bit does not affect any interrupt status bit within GPIO, but is used as the last qualifier for the SMI indication to the intended recipient(s). This bit only applies to a GPIO that has SMI capability. Otherwise, the bit is RO.

Bit Range	Default & Access	Field Name (ID): Description
17	0h RW	**GPIO Input Route NMI (GPIROUTNMI):** Determines if the pad can be routed to cause NMI when configured in GPIO input mode. If the pad is not configured in GPIO input mode, this field has no effect. 0 = Routing does not cause NMI. 1 = Routing can cause NMI. ***Note:*** This bit also affects GPI_NMI_STS. If '0', GPI_NMI_STS is always clear. If '1', GPI_NMI_STS could be set (depending on GPIOOwn setting) when there is an event. Whether a NMI indication is generated and sent to the intended recipient(s) is also depending on the corresponding GPI_NMI_EN bit. This bit only applies to a GPIO that has NMI capability. Otherwise, the bit is RO.
16:12	0h RO	Reserved.
11	0h RW	**Pad Mode bit 1 (PMODE1):** See Pad Mode Bit 0 description.
10	1b RW	**Pad Mode bit 0 (PMODE0):** This bit is used in conjunction with Pad Mode bit r. These two-bit field determines whether the Pad is controlled by GPIO controller logic or one of the native functions muxed onto the Pad. 0h = GPIO control the Pad. 1h = native function 1, if applicable, controls the Pad 2h = native function 2, if applicable, controls the Pad 3h = native function 3, if applicable, controls the Pad Dedicated (unmuxed) GPIO shall report RO of all 0's in this register field If GPIO vs. native mode is configured via soft strap, this bit has no effect.
9	1h RW	**GPIO RX Disable (GPIORXDIS):** 0 = Enable the input buffer (active low enable) of the pad. 1 = Disable the input buffer of the pad. ***Note:*** When the input buffer is disabled, the internal pad state is always driven to '0'.
8	0b RW	**GPIO TX Disable (GPIOTXDIS):** 0 = Enable the output buffer (active low enable) of the pad. 1 = Disable the output buffer of the pad; i.e. Hi-Z
7:2	0h RO	Reserved.
1	0h RO	**GPIO RX State (GPIORXSTATE):** This is the current internal RX pad state after Glitch Filter logic stage and is not affected by PMode and RXINV settings.
0	0h RW	**GPIO TX State (GPIOTXSTATE):** 0 = Drive a level '0' to the TX output pad. 1 = Drive a level '1' to the TX output pad

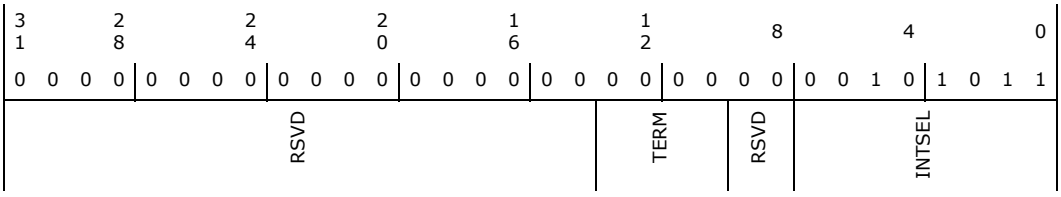
18.11.168 Pad Configuration DW1 (PAD_CFG_DW1_GPP_L_19)—Offset 61Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 2Bh



Bit Range	Default & Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Termination (TERM):** The Pad Termination state defines the different weak pull-up and pull-down settings that are supported by the buffer. The settings for [13:10] correspond to: 0000: none 0010: 5k pd 0100: 20k pd 1000: none 1010: 5k pu 1100: 20k pu All others reserved. If a reserved value is programmed, pad may malfunction. The setting of this field is applicable in all Pad Mode including GPIO. As each Pad Mode may require different termination and isolation, care must be taken in SW/FW in the transition with appropriate register programming. The actual transition sequence requirement may vary on case by case basis depending on the native functions involved. For example, before changing the pad from output to input direction, pu/pd settings should be programmed first to ensure the input does not float momentarily.
9:8	0h RO	Reserved.
7:0	2Bh RO	**Interrupt Select (INTSEL):** The Interrupt Select defines which interrupt line driven from the GPIO Controller toggles when an interrupt is detected on this pad. 0 = Interrupt Line 0 1 = Interrupt Line 1 .... Up to the max IOxAPIC IRQ supported

§

19 Gigabit Ethernet Controller

19.1 Acronyms

Acronyms	Description
GbE	Gigabit Ethernet
ACPI	Advanced Configuration and Power Interface
APM	Advanced Power Management

19.2 References

Specification	Location
Alert Standard Format Specification, Version 1.03	http://www.dmtf.org/standards/asf
IEEE 802.3 Fast Ethernet	http://standards.ieee.org/getieee802/
Intel® i219 Gigabit Ethernet LAN Connect Device Datasheet	https://www.intel.com/content/www/us/en/products/network-io/ethernet/controllers/connection-i219-lm.html

19.3 Overview

The Gigabit Ethernet controller(D31:F6) in conjunction with the Intel® i219 Gigabit Ethernet LAN Connect Device provide a complete LAN solution. This chapter describes the behavior of the Gigabit Ethernet Controller. For details on the Intel® i219 Gigabit Ethernet LAN Connect Device, refer to document 544486 on the Resource and Design Center (RDC). The Gigabit Ethernet Controller can operate at multiple speeds (10/100/1000 Mbps) and in either full duplex or half duplex mode.

19.4 Signal Description

Table 19-1. GbE LAN Signals (Sheet 1 of 2)

Name	Type	Description
USB3_10_PCIE3_GBe_TXP **USB3_10_PCIE3_GBE_TXN** **PCIE4_GBE_TXP** **PCIE4_GBE_TXN** **PCIE5_GBE_TXP** **PCIE5_GBE_TXN** **PCIE8_SSATA2_GBE_TXP** **PCIE8_SSATA2_GBE_TXN** **PCIE11_SSATA5_GBE_TXP** **PCIE11_SSATA5_GBE_TXN**	O	***Note:*** Refer to Chapter 22 for details on the PCI Express transmit signals.The Intel® i219 Gigabit Ethernet LAN Connect Device can be connected to one of the following PCI Express ports 3, 4, 5, 8 and 11.

Table 19-1. GbE LAN Signals (Sheet 2 of 2)

Name	Type	Description
USB3_10_PCIE3_GBe_RXP **USB3_10_PCIE3_GBE_RXN** **PCIE4_GBE_RXP** **PCIE4_GBE_RXN** **PCIE5_GBE_RXP** **PCIE5_GBE_RXN** **PCIE8_SSATA2_GBE_RXP** **PCIE8_SSATA2_GBE_RXN** **PCIE11_SSATA5_GBE_RXP** **PCIE11_SSATA5_GBE_RXN**	I	***Note:*** Refer to Chapter 22 for details on the PCI Express receive signals.The Intel® i219 Gigabit Ethernet LAN Connect Device can be connected to one of the following PCI Express ports 3, 4, 5, 8 and 11. on PCH-H.
GPP_C4_SML0DATA_IE	I/OD	Refer to Chapter 25 for details on the SML0DATA signal. ***Note:*** The Intel® i219 Gigabit Ethernet LAN Connect Device connects to SML0DATA signal.
GPP_C3_SML0CLK_IE	I/OD	Refer to Chapter 25 for details on the SML0CLK signal. ***Note:*** The Intel® i219 Gigabit Ethernet LAN Connect Device connects to SML0CLK signal.
GPD11_GBEPHY	O	**GBE PHY Power Control:** GBD11_GBEPHY should be connected to LAN_DISABLE_N on the PHY. PCH will drive this signal low to put the PHY into a low power state when functionality is not needed. ***Note:*** GPD11_GBEPHY can only be driven low if SLP_GBE# is deasserted. ***Note:*** Signal can instead be used as GPD11.
SLP_GBE#	O	**LAN Sub-System Sleep Control:** When SLP_GBE# is de-asserted it indicates that the PHY device must be powered. When SLP_GBE# is asserted, power can be shut off to the PHY device. SLP_GBE# will always be deasserted in S0 and anytime SLP_A# is de-asserted.
GPD2_GBE_WAKE#	I	**GBE WAKE:** GbE Wake Indicator from the GbE PHY. ***Note:*** Signal can instead be used as GPD2.

19.5 Integrated Pull-Ups and Pull-Downs

Table 19-2. Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value Ω	Notes
GPD2_GBE_WAKE#	External pull-up required.	15k-40k	

19.6 I/O Signal Planes and States

Table 19-3. Power Plane and States for Output Signals

Signal Name	Power Plane	During Reset	Immediately After Reset	S3/S4/S5	Deep Sx
GPD11_GBEPHY	DSW	Driven Low	Driven Low	Driven Low	Driven Low
SLP_GBE#	DSW	Driven Low	Driven Low	0/1[1]	0/1[1]

Note:
1. Based on wake events and Intel ME state

Table 19-4. Power Plane and States for Input Signals

Signal Name	Power Plane	During Reset	Immediately After Reset	S3/S4/S5	Deep Sx
GPD2_G-BE_WAKE#	DSW	Undriven	Undriven	Undriven	Undriven/ Internal Pull-Down[2]

Notes:
1. Configurable
2. Configurable based on PMC configuration bit.
   '1' (pin will be driven by platform in DeepSx) -> Undriven;
   '0' (pin will NOT be driven by platform in DeepSx) -> Internal pull-down (15k-40k) enabled

19.7 Functional Description

The PCH integrates a Gigabit Ethernet controller. The integrated GbE controller is compatible with the Intel® i219 Gigabit Ethernet LAN Connect Device. The integrated GbE controller provides two interfaces for 10/100/1000 Mbps and manageability operation:

- Data link based on PCI Express – A high-speed interface that uses PCIe electrical signaling at half speed and custom logical protocol for active state operation mode.
- System Management Link (SMLink0) – A low speed connection for low power state mode for manageability communication only. The frequency of this connection can be configured to one of three different speeds (100 kHz, 400 kHz or 1 MHz).

The Intel® i219 Gigabit Ethernet LAN Connect Device only runs at a speed of 1250 Mbps, which is 1/2 of the 2.5 GB/s PCI Express frequency. Each of the PCI Express root ports in the PCH have the ability to run at the 1250 Mbps rate. There is no need to implement a mechanism to detect that the platform LAN device is connected. The port configuration (if any), attached to the platform LAN device, is pre-loaded from the NVM. The selected port adjusts the transmitter to run at the 1250 Mbps rate and does not need to be PCI Express compliant.

Note: PCIe validation tools cannot be used for electrical validation of this interface; however, PCIe layout rules apply for on-board routing.

The integrated GbE controller operates at full-duplex at all supported speeds or half-duplex at 10/100 Mbps. It also adheres to the *IEEE 802.3x Flow Control Specification.*

Note: GbE operation (1000 Mbps) is only supported in S0 mode. In Sx modes, SMBus is the only active bus and is used to support manageability/remote wake-up functionality.

Note: It is not possible to disable the device via the PSF registers. With the C620 series you must use the soft strap 119 to disable the device.

The integrated GbE controller provides a system interface using a PCI Express function. A full memory-mapped or I/O-mapped interface is provided to the software, along with DMA mechanisms for high performance data transfer.

The integrated GbE controller features are:

- Network Features
  - — Compliant with the 1 GB/s Ethernet 802.3, 802.3u, 802.3ab specifications
  - — Multi-speed operation: 10/100/1000 Mbps
  - — Full-duplex operation at 10/100/1000 Mbps: Half-duplex at 10/100 Mbps
  - — Flow control support compliant with the 802.3X specification

  - — VLAN support compliant with the 802.3q specification
  - — MAC address filters: perfect match unicast filters; multicast hash filtering, broadcast filter and promiscuous mode
  - — PCI Express/SMLink interface to GbE PHYs
- Host Interface Features
  - — 64-bit address master support for systems using more than 4 GB of physical memory
  - — Programmable host memory receive buffers (256 bytes to 16 KB)
  - — Intelligent interrupt generation features to enhance driver performance
  - — Descriptor ring management hardware for transmit and receive
  - — Software controlled reset (resets everything except the configuration space)
  - — Message Signaled Interrupts
- Performance Features
  - — Configurable receive and transmit data FIFO, programmable in 1 KB increments
  - — TCP segmentation off loading features
  - — Fragmented UDP checksum off load for packet reassembly
  - — IPv4 and IPv6 checksum off load support (receive, transmit, and large send)
  - — Split header support to eliminate payload copy from user space to host space
  - — Receive Side Scaling (RSS) with two hardware receive queues
  - — Supports 9018 bytes of jumbo packets
  - — Packet buffer size 32 KB
  - — TimeSync off load compliant with 802.1as specification
  - — Platform time synchronization
- Power Management Features
  - — Magic Packet* wake-up enable with unique MAC address
  - — ACPI register set and power down functionality supporting D0 and D3 states
  - — Full wake up support (APM, ACPI)
  - — MAC power down at Sx, DM-Off with and without WoL
  - — Auto connect battery saver at S0 no link and Sx no link
  - — Energy Efficient Ethernet (EEE) support
  - — Latency Tolerance Reporting (LTR)
  - — ARP and ND proxy support through LAN Connected Device proxy
  - — Wake on LAN (WoL) from Deep Sx
  - — Windows* InstantGo* Support

19.7.1 GbE PCI Express Bus Interface

The GbE controller has a PCI Express interface to the host processor and host memory. The following sections detail the bus transactions.

19.7.1.1 Transaction Layer

The upper layer of the host architecture is the transaction layer. The transaction layer connects to the device GbE controller using an implementation specific protocol. Through this GbE controller-to-transaction-layer protocol, the application-specific parts of the device interact with the subsystem and transmit and receive requests to or from the remote agent, respectively.

19.7.1.2 Data Alignment

19.7.1.2.1 4-KB Boundary

PCI requests must never specify an address/length combination that causes a memory space access to cross a 4-KB boundary. It is hardware's responsibility to break requests into 4 KB-aligned requests (if needed). This does not pose any requirement on software. However, if software allocates a buffer across a 4-KB boundary, hardware issues multiple requests for the buffer. Software should consider aligning buffers to a 4-KB boundary in cases where it improves performance. The alignment to the 4-KB boundaries is done by the GbE controller. The transaction layer does not do any alignment according to these boundaries.

19.7.1.2.2 PCI Request Size

PCI requests are 128 bytes or less and are aligned to make better use of memory controller resources. Writes, however, can be on any boundary and can cross a 64-byte alignment boundary.

19.7.1.3 Configuration Request Retry Status

The integrated GbE controller might have a delay in initialization due to an NVM read. If the NVM configuration read operation is not completed and the device receives a configuration request, the device responds with a configuration request retry completion status to terminate the request, and thus effectively stalls the configuration request until such time that the sub-system has completed local initialization and is ready to communicate with the host.

19.7.2 Error Events and Error Reporting

19.7.2.1 Completer Abort Error Handling

A received request that violates the LAN Controller programing model will be discarded, for non posted transactions an unsuccessful completion with CA completion status will be returned. For posted transactions if both SERR# enable and URRE# enable are enabled, the LAN Controller will assert SERR#.

19.7.2.2 Unsupported Request Error Handling

A received unsupported request to the LAN Controller will be discarded, for non posted transactions an unsuccessful completion with UR completion status will be returned. The URD bit will be set in ECTL register, If both SERR# enable and URRE# enable are enabled, the LAN Controller will assert SERR#. For posted transactions if both SERR# enable and URRE# enable are enabled, the LAN Controller will assert SERR#.

19.7.3 Ethernet Interface

The integrated GbE controller provides a complete CSMA/CD function supporting IEEE 802.3 (10 Mbps), 802.3u (100 Mbps) implementations. It also supports the IEEE 802.3z and 802.3ab (1000 Mbps) implementations. The device performs all of the functions required for transmission, reception, and collision handling called out in the standards.

The mode used to communicate between the PCH and the Intel® i219 Gigabit Ethernet LAN Connect Device supports 10/100/1000 Mbps operation, with both half- and full-duplex operation at 10/100 Mbps, and full-duplex operation at 1000 Mbps.

19.7.3.1 Intel® i219 Gigabit Ethernet LAN Connect Device

The integrated GbE controller and the Intel® i219 Gigabit Ethernet LAN Connect Device communicate through the PCIe* and SMLink0 interfaces. All integrated GbE controller configuration is performed using device control registers mapped into system memory or I/O space. The Platform LAN PHY is configured using the PCI Express or SMLink0 interface.

The integrated GbE controller supports various modes as listed in the following table.

Table 19-5. LAN Mode Support

Mode	System State	Interface Active	Connections
Normal 10/100/1000 Mbps	S0	PCI Express or SMLink0[1]	Intel® i219 Gigabit Ethernet LAN Connect Device
Manageability and Remote Wake-up	Sx	SMLink0	Intel® i219 Gigabit Ethernet LAN Connect Device
Note: [1]GbE operation is not supported in Sx state.			

19.7.4 PCI Power Management

The integrated GbE controller supports the Advanced Configuration and Power Interface specification as well as Advanced Power Management. This enables the network-related activity (using an internal host wake signal) to wake up the host. For example, from Sx (S3–S5) and Deep Sx to S0.

Note: The Intel® i219 Gigabit Ethernet LAN Connect Device must be powered during the Deep Sx state in order to support host wake up from Deep Sx. GPD_2_LAN_WAKE# on the PCH must be configured to support wake from Deep Sx and must be connected to LANWAKE_N on the Platform LAN Connect Device. The SLP_GBE# signal must be driven high (de-asserted) in the Deep Sx state to maintain power to the Platform LAN Connect Device.

The integrated GbE controller contains power management registers for PCI and supports D0 and D3 states. PCIe* transactions are only allowed in the D0 state, except for host accesses to the integrated GbE controller's PCI configuration registers.

19.8 GbE Configuration Registers Summary

Table 19-6. Summary of GbE Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	GbE Vendor and Device Identification Register (GBE_VID_DID)—Offset 0h	156F8086h
4h	7h	PCI Command and Status Register (PCICMD_STS)—Offset 4h	100000h
8h	Bh	Revision Identification and Class Code Register (RID_CC)—Offset 8h	2000000h
Ch	Fh	Cache Line Size Primary Latency Timer and Header Type Register (CLS_PLT_HEADTYP)—Offset Ch	0h
10h	13h	Memory Base Address Register A (MBARA)—Offset 10h	0h
2Ch	2Fh	Subsystem Vendor and Subsystem ID (DMI_CONFIG11)—Offset 2Ch	8086h
30h	33h	Expansion ROM Base Address Register (ERBA)—Offset 30h	0h
34h	37h	Capabilities List Pointer Register (CAPP)—Offset 34h	C8h
3Ch	3Fh	Interrupt Information and Maximum Latency/Minimum Grant Register (INTR_MLMG)—Offset 3Ch	100h
A0h	A3h	LAN Disable Control (LANDISCTRL)—Offset A0h	0h
A4h	A7h	Lock LAN Disable (LOCKLANDIS)—Offset A4h	0h
A8h	ABh	System Time Control High Register (LTRCAP)—Offset A8h	0h
C8h	CBh	Capabilities List and Power Management Capabilities Register (CLIST1_PMC)—Offset C8h	23D001h
CCh	CFh	PCI Power Management Control Status and Data Register (PMCS_DR)—Offset CCh	0h
D0h	D3h	Capabilities List 2 and Message Control Register (CLIST2_MCTL)—Offset D0h	80E005h
D4h	D7h	Message Address Low Register (MADDL)—Offset D4h	0h
D8h	DBh	Message Address High Register (MADDH)—Offset D8h	0h
DCh	DFh	Message Data Register (MDAT)—Offset DCh	0h

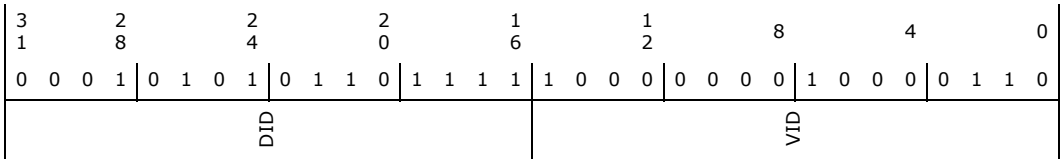
19.8.1 GbE Vendor and Device Identification Register (GBE_VID_DID)—Offset 0h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 6

Default: 156F8086h



Bit Range	Default and Access	Field Name (ID): Description
31:16	156Fh RW/V	**Device ID (DID):** This is a 16-bit value assigned to the PCH Gigabit LAN controller. The field may be auto-loaded from the NVM word 0Dh during initialization time depending on the "Load Vendor/ Device ID" bit field in NVM word 0Ah.
15:0	8086h RW/V	**Vendor ID (VID):** This is a 16-bit value assigned to Intel. The field may be auto-loaded from the NVM at address 0Dh during INIT time depending on the "Load Vendor/Device ID" bit field in NVM word 0Ah with a default value of 8086h.

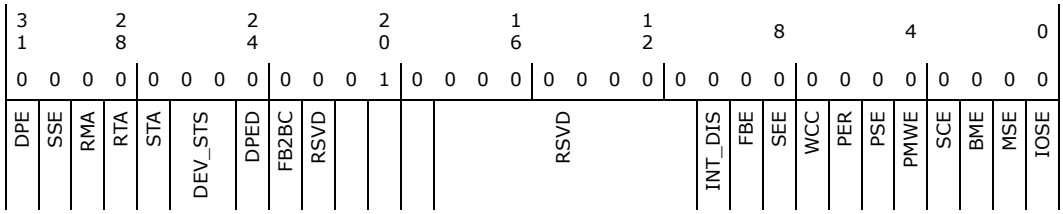
19.8.2 PCI Command and Status Register (PCICMD_STS)—Offset 4h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 6

Default: 100000h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RW/V	**Detected Parity Error (DPE):** 0 = No parity error detected. 1 = Set when the Gb LAN controller receives a command or data from the backbone with a parity error. This is set even if PCIMD.PER (D25:F0, bit 6) is not set.
30	0h RW/V	**Signaled System Error (SSE):** 0 = No system error signaled. 1 = Set when the Gb LAN controller signals a system error to the internal SERR# logic.
29	0h RW/V	**Received Master Abort (RMA):** 0 = Root port has not received a completion with unsupported request status from the backbone. 1 = Set when the GbE LAN controller receives a completion with unsupported request status from the backbone.
28	0h RW/V	**Received Target Abort (RTA):** 0 = Root port has not received a completion with completer abort from the backbone. 1 = Set when the Gb LAN controller receives a completion with completer abort from the backbone.
27	0h RW/V	**Signaled Target Abort (STA):** 0 = No target abort received. 1 = Set whenever the Gb LAN controller forwards a target abort received from the downstream device onto the backbone.
26:25	0h RW/V	**DEVSEL# Timing Status (DEV_STS):** Hardwired to 0.
24	0h RW/V	**Master Data Parity Error Detected (DPED):** 0 = No data parity error received. 1 = Set when the Gb LAN Controller receives a completion with a data parity error on the backbone and PCIMD.PER (D25:F0, bit 6) is set.
23	0h RW/V	**Fast Back to Back Capable (FB2BC):** Hardwired to 0.
22	0h RO	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
21	0h RW/V	**66 MHz Capable:** Hardwired to 0.
20	1h RW/V	**Capabilities List:** Hardwired to 1. Indicates the presence of a capabilities list.
19	0h RW/V	**Interrupt Status:** Indicates status of hot-plug and power management interrupts on the root port that result in INTx# message generation. 0 = Interrupt is de-asserted. 1 = Interrupt is asserted. This bit is not set if MSI is enabled. If MSI is not enabled, this bit is set regardless of the state of PCICMD.Interrupt Disable bit (D25:F0:04h:bit 10).
18:11	0h RO	Reserved.
10	0h RW	**Interrupt Disable (INT_DIS):** This disables pin-based INTx# interrupts on enabled hot-plug and power management events. This bit has no effect on MSI operation. 0 = Internal INTx# messages are generated if there is an interrupt for hot-plug or power management and MSI is not enabled. 1 = Internal INTx# messages will not be generated. This bit does not affect interrupt forwarding from devices connected to the root port. Assert_INTx and de-assert_INTx messages will still be forwarded to the internal interrupt controllers if this bit is set.
9	0h RW/V	**Fast Back to Back Enable (FBE):** Hardwired to 0.
8	0h RW	**SERR# Enable (SEE):** 0 = Disable 1 = Enables the Gb LAN controller to generate an SERR# message when PSTS.SSE is set.
7	0h RW/V	**Wait Cycle Control (WCC):** Hardwired to 0.
6	0h RW	**Parity Error Response (PER):** 0 = Disable. 1 = Indicates that the device is capable of reporting parity errors as a master on the backbone.
5	0h RW/V	**Palette Snoop Enable (PSE):** Hardwired to 0.
4	0h RW/V	**Postable Memory Write Enable (PMWE):** Hardwired to 0.
3	0h RW/V	**Special Cycle Enable (SCE):** Hardwired to 0.
2	0h RW	**Bus Master Enable (BME):** 0 = Disable. All cycles from the device are master aborted 1 = Enable. Allows the root port to forward cycles onto the backbone from a Gigabit LAN device.
1	0h RW	**Memory Space Enable (MSE):** 0 = Disable. Memory cycles within the range specified by the memory base and limit registers are master aborted on the backbone. 1 = Enable. Allows memory cycles within the range specified by the memory base and limit registers can be forwarded to the Gigabit LAN device.
0	0h RW/V	**I/O Space Enable (IOSE):** This bit controls access to the I/O space registers. 0 = Disable. I/O cycles within the range specified by the I/O base and limit registers are master aborted on the backbone. 1 = Enable. Allows I/O cycles within the range specified by the I/O base and limit registers can be forwarded to the Gigabit LAN device.

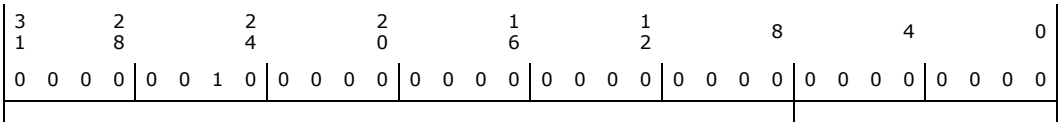
19.8.3 Revision Identification and Class Code Register (RID_CC)—Offset 8h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 6

Default: 2000000h



Bit Range	Default and Access	Field Name (ID): Description
31:8	20000h RW/V	**Class Code:** Identifies the device as an Ethernet Adapter. 020000h = Ethernet Adapter.
7:0	0h RW/V	**Revision ID:** See Table 2-1, "Intel® C620 Series ChipsetDevice and Revision ID Table" on page 38 for the value of the RID Register.

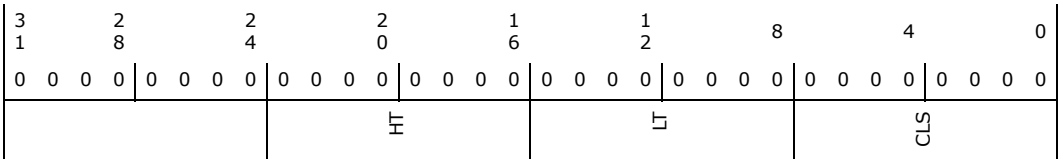
19.8.4 Cache Line Size Primary Latency Timer and Header Type Register (CLS_PLT_HEADTYP)—Offset Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 6

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RW/V	**Reserved**
23:16	0h RW/V	**Header Type (HT):** 00h = Indicates this is a single function device.
15:8	0h RW/V	**Latency Timer (LT):** Hardwired to 0.
7:0	0h RW/V	**Cache Line Size (CLS):** This field is implemented by PCI devices as a read/write field for legacy compatibility purposes but has no impact on any device functionality.

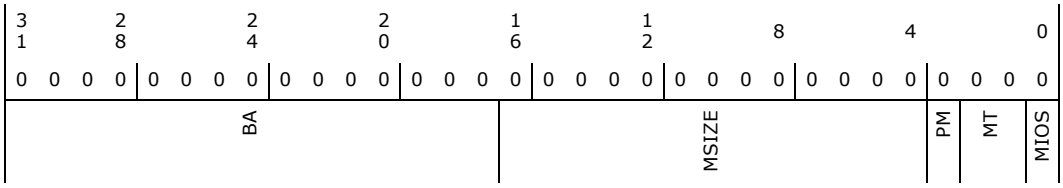
19.8.5 Memory Base Address Register A (MBARA)—Offset 10h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 6

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:17	0h RW	**Base Address (BA):** Software programs this field with the base address of this region.
16:4	0h RW/V	**Memory Size (MSIZE):** Memory size is 128KB.
3	0h RW/V	**Prefetchable Memory (PM):** The GbE LAN controller does not implement prefetchable memory.
2:1	0h RW/V	**Memory Type (MT):** Set to 00b indicating a 32-bit BAR.
0	0h RW/V	**Memory/I/O Space (MIOS):** Set to 0 indicating a Memory Space BAR.

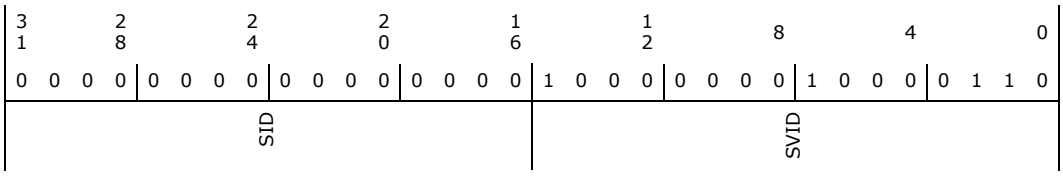
19.8.6 Subsystem Vendor and Subsystem ID (DMI_CONFIG11)—Offset 2Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 6

Default: 8086h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RW/V	**Subsystem ID (SID):** This value may be loaded automatically from the NVM Word 0Bh upon power up or reset depending on the "Load Subsystem ID" bit field in NVM word 0Ah with a default value of 0000h. This value is loadable from NVM word location 0Ah.
15:0	8086h RW/V	**Subsystem Vendor ID (SVID):** This value may be loaded automatically from the NVM Word 0Ch upon power up or reset depending on the "Load Subsystem ID" bit field in NVM word 0Ah. A value of 8086h is default for this field upon power up if the NVM does not respond or is not programmed. All functions are initialized to the same value.

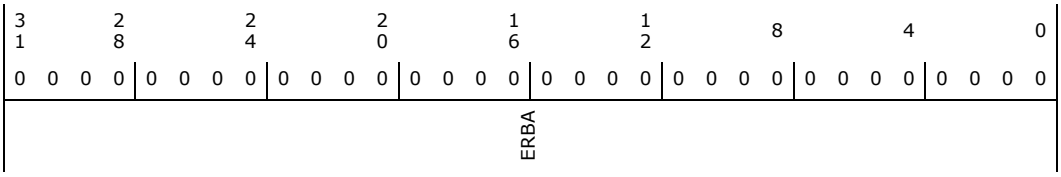
19.8.7 Expansion ROM Base Address Register (ERBA)—Offset 30h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 6

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/V	**Expansion ROM Base Address (ERBA):** This register is used to define the address and size information for boot-time access to the optional FLASH memory. If no Flash memory exists, this register reports 00000000h.

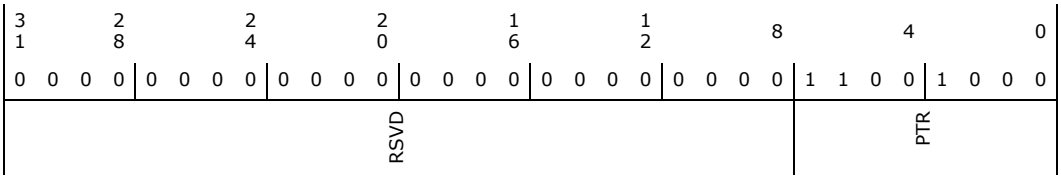
19.8.8 Capabilities List Pointer Register (CAPP)—Offset 34h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 6

Default: C8h



Bit Range	Default and Access	Field Name (ID): Description
31:8	0h RO	Reserved.
7:0	C8h RW/V	**Capabilities Pointer (PTR):** Indicates that the pointer for the first entry in the capabilities list is at C8h in configuration space.

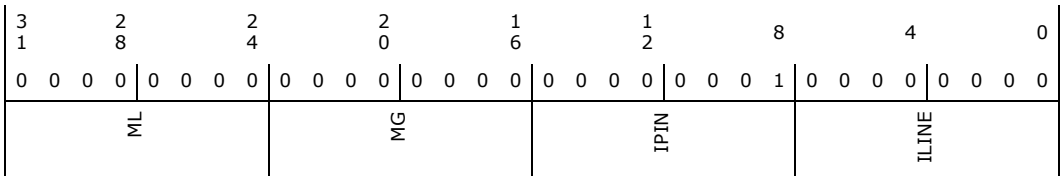
19.8.9 Interrupt Information and Maximum Latency/Minimum Grant Register (INTR_MLMG)—Offset 3Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 6

Default: 100h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RW/V	**Maximum Latency (ML):** Not used. Hardwired to 00h.
23:16	0h RW/V	**Minimum Grant (MG):** Not used. Hardwired to 00h.
15:8	1h RW/V	**Interrupt Pin (IPIN):** Indicates the interrupt pin driven by the GbE LAN controller. 01h = The GbE LAN controller implements legacy interrupts on INTA.
7:0	0h RW	**Interrupt Line (ILINE):** Default = 00h. Software written value indicates which interrupt line (vector) the interrupt is connected. No hardware action is taken on this register.

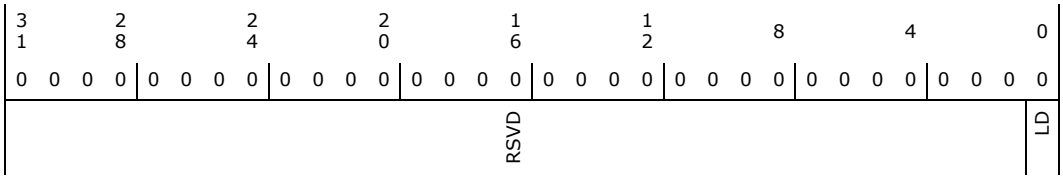
19.8.10 LAN Disable Control (LANDISCTRL)—Offset A0h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 6

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h RW	**LAN Disable (LD):** Setting this bit to 1 will disable the LAN Controller functionality.

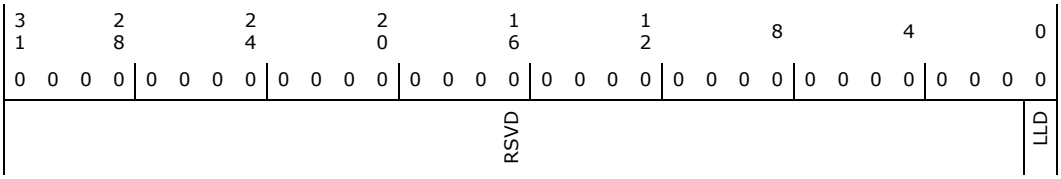
19.8.11 Lock LAN Disable (LOCKLANDIS)—Offset A4h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 6

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h RW	**Lock LAN Disable (LLD):** When set this bit blocks writes to the LANDISCTRL register. ***Note:*** Once set this bit will only be cleared on host reset.

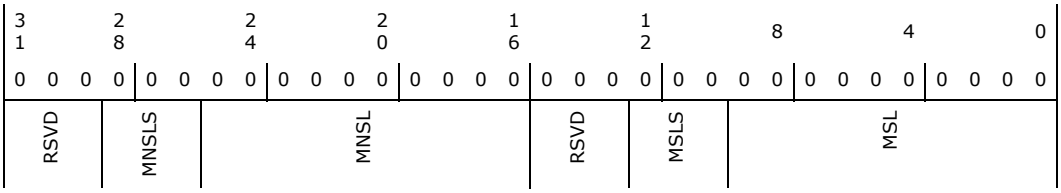
19.8.12 System Time Control High Register (LTRCAP)—Offset A8h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 6

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:29	0h RO	Reserved.
28:26	0h RW	**Maximum Non-Snoop Latency Scale (MNSLS):** Provides a scale for the value contained within the Maximum Non-Snoop Latency Value field. 000b = Value times 1 ns 001b = Value times 32 ns 010b = Value times 1,024 ns 011b = Value times 32,768 ns 100b = Value times 1,048,576 ns 101b = Value times 33,554,432 ns 110b–111b = Reserved
25:16	0h RW	**Maximum Non-Snoop Latency (MNSL):** Specifies the maximum non-snoop latency that a device is permitted to request. Software should set this to the platform's maximum supported latency or less. This field is also an indicator of the platforms maximum latency, should an endpoint send up LTR Latency Values with the Requirement bit not set.
15:13	0h RO	Reserved.
12:10	0h RW	**Maximum Snoop Latency Scale (MSLS):** Provides a scale for the value contained within the Maximum Snoop Latency Value field. 000b = Value times 1 ns 001b = Value times 32 ns 010b = Value times 1,024 ns 011b = Value times 32,768 ns 100b = Value times 1,048,576 ns 101b = Value times 33,554,432 ns 110b–111b = Reserved
9:0	0h RW	**Maximum Snoop Latency (MSL):** Specifies the maximum snoop latency that a device is permitted to request. Software should set this to the platform's maximum supported latency or less. This field is also an indicator of the platforms maximum latency, should an endpoint send up LTR Latency Values with the Requirement bit not set.

19.8.13 Capabilities List and Power Management Capabilities Register (CLIST1_PMC)—Offset C8h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 6

Default: 23D001h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1
PMES					D2S	D1S	AC				DSI	RSVD	PMEC	VER					NEXT					CID								

Bit Range	Default and Access	Field Name (ID): Description
31:27	0h RW/V	**PME_SUPPORT (PMES):** This five-bit field indicates the power states in which the function may assert PME#. It depend on PM Ena and AUX-PWR bits in word 0Ah in the NVM: Condition Functionality Value PM Ena=0 No PME at all states 0000b PM Ena and AUX-PWR=0 PME at D0 and $D3_{HOT}$ 01001b PM Ena and AUX-PWR=1 PME at D0, $D3_{HOT}$ and $D3_{COLD}$ 11001b These bits are not reset by Function Level Reset.
26	0h RW/V	**D2_SUPPORT (D2S):** The D2 state is not supported.
25	0h RW/V	**D1_SUPPORT (D1S):** The D1 state is not supported.
24:22	0h RW/V	**AUX_CURRENT (AC):** Required current defined in the Data register.
21	1h RW/V	**Device Specific Initialization (DSI):** Set to 1. The GbE LAN Controller requires its device driver to be executed following transition to the D0 un-initialized state.
20	0h RO	Reserved.
19	0h RW/V	**PME Clock (PMEC):** Hardwired to 0.
18:16	3h RW/V	**Version (VER):** Hardwired to 011b to indicate support for Revision 1.2 of the *PCI Power Management Specification*.
15:8	D0h RW/V	**Next Capability (NEXT):** Value of D0h indicates the location of the next pointer.
7:0	1h RW/V	**Capability ID (CID):** Indicates the linked list item is a PCI Power Management Register.

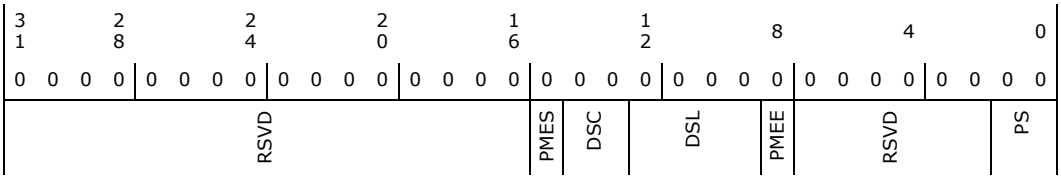
19.8.14 PCI Power Management Control Status and Data Register (PMCS_DR)—Offset CCh

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 6

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15	0h RW/V	**PME STATUS (PMES):** This bit is set to 1 when the function detects a wake-up event independent of the state of the PMEE bit. Writing a 1 will clear this bit.
14:13	0h RW/V	**DATA SCALE (DSC):** This field indicates the scaling factor to be used when interpreting the value of the Data register. For the GbE LAN and common functions this field equals 01b (indicating 0.1 watt units) if the PM is enabled in the NVM, and the Data_Select field is set to 0, 3, 4, 7, (or 8 for Function 0). Otherwise, it equals 00b. For the manageability functions, this field equals 10b (indicating 0.01 watt units) if the PM is enabled in the NVM, and the Data_Select field is set to 0, 3, 4, 7. Otherwise, it equals 00b.
12:9	0h RW	**Data Select (DSL):** This four-bit field is used to select which data is to be reported through the Data register (offset CFh) and Data_Scale field. These bits are writable only when Power Management is enabled using NVM. 0h = D0 Power Consumption 3h = D3 Power Consumption 4h = D0 Power Dissipation 7h = D3 Power Dissipation 8h = Common Power All other values are reserved.
8	0h RW	**PME Enable (PMEE):** If Power Management is enabled in the NVM, writing a 1 to this bit will enable Wakeup. If Power Management is disabled in the NVM, writing a 1 to this bit has no affect, and will not set the bit to 1. This bit is not reset by Function Level Reset.
7:2	0h RO	Reserved.
1:0	0h RW/V	**Power State (PS):** This field is used both to determine the current power state of the GbE LAN Controller and to set a new power state. The values are: 00 = D0 state (default) 01 = Ignored 10 = Ignored 11 = D3 state (Power Management must be enabled in the NVM or this cycle will be ignored).

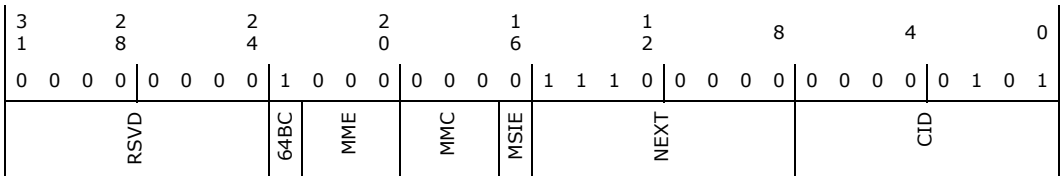
19.8.15 Capabilities List 2 and Message Control Register (CLIST2_MCTL)—Offset D0h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 6

Default: 80E005h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23	1h RW/V	**64-bit Capable (64BC):** Set to 1 to indicate that the GbE LAN Controller is capable of generating 64-bit message addresses.
22:20	0h RW/V	**Multiple Message Enable (MME):** Returns 000b to indicate that the GbE LAN controller only supports a single message.
19:17	0h RW/V	**Multiple Message Capable (MMC):** The GbE LAN controller does not support multiple messages.
16	0h RW	**Message Signal Interrupt Enable (MSIE):** 0 = MSI generation is disabled. 1 = The Gb LAN controller will generate MSI for interrupt assertion instead of INTx signaling.
15:8	E0h RW/V	**Next Capability (NEXT):** Value of E0h points to the Function Level Reset capability structure. These bits are not reset by Function Level Reset.
7:0	5h RW/V	**Capability ID (CID):** Indicates the linked list item is a Message Signaled Interrupt Register.

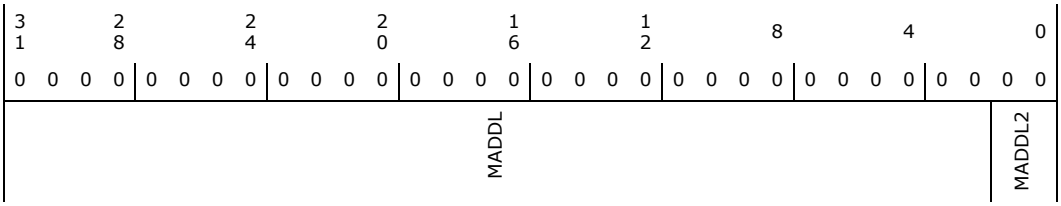
19.8.16 Message Address Low Register (MADDL)—Offset D4h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 6

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:2	0h RW/V	**Message Address Low (MADDL):** These bits combine with MADDL2 to create one 32 bit field which is written by the system to indicate the lower 32 bits of the address to use for the MSI memory write transaction.
1:0	0h RW/V	**Message Address Low 2 (MADDL2):** These bits combine with MADDL to create one 32 bit field which is written by the system to indicate the lower 32 bits of the address to use for the MSI memory write transaction. These lower two bits will always return 0.

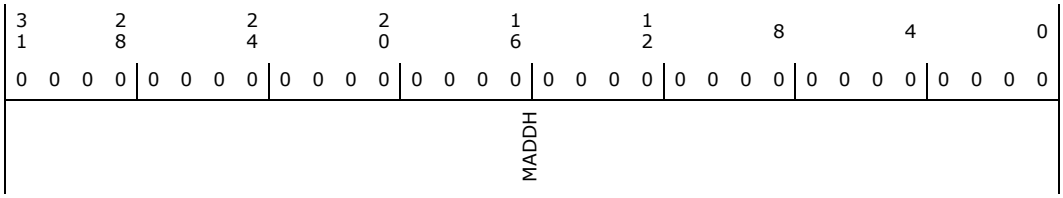
19.8.17 Message Address High Register (MADDH)—Offset D8h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 6

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Message Address High (MADDH):** Written by the system to indicate the upper 32 bits of the address to use for the MSI memory write transaction.

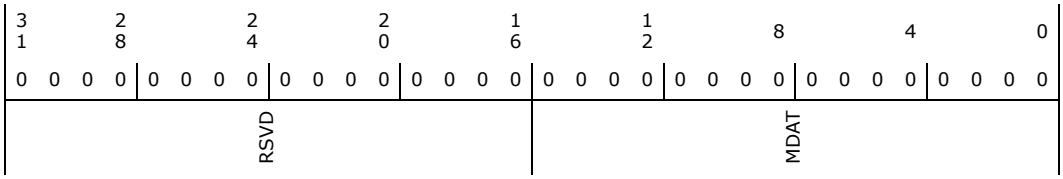
19.8.18 Message Data Register (MDAT)—Offset DCh

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 6

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:0	0h RW	**Message Data (MDAT):** Written by the system to indicate the lower 16 bits of the data written in the MSI memory write DWord transaction. The upper 16 bits of the transaction are written as 0000h.

19.9 GbE Memory Mapped I/O Registers Summary

Table 19-7. Summary of GbE Memory Mapped I/O Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	Gigabit Ethernet Capabilities and Status (GBECSR_00)—Offset 0h	0h
18h	1Bh	Gigabit Ethernet Capabilities and Status (GBECSR_18)—Offset 18h	0h
20h	23h	Gigabit Ethernet Capabilities and Status (GBECSR_20)—Offset 20h	10000000h
2Ch	2Fh	Gigabit Ethernet Capabilities and Status (GBECSR_2C)—Offset 2Ch	0h
F00h	F03h	Gigabit Ethernet Capabilities and Status (GBECSR_F00)—Offset F00h	0h
F10h	F13h	Gigabit Ethernet Capabilities and Status F10 (GBECSR_F10)—Offset F10h	Ch
5400h	5403h	Gigabit Ethernet Capabilities and Status (GBECSR_5400)—Offset 5400h	0h
5404h	5407h	Gigabit Ethernet Capabilities and Status (GBECSR_5404)—Offset 5404h	0h
5800h	5803h	Gigabit Ethernet Capabilities and Status (GBECSR_5800)—Offset 5800h	0h
5B54h	5B57h	Gigabit Ethernet Capabilities and Status (GBECSR_5B54)—Offset 5B54h	0h

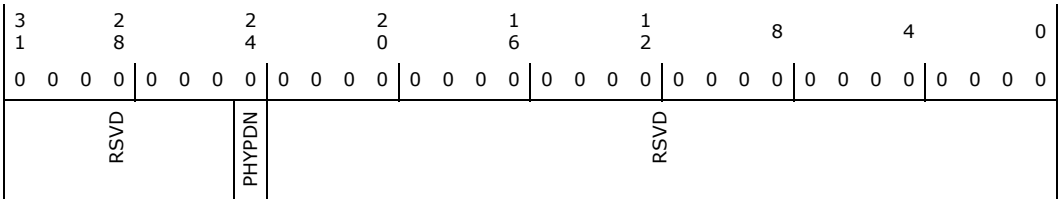
19.9.1 Gigabit Ethernet Capabilities and Status (GBECSR_00)—Offset 0h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 6

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:25	0h RO	Reserved.
24	0h RW	**PHY Power Down (PHYPDN):** When cleared (0b), the PHY power down setting is controlled by the internal logic of PCH.
23:0	0h RO	Reserved.

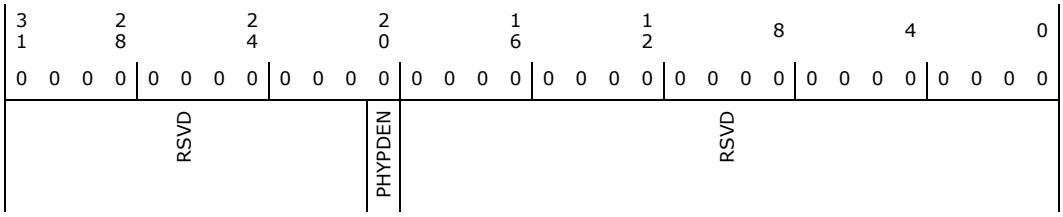
19.9.2 Gigabit Ethernet Capabilities and Status (GBECSR_18)—Offset 18h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 6

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:21	0h RO	Reserved.
20	0h RW	**PHY Power Down Enable (PHYPDEN):** When set, this bit enables the PHY to enter a low-power state when the LAN controller is at the DMOff/ D3 or with no WOL.
19:0	0h RO	Reserved.

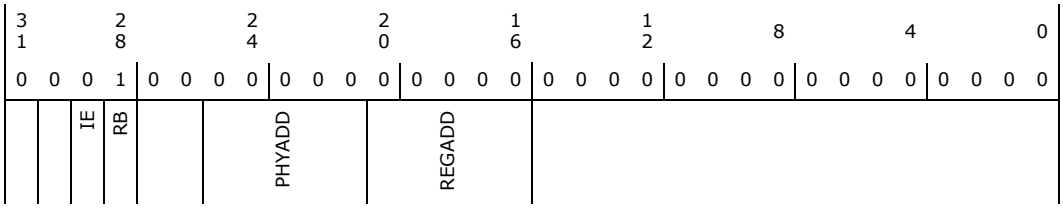
19.9.3 Gigabit Ethernet Capabilities and Status (GBECSR_20)—Offset 20h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 6

Default: 10000000h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RW/V	**Wait:** Set to 1 by the Gigabit Ethernet Controller to indicate that a PCI Express* to SMBus transition is taking place. The Intel ME/Host should not issue new MDIC transactions while this bit is set to 1. This bit is auto cleared by hardware after the transition has occurred.
30	0h RW/V	**Error:** Set to 1 by the Gigabit Ethernet Controller when it fails to complete an MDI read. Software should make sure this bit is clear before making an MDI read or write command.
29	0h RW/V	**Interrupt Enable (IE):** When set to 1 by software, it will cause an Interrupt to be asserted to indicate the end of an MDI cycle.
28	1h RW/V	**Ready Bit (RB):** Set to 1 by the Gigabit Ethernet Controller at the end of the MDI transaction. This bit should be reset to 0 by software at the same time the command is written.
27:26	0h RW/V	**MDI Type:** 01 = MDI Write 10 = MDI Read All other values are reserved.
25:21	0h RW/V	**LAN Connected Device Address (PHYADD)**
20:16	0h RW/V	**LAN Connected Device Register Address (REGADD)**
15:0	0h RW/V	**DATA:** In a Write command, software places the data bits and the MAC shifts them out to the LAN Connected Device. In a Read command, the MAC reads these bits serially from the LAN Connected Device and software can read them from this location.

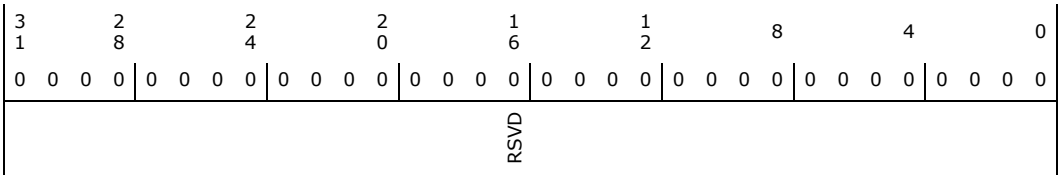
19.9.4 Gigabit Ethernet Capabilities and Status (GBECSR_2C)—Offset 2Ch

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 6

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RO	**Reserved**

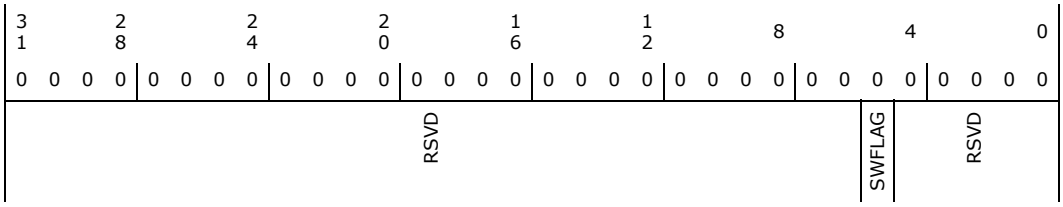
19.9.5 Gigabit Ethernet Capabilities and Status (GBECSR_F00)—Offset F00h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 6

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:6	0h RO	Reserved.
5	0h RW/V	**Software Semaphore FLAG (SWFLAG):** This bit is set by the device driver to gain access permission to shared CSR registers with the firmware and hardware.
4:0	0h RO	Reserved.

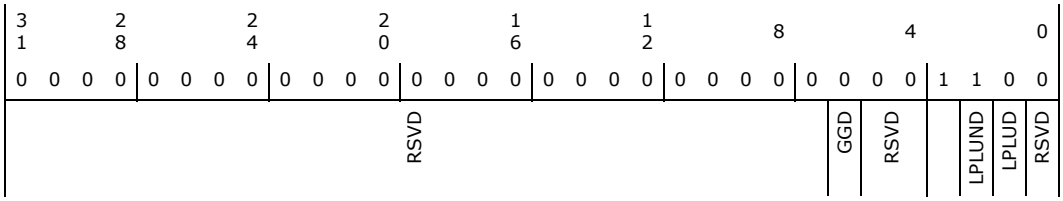
19.9.6 Gigabit Ethernet Capabilities and Status F10 (GBECSR_F10)—Offset F10h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 6

Default: Ch



Bit Range	Default and Access	Field Name (ID): Description
31:7	0h RO	Reserved.
6	0h RW	**Global GbE Disable (GGD):** Prevents the PHY from auto-negotiating 1000Mb/s link in all power states.
5:4	0h RO	Reserved.
3	1h RW	**GbE Disable at non D0a—:** Prevents the PHY from auto-negotiating 1000Mb/s link in all power states except D0a. This bit must be set since GbE is not supported in Sx states.
2	1h RW	**LPLU in non D0a (LPLUND):** Enables the PHY to negotiate for the slowest possible link in all power states except D0a.
1	0h RW	**LPLU in D0a (LPLUD):** Enables the PHY to negotiate for the slowest possible link in all power states. This bit overrides bit 2.
0	0h RO	Reserved.

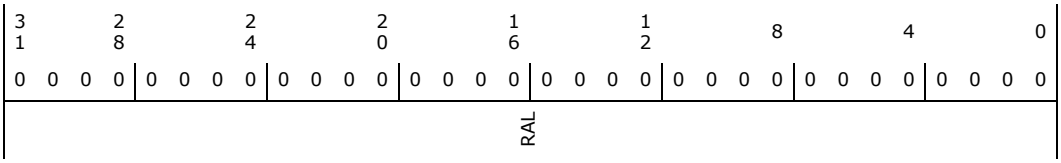
19.9.7 Gigabit Ethernet Capabilities and Status (GBECSR_5400)—Offset 5400h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 6

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Receive Address Low (RAL):** The lower 32 bits of the 48-bit Ethernet Address.

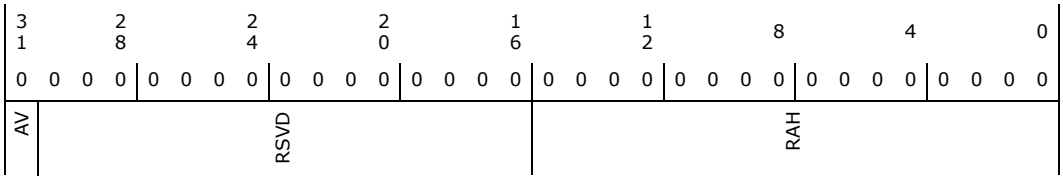
19.9.8 Gigabit Ethernet Capabilities and Status (GBECSR_5404)—Offset 5404h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 6

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RW	**Address Valid (AV)**
30:16	0h RO	Reserved.
15:0	0h RW	**Receive Address High (RAH):** The lower 16 bits of the 48-bit Ethernet Address.

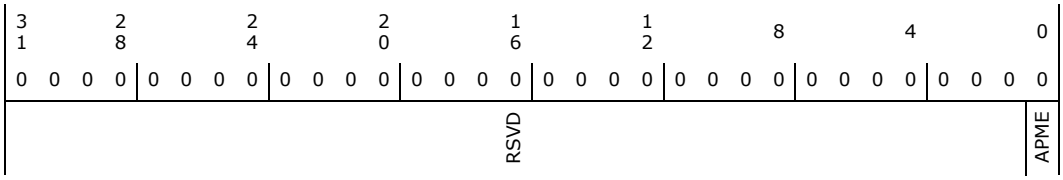
19.9.9 Gigabit Ethernet Capabilities and Status (GBECSR_5800)—Offset 5800h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 6

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h RW	**Advanced Power Management Enable (APME):** 1 = APM Wakeup is enabled 0 = APM Wakeup is disabled

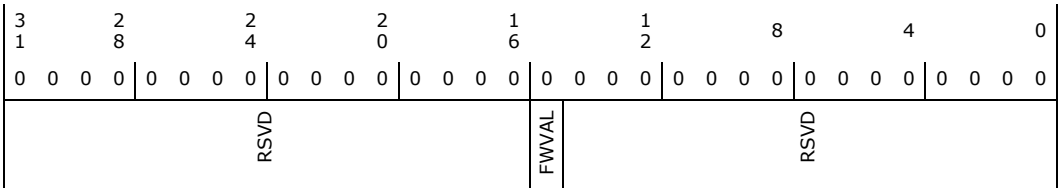
19.9.10 Gigabit Ethernet Capabilities and Status (GBECSR_5B54)—Offset 5B54h

Access Method

Type: MEM Register (Size: 32 bits)

Device: 31
Function: 6

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15	0h RW	**Firmware Valid Bit (FWVAL):** 1 = Firmware is ready 0 = Firmware is not ready
14:0	0h RO	Reserved.

§

20 Interrupt Interface

20.1 Acronyms

Acronyms	Description
AEOI	Automatic End Of Interrupt
APIC	Advanced Programmable Interrupt Controller
HPET	High Precision Event Timer
EOI	End of Interrupt
PIC	Programmable Interrupt Controller
PIRQ	Peripheral Interrupt Request

20.2 References

Specification	Location
None	

20.3 Overview

The interrupt controllers are used by the OS to dynamically route PCI interrupts to interrupt requests. Refer to the BIOS Writer's Guide for the additional information.

20.4 Signal Description

Name	Type	Description
GPP_A6_SERIRQ_ESPI_CS1#	I/O	Serial Interrupt Request ***Note:*** An external pull-up is required
GPP_A7_PIRQA#_ESPI_ALERT0#	I/OD	PCI Interrupt Request A ***Note:*** An external pull-up is required
GPP_G18_NMI#	OD	Active low signal driven by the PCH when it is generating an NMI#. This can be used by external components to determine that the PCH is generating an NMI#.
GPP_G19_SMI#	OD	Active low signal driven by the PCH when it is generating an SMI#. This can be used by external components to determine that the PCH is generating an SMI#.

20.5 Integrated Pull-Ups and Pull-Downs

None.

20.6 I/O Signal Planes and States

Signal Name	Power Plane	During Reset	Immediately after Reset	S3/S4/S5	Deep Sx
GPP_A6_SERIRQ_ESPI_CS1#	Primary	Undriven	Undriven	Undriven	OFF
GPP_A7_PIRQA#_ESPI_ALERT0#	Primary	Undriven	Undriven	Undriven	OFF
GPP_G18_NMI#	Primary	Undriven	Undriven	Undriven	OFF
GPP_G19_SMI#	Primary	Undriven	Undriven	Undriven	OFF

20.7 Functional Description

The PCH supports both APIC and PIC modes.

Interrupt sharing from the perspective of the interrupt controller that receives the interrupts is limited to IRQ 0-23.

- Shareable interrupts require the interrupt controller to track the Assert/De-assert Sideband message from each interrupt source. The interrupt controller achieves this through Source ID decode of the message.
- Maintains backwards compatibility with the prior generations where only the lower 24 IRQs are available to support interrupt sharing.
- Interrupts are dedicated and not shareable from the perspective of the interrupt controller for IRQ 24-119. In other words, not more than one interrupt initiator is allowed to be assigned to the same IRQ# for IRQ 24-119. For example, GPIO (multi-cause Interrupt Initiator) and LPSS (multi-function Interrupt Initiator) should not each generate Assert/De-assert IRQn that maps to IRQ24.
- Possible multi-cause interrupt initiators that map to IRQ24-119 are GPIO, eSPI, etc.

Interrupt Sharing Compliance Requirement for the interrupt initiator are as follows:

1. For multi-cause Initiators (Multiple Interrupt Cause from Single Source and Single SB Port ID, i.e., GPIO, eSPI): If more than 1interrupt cause has to use the same IRQ#, it has to be aggregated or guaranteed through BIOS/SW to assign a unique IRQ per interrupt cause.
2. For multi-function devices (one interrupt cause per source but many sources have behind single SB Port ID, i.e., LPSS): Again if sharing is needed, the interrupts have to be aggregated or guaranteed through SW to ensure a unique IRQ is assigned per interrupt cause.
3. IPs that have 1:1 mapping to the IRQ# such as eSPI and LPC is not impacted by this requirement. For eSPI, it is expected that the EC devices aggregate the interrupts before it is communicated to eSPI.
4. Single-cause or Single-function device behind a unique SB Port ID is not subjected to this requirement.

Only level-triggered interrupts can be shared. PCI interrupts are inherently shared on the board; these should, therefore, be programmed as level-triggered.

The following tables show the mapping of the various interrupts in Non-APIC and APIC modes.

Table 20-1. Interrupt Options - 8259 Mode

IRQ#	Pin	SERIRQ	PCI Message	Internal Modules
0	No	No	No	8254 Counter 0, HPET#0
1	No	Yes	No	Option for configurable sources including GPIO, eSPI and internal PCI/ACPI devices
2	No	No	No	8259 #2 cascade only
3:7	PIRQA	Yes	Yes	Option for configurable sources including PIRQx, GPIO, eSPI and internal PCI/ACPI devices
8	No	No	No	RTC, HPET#1
9:10	PIRQA	Yes	Yes	Option for configurable sources including PIRQx, GPIO, eSPI, internal PCI/ACPI devices, SCI and TCO.
11	PIRQA	Yes	Yes	Option for configurable sources including PIRQx, GPIO, eSPI, internal ACPI devices, SCI, TCO, HPET #2
12	PIRQA	Yes	Yes	Option for configurable sources including PIRQx, GPIO, eSPI, internal ACPI devices, HPET#3
13	No	No	Yes	Option for configurable sources including GPIO, eSPI, internal ACPI devices
14:15	PIRQA	Yes	Yes	Option for configurable sources including PIRQx, GPIO, eSPI and internal ACPI devices

Notes:
1. 8259 Interrupt Request Lines 0, 2 and 8 are non-shareable and dedicated. Only one interrupt source is allowed to use the Interrupt Request Line at any one time.
2. If an interrupt is used for PCI IRQ [A:H], SCI, or TCO, it should not be used for ISA-style interrupts (via SERIRQ).
3. In 8259 mode, PCI interrupts are mapped to IRQ3, 4, 5, 6, 7, 9, 10, 11, 12, 14, or 15. It can be programmed via 10.1.4 Interrupt Control Offset 60h-63h, 68h-6Bh.

Table 20-2. Interrupt Options - APIC Mode (Sheet 1 of 2)

IRQ#	Pin	SERIRQ	PCI Message	IRQ Sharable?	Internal Modules
0	No	No	No	No	Cascade from 8259 #1
1	No	Yes	No	Yes	Option for configurable sources including GPIO, eSPI, internal ACPI/ PCI devices
2	No	No	No	No	8254 Counter 0, HPET #0 (legacy mode)
3:7	No	Yes	No	Yes	Option for configurable sources including GPIO, eSPI, internal ACPI/ PCI devices
8	No	No	No	No	RTC, HPET #1 (legacy mode)
9:10	No	Yes	No	Yes	Option for configurable sources including GPIO, eSPI, internal ACPI/ PCI devices, SCI and TCO
11	No	Yes	No	Yes	Option for configurable sources including GPIO, eSPI, internal ACPI/ PCI devices, SCI, TCO, HPET #2
12	No	Yes	No	Yes	Option for configurable sources including GPIO, eSPI, internal ACPI/ PCI devices, HPET#3
13	No	No	No	Yes	Option for configurable sources including GPIO, eSPI and internal ACPI/PCI devices

Table 20-2. Interrupt Options - APIC Mode (Sheet 2 of 2)

IRQ#	Pin	SERIRQ	PCI Message	IRQ Sharable?	Internal Modules
14:15	No	Yes	No	Yes	Option for configurable sources including GPIO, eSPI and internal ACPI/PCI devices
16	PIRQA	PIRQA	Yes	Yes	Option for configurable sources including internal PIRQA, GPIO, eSPI and internal ACPI/PCI devices
17:19	No	PIRQ[B-D]	Yes	Yes	Option for configurable sources including internal PIRQ[B-D], GPIO, eSPI and internal ACPI/PCI devices
20:23	No	No	No	Yes	Option for configurable sources including internal PIRQ[E-H], GPIO, eSPI, SCI, TCO, internal ACPI/PCI devices and HPET
24:119	No	No	No	No	Option for configurable sources including GPIO, eSPI and internal ACPI/PCI devices

Notes:
1. Interrupts 24 through 119 are dedicated and not shareable from the perspective of the Interrupt Controller. Not more than one interrupt source is allowed to be assigned to the same IRQ#. For example, GPIO and LPSS should not generate Assert/Deassert_IRQn that maps to IRQ24. Although dedicated, Interrupts 24 through 119 can be configured to be level or edge-triggered.
2. If an interrupt is used for PCI IRQ [A:H], SCI, or TCO, it should not be used for ISA-style interrupts (via SERIRQ).
3. In APIC mode, the PCI interrupts [A:H] are directly mapped to IRQ[16:23].
4. When programming the polarity of internal interrupt sources on the APIC, interrupts 0 through 15, and 24 through 119 receive active-high internal interrupt sources; interrupts 16 through 23 receive active-low internal interrupt sources.
5. PIRQA is muxed with GPIO pins for assertion by external devices. Interrupt PIRQA will not be exposed if they are configured as GPIOs. When configured as GPIO pin, the internal PIRQA# is delivered internally to internal interrupt controller.
6. The internal ACPI/PCI devices refer to PCI/PCIe devices configured to the ACPI or PCI function mode. If in ACPI function mode, the device interrupt is map directly to one of the available IRQ. If in PCI function mode, the device interrupt is map to INT[A-D] and then to the IRQ before these devices issue the Interrupt Message using Assert/Deassert_IRQn.
7. PCI Message refers to the downstream Assert/Deassert_INT[A-D] messages forwarded from the CPU complex.

The following signals are associated with the interrupt logic.

Table 20-3. Interrupt Logic Signals

Signal Name	C3	S1-D	S1-M	S3	S5
SERIRQ	Can be running	Tri-State (high)	Tri-State (high)	Off	Off
PIRQA#	Can go active	Tri-State (high)	Tri-State (high)	Off	Off

20.7.1 8259 Interrupt Controllers

The ISA-compatible interrupt controller incorporates the functionality of two 8259 interrupt controllers. The following table shows how the cores are connected.

Table 20-4. Interrupt Controllers PIC

8259	8259 Input	Typical Interrupt Source	Connected Pin / Function
Master	0	Internal	Internal Timer / Counter 0 output or Multimedia Timer #0
	1	Keyboard	IRQ1 via SERIRQ. Option for configurable sources including eSPI, GPIO, internal ACPI devices.
	2	Internal	Slave Controller INTR output
	3	Serial Port A	IRQ3 from configurable sources including PIRQx, SERIRQ, eSPI, GPIO, internal ACPI devices.
	4	Serial Port B	IRQ4 from configurable sources including PIRQx, SERIRQ, eSPI, GPIO, internal ACPI devices.
	5	Parallel Port / Generic	IRQ5 from configurable sources including PIRQx, SERIRQ, eSPI, GPIO, internal ACPI devices.
	6	Floppy Disk	IRQ6 from configurable sources including PIRQx, SERIRQ, eSPI, GPIO, internal ACPI devices.
	7	Parallel Port / Generic	IRQ7 from configurable sources including PIRQx, SERIRQ, eSPI, GPIO, internal ACPI devices.
Slave	0	Real Time Clock	Inverted IRQ8# from internal RTC or Multimedia Timer #1
	1	Generic	IRQ9 from configurable sources including PIRQx, SERIRQ, eSPI, GPIO, internal ACPI devices, SCI, TCO.
	2	Generic	IRQ10 from configurable sources including PIRQx, SERIRQ, eSPI, GPIO, internal ACPI devices, SCI, TCO.
	3	Generic	IRQ11 from configurable sources including PIRQx, SERIRQ, eSPI, GPIO, internal ACPI devices, SCI, TCO or HPET #2.
	4	PS/2 Mouse	IRQ12 from configurable sources including PIRQx, SERIRQ, eSPI, GPIO, internal ACPI devices, SCI, TCO or HPET #3.
	5	Internal	IRQ13 from configurable sources including PIRQx, eSPI, GPIO, internal ACPI devices.
	6	Internal	IRQ14 from configurable sources including PIRQx, SERIRQ, eSPI, GPIO, internal ACPI devices.
	7	Internal	IRQ15 from configurable sources including PIRQx, SERIRQ, eSPI, GPIO, internal ACPI devices.

The slave controller is cascaded onto the master controller through master controller interrupt input 2. This means there are only 15 possible interrupts for PCH PIC.

Interrupts can individually be programmed to be edge or level triggered, except for IRQ0, IRQ1, IRQ2 and IRQ8# which always default to edge.

Active-low interrupt sources, such as the PIRQ#'s, are internally inverted before being sent to the PIC. In the following descriptions of the 8259's, the interrupt levels are in reference to the signals at the internal interface of the 8259's, after the required inversions have occurred. Therefore, the term "high" indicates "active", which means "low" on an originating PIRQ#.

20.7.2 Interrupt Handling

20.7.2.1 Generating Interrupts

The PIC interrupt sequence involves three bits, from the IRR, ISR, and IMR, for each interrupt level. These bits are used to determine the interrupt vector returned, and status of any other pending interrupts. Table 20-5 defines the IRR, ISR, and IMR.

Table 20-5. Interrupt Status Registers

Bit	Description
IRR	**Interrupt Request Register.** This bit is set on a low to high transition of the interrupt line in edge mode, and by an active high level in level mode. This bit is set whether or not the interrupt is masked. However, a masked interrupt will not generate INTR.
ISR	**Interrupt Service Register.** This bit is set, and the corresponding IRR bit cleared, when an interrupt acknowledge cycle is seen, and the vector returned is for that interrupt.
IMR	**Interrupt Mask Register.** This bit determines whether an interrupt is masked. Masked interrupts will not generate INTR.

20.7.2.2 Acknowledging Interrupts

The processor generates an interrupt acknowledge cycle that is translated by the host bridge into a PCI Interrupt Acknowledge Cycle to the PCH. The PIC translates this command into two internal INTA# pulses expected by the 8259 cores. The PIC uses the first internal INTA# pulse to freeze the state of the interrupts for priority resolution. On the second INTA# pulse, the master or slave sends the interrupt vector to the processor with the acknowledged interrupt code. This code is based upon bits [7:3] of the corresponding ICW2 register, combined with three bits representing the interrupt within that controller.

Table 20-6. Content of Interrupt Vector Byte

Master, Slave Interrupt	Bits [7:3]	Bits [2:0]
IRQ7,15	ICW2[7:3]	111
IRQ6,14		110
IRQ5,13		101
IRQ4,12		100
IRQ3,11		011
IRQ2,10		010
IRQ1,9		001
IRQ0,8		000

20.7.2.3 Hardware/Software Interrupt Sequence

1. One or more of the interrupt request lines are raised high in edge mode, or seen high in level mode, setting the corresponding IRR bit.
2. The PIC sends INTR active to the processor if an asserted interrupt is not masked.
3. The processor acknowledges the INTR and responds with an interrupt acknowledge cycle. The cycle is translated into a PCI interrupt acknowledge cycle by the host bridge. This command is broadcast over PCI by the PCH.
4. Upon observing its own interrupt acknowledge cycle on PCI, the PCH converts it into the two cycles that the internal 8259 pair can respond to. Each cycle appears as an interrupt acknowledge pulse on the internal INTA# pin of the cascaded interrupt controllers.

5. Upon receiving the first internally generated INTA# pulse, the highest priority ISR bit is set and the corresponding IRR bit is reset. On the trailing edge of the first pulse, a slave identification code is broadcast by the master to the slave on a private, internal three bit wide bus. The slave controller uses these bits to determine if it must respond with an interrupt vector during the second INTA# pulse.
6. Upon receiving the second internally generated INTA# pulse, the PIC returns the interrupt vector. If no interrupt request is present because the request was too short in duration, the PIC returns vector 7 from the master controller.
7. This completes the interrupt cycle. In AEOI mode the ISR bit is reset at the end of the second INTA# pulse. Otherwise, the ISR bit remains set until an appropriate EOI command is issued at the end of the interrupt subroutine.

20.7.3 Initialization Command Words

Before operation can begin, each 8259 must be initialized. In the PCH, this is a four byte sequence. The four initialization command words are referred to by their acronyms: ICW1, ICW2, ICW3, and ICW4.

The base address for each 8259 initialization command word is a fixed location in the I/O memory space: 20h for the master controller, and A0h for the slave controller.

20.7.3.1 ICW1

An I/O write to the master or slave controller base address with data bit 4 equal to 1 is interpreted as a write to ICW1. Upon sensing this write, the PCH's PIC expects three more byte writes to 21h for the master controller, or A1h for the slave controller, to complete the ICW sequence.

A write to ICW1 starts the initialization sequence during which the following automatically occur:

1. Following initialization, an interrupt request input must make a low-to-high transition to generate an interrupt.
2. The Interrupt Mask Register is cleared.
3. IRQ7 input is assigned priority 7.
4. The slave mode address is set to 7.
5. Special mask mode is cleared and Status Read is set to IRR.

20.7.3.2 ICW2

The second write in the sequence (ICW2) is programmed to provide bits [7:3] of the interrupt vector that will be released during an interrupt acknowledge. A different base is selected for each interrupt controller.

20.7.3.3 ICW3

The third write in the sequence (ICW3) has a different meaning for each controller.

- For the master controller, ICW3 is used to indicate which IRQ input line is used to cascade the slave controller. Within the PCH, IRQ2 is used. Therefore, Bit 2 of ICW3 on the master controller is set to a 1, and the other bits are set to 0s.

- For the slave controller, ICW3 is the slave identification code used during an interrupt acknowledge cycle. On interrupt acknowledge cycles, the master controller broadcasts a code to the slave controller if the cascaded interrupt won arbitration on the master controller. The slave controller compares this identification code to the value stored in its ICW3, and if it matches, the slave controller assumes responsibility for broadcasting the interrupt vector.

20.7.3.4 ICW4

The final write in the sequence (ICW4) must be programmed for both controllers. At the very least, Bit 0 must be set to a 1 to indicate that the controllers are operating in an Intel Architecture-based system.

20.7.4 Operation Command Words

These command words reprogram the Interrupt controller to operate in various interrupt modes.

- OCW1 masks and unmasks interrupt lines.
- OCW2 controls the rotation of interrupt priorities when in rotating priority mode, and controls the EOI function.
- OCW3 sets up ISR/IRR reads, enables/disables the Special Mask Mode (SMM), and enables/disables polled interrupt mode.

20.7.5 Modes of Operation

20.7.5.1 Fully-Nested Mode

In this mode, interrupt requests are ordered in priority from 0 through 7, with 0 being the highest. When an interrupt is acknowledged, the highest priority request is determined and its vector placed on the bus. Additionally, the ISR for the interrupt is set. This ISR bit remains set until: the processor issues an EOI command immediately before returning from the service routine; or if in AEOI mode, on the trailing edge of the second INTA#. While the ISR bit is set, all further interrupts of the same or lower priority are inhibited, while higher levels generate another interrupt. Interrupt priorities can be changed in the rotating priority mode.

20.7.5.2 Special Fully-Nested Mode

This mode is used in the case of a system where cascading is used, and the priority has to be conserved within each slave. In this case, the special fully-nested mode is programmed to the master controller. This mode is similar to the fully-nested mode with the following exceptions:

- When an interrupt request from a certain slave is in service, this slave is not locked out from the master's priority logic and further interrupt requests from higher priority interrupts within the slave are recognized by the master and initiate interrupts to the processor. In the normal-nested mode, a slave is masked out when its request is in service.
- When exiting the Interrupt Service Routine, software has to check whether the interrupt serviced was the only one from that slave. This is done by sending a Non-Specific EOI command to the slave and then reading its ISR. If it is 0, a Non-Specific EOI can also be sent to the master.

20.7.5.3 Automatic Rotation Mode (Equal Priority Devices)

In some applications, there are a number of interrupting devices of equal priority. Automatic rotation mode provides for a sequential 8-way rotation. In this mode, a device receives the lowest priority after being serviced. In the worst case, a device requesting an interrupt has to wait until each of seven other devices are serviced at most once.

There are two ways to accomplish automatic rotation using OCW2; the Rotation on Non-Specific EOI Command (R=1, SL=0, EOI=1) and the rotate in automatic EOI mode which is set by (R=1, SL=0, EOI=0).

20.7.5.4 Specific Rotation Mode (Specific Priority)

Software can change interrupt priorities by programming the bottom priority. For example, if IRQ5 is programmed as the bottom priority device, then IRQ6 is the highest priority device. The Set Priority Command is issued in OCW2 to accomplish this, where: R=1, SL=1, and LO–L2 is the binary priority level code of the bottom priority device.

In this mode, internal status is updated by software control during OCW2. However, it is independent of the EOI command. Priority changes can be executed during an EOI command by using the Rotate on Specific EOI Command in OCW2 (R=1, SL=1, EOI=1 and LO–L2=IRQ level to receive bottom priority.

20.7.5.5 Poll Mode

Poll mode can be used to conserve space in the interrupt vector table. Multiple interrupts that can be serviced by one Interrupt Service Routine do not need separate vectors if the service routine uses the poll command. Poll mode can also be used to expand the number of interrupts. The polling Interrupt Service Routine can call the appropriate service routine, instead of providing the interrupt vectors in the vector table. In this mode, the INTR output is not used and the microprocessor internal Interrupt Enable flip-flop is reset, disabling its interrupt input. Service to devices is achieved by software using a Poll command.

The Poll command is issued by setting P=1 in OCW3. The PIC treats its next I/O read as an interrupt acknowledge, sets the appropriate ISR bit if there is a request, and reads the priority level. Interrupts are frozen from the OCW3 write to the I/O read. The byte returned during the I/O read contains a 1 in Bit 7 if there is an interrupt, and the binary code of the highest priority level in bits [2:0].

20.7.5.6 Edge and Level Triggered Mode

In ISA systems this mode is programmed using Bit 3 in ICW1, which sets level or edge for the entire controller. In the PCH, this bit is disabled and a register for edge and level triggered mode selection, per interrupt input, is included. This is the Edge/Level control Registers ELCR1 and ELCR2.

If an ELCR bit is 0, an interrupt request will be recognized by a low-to-high transition on the corresponding IRQ input. The IRQ input can remain high without generating another interrupt. If an ELCR bit is 1, an interrupt request will be recognized by a high level on the corresponding IRQ input and there is no need for an edge detection. The interrupt request must be removed before the EOI command is issued to prevent a second interrupt from occurring.

In both the edge and level triggered modes, the IRQ inputs must remain active until after the falling edge of the first internal INTA#. If the IRQ input goes inactive before this time, a default IRQ7 vector is returned.

20.7.5.7 End Of Interrupt Operations

An EOI can occur in one of two fashions: by a command word write issued to the PIC before returning from a service routine, the EOI command; or automatically when AEOI bit in ICW4 is set to 1.

20.7.5.8 Normal End of Interrupt

In normal EOI, software writes an EOI command before leaving the Interrupt Service Routine to mark the interrupt as completed. There are two forms of EOI commands: Specific and Non-Specific. When a Non-Specific EOI command is issued, the PIC clears the highest ISR bit of those that are set to 1. Non-Specific EOI is the normal mode of operation of the PIC within the PCH, as the interrupt being serviced currently is the interrupt entered with the interrupt acknowledge. When the PIC is operated in modes that preserve the fully nested structure, software can determine which ISR bit to clear by issuing a Specific EOI. An ISR bit that is masked is not cleared by a Non-Specific EOI if the PIC is in the special mask mode. An EOI command must be issued for both the master and slave controller.

20.7.5.9 Automatic End of Interrupt Mode

In this mode, the PIC automatically performs a Non-Specific EOI operation at the trailing edge of the last interrupt acknowledge pulse. From a system standpoint, this mode should be used only when a nested multi-level interrupt structure is not required within a single PIC. The AEOI mode can only be used in the master controller and not the slave controller.

20.7.6 Masking Interrupts

20.7.6.1 Masking on an Individual Interrupt Request

Each interrupt request can be masked individually by the Interrupt Mask Register (IMR). This register is programmed through OCW1. Each bit in the IMR masks one interrupt channel. Masking IRQ2 on the master controller masks all requests for service from the slave controller.

20.7.6.2 Special Mask Mode

Some applications may require an Interrupt Service Routine to dynamically alter the system priority structure during its execution under software control. For example, the routine may wish to inhibit lower priority requests for a portion of its execution but enable some of them for another portion.

The special mask mode enables all interrupts not masked by a bit set in the Mask Register. Normally, when an Interrupt Service Routine acknowledges an interrupt without issuing an EOI to clear the ISR bit, the interrupt controller inhibits all lower priority requests. In the special mask mode, any interrupts may be selectively enabled by loading the Mask Register with the appropriate pattern. The special Mask Mode is set by OCW3.SSMM and OCW3.SMM being set, and cleared when OCW3.SSMM and OCW3.SMM are cleared.

20.7.7 Steering PCI Interrupts

PCH can be programmed to allow PIRQ[A:D]# to be internally routed to interrupts 3-7, 9-12, 14 or 15, through the PARC, PBRC, PCRC, PDRC, PERC, PFRC, PGRC, and PHRC registers in the chipset configuration section. One or more PIRQx# lines can be routed to the same IRQx input.

The PIRQx# lines are defined as active low, level sensitive. When PIRQx# is routed to specified IRQ line, software must change the corresponding ELCR1 or ELCR2 register to level sensitive mode. PCH will internally invert the PIRQx# line to send an active high level to the PIC. When a PCI interrupt is routed onto the PIC, the selected IRQ can no longer be used by an ISA device.

20.8 Advanced Programmable Interrupt Controller (APIC) (D31:F0)

In addition to the standard ISA-compatible PIC described in the previous section, the PCH incorporates the APIC. While the standard interrupt controller is intended for use in a uni-processor system, APIC can be used in either a uni-processor or multi-processor system.

20.8.1 Interrupt Handling

The I/O APIC handles interrupts very differently than the 8259. Briefly, these differences are:

- **Method of Interrupt Transmission.** The I/O APIC transmits interrupts through memory writes on the normal data path to the processor, and interrupts are handled without the need for the processor to run an interrupt acknowledge cycle.
- **Interrupt Priority.** The priority of interrupts in the I/O APIC is independent of the interrupt number. For example, interrupt 10 can be given a higher priority than interrupt 3.
- **More Interrupts.** The I/O APIC in the PCH supports a total of 24 interrupts.
- **Multiple Interrupt Controllers.** The I/O APIC architecture allows for multiple I/O APIC devices in the system with their own interrupt vectors.

20.8.2 Interrupt Mapping

The I/O APIC within the PCH supports 40 APIC interrupts. Each interrupt has its own unique vector assigned by software. The interrupt vectors are mapped as follows.

Table 20-7. APIC Interrupt Mapping[1] (Sheet 1 of 2)

IRQ #	Using SERIRQ	Direct from Pin	Using PCI Message	Internal Modules
0	No	No	No	Cascade from 8259 #1
1	Yes	No	Yes	
2	No	No	No	8254 Counter 0, HPET #0 (legacy mode)
3	Yes	No	Yes	
4	Yes	No	Yes	
5	Yes	No	Yes	GSPI, UART, I^2C, SDIO
6	Yes	No	Yes	GSPI, UART, I^2C, SDIO
7	Yes	No	Yes	GSPI, UART, I^2C, SDIO

Table 20-7. APIC Interrupt Mapping[1] (Sheet 2 of 2)

IRQ #	Using SERIRQ	Direct from Pin	Using PCI Message	Internal Modules
8	No	No	No	RTC, HPET #1 (legacy mode)
9	Yes	No	Yes	Option for SCI, TCO
10	Yes	No	Yes	Option for SCI, TCO
11	Yes	No	Yes	HPET #2, Option for SCI, TCO (Note 2)
12	Yes	No	Yes	HPET #3 (Note 3)
13	No	No	No	GSPI, UART, I^2C, SDIO
14	Yes	No	Yes	GPIO
15	Yes	No	Yes	GPIO
16	PIRQA#	PIRQA#[5]	Yes	Internal devices are routable.
17	PIRQB#	PIRQB#[5]		
18	PIRQC#	PIRQC#[5]		
19	PIRQD#	PIRQD#[5]		
20	N/A	PIRQE#[4]	Yes	Option for SCI, TCO, HPET #0, 1, 2, 3. Other internal devices are routable.
21	N/A	PIRQF#[4]		
22	N/A	PIRQG#[4]		
23	N/A	PIRQH#[4]		
24	No	PIRQI	No	
25	No	PIRQJ	No	
26	No	PIRQK	No	
27	No	PIRQL	No	
28	No	PIRQM	No	
29	No	PIRQN	No	
30	No	PIRQO	No	
31	No	PIRQP	No	
32	No	PIRQQ	No	
33	No	PIRQR	No	
34	No	PIRQS	No	
35	No	PIRQT	No	
36	No	PIRQU	No	
37	No	PIRQV	No	
38	No	PIRQW	No	
39	No	PIRQX	No	

Notes:

1. When programming the polarity of internal interrupt sources on the APIC, interrupts 0 through 15 receive active-high internal interrupt sources, while interrupts 16 through 23 receive active-low internal interrupt sources.
2. If IRQ 11 is used for HPET #2, software should ensure IRQ 11 is not shared with any other devices to ensure the proper operation of HPET #2. The PCH hardware does not prevent sharing of IRQ 11.
3. If IRQ 12 is used for HPET #3, software should ensure IRQ 12 is not shared with any other devices to ensure the proper operation of HPET #3. The PCH hardware does not prevent sharing of IRQ 12.
4. PIRQ[E:H] physical pins are not supported. The internal PIRQ[E:H]# are delivered internally to internal interrupt controller.
5. PIRQ[A:D] are Multiplexed with GPIO pins. Interrupts PIRQ[A:D] will not be exposed if they are configured as GPIOs. When configured as GPIO pins, the internal PIRQ[A:D]# are delivered internally to internal interrupt controller.

20.8.3 PCI/PCI Express Message-Based Interrupts

When external devices through PCI/PCI Express wish to generate an interrupt, they will send the message defined in the *PCI Express* Base Specification,* Revision 2.0 for generating INTA# – INTD#. These will be translated internal assertions/de-assertions of INTA# – INTD#.

20.8.4 IOxAPIC Address Remapping

To support Intel Virtualization Technology, interrupt messages are required to go through similar address remapping as any other memory request. Address remapping allows for domain isolation for interrupts, so a device assigned in one domain is not allowed to generate an interrupt to another domain.

The address remapping is based on the Bus: Device: Function field associated with the requests. The internal APIC is required to initiate the interrupt message using a unique Bus: Device: Function.

The PCH allows BIOS to program the unique Bus: Device: Function address for the internal APIC. This address field does not change the APIC functionality and the APIC is not promoted as a stand-alone PCI device. See Device 31: Function 0 Offset 6Ch for additional information.

20.8.5 External Interrupt Controller Support

The PCH supports external APICs off of PCI Express ports but does not support APICs on the PCI bus. The EOI special cycle is only forwarded to PCI Express ports.

20.9 Serial Interrupt

The PCH supports a serial IRQ scheme. This allows a single signal to be used to report interrupt requests. The signal used to transmit this information is shared between the PCH and all participating peripherals. The signal line, SERIRQ, is synchronous to 24 MHz CLKOUT_LPC, and follows the sustained tri-state protocol that is used by all PCI signals. This means that if a device has driven SERIRQ low, it will first drive it high synchronous to PCI clock and release it the following PCI clock. The serial IRQ protocol defines this sustained tri-state signaling in the following fashion:

- **S – Sample Phase,** Signal driven low
- **R** – **Recovery Phase,** Signal driven high
- **T** – **Turn-around Phase,** Signal released

The PCH supports a message for 21 serial interrupts. These represent the 15 ISA interrupts (IRQ0–1, 3–15), the four PCI interrupts, and the control signals SMI# and IOCHK#. The serial IRQ protocol does not support the additional APIC interrupts (20–23).

Note: IRQ14 and IRQ15 are special interrupts and maybe used by the GPIO controller when it is running GPIO driver mode. When the GPIO controller operates in GPIO driver mode, IRQ14 and IRQ15 shall not be utilized by the SERIRQ stream nor mapped to other interrupt sources, and instead come from the GPIO controller. If the GPIO controller is entirely in ACPI mode, these interrupts can be mapped to other devices accordingly.

20.9.1 Start Frame

The serial IRQ protocol has two modes of operation which affect the start frame. These two modes are: Continuous, where the PCH is solely responsible for generating the start frame; and Quiet, where a serial IRQ peripheral is responsible for beginning the start frame.

The mode that must first be entered when enabling the serial IRQ protocol is continuous mode. In this mode, the PCH asserts the start frame. This start frame is four, six or eight PCI clocks wide based upon the Serial IRQ Control Register, bits 1:0 at 64h in D31:F0 configuration space. This is a polling mode.

When the serial IRQ stream enters quiet mode (signaled in the Stop Frame), the SERIRQ line remains inactive and pulled up between the Stop and Start Frame until a peripheral drives the SERIRQ signal low. The PCH senses the line low and continues to drive it low for the remainder of the Start Frame. Since the first PCI clock of the start frame was driven by the peripheral in this mode, the PCH drives the SERIRQ line low for one PCI clock less than in continuous mode. This mode of operation allows for a quiet, and therefore lower power, operation.Data Frames

Once the Start frame has been initiated, all of the SERIRQ peripherals must start counting frames based on the rising edge of SERIRQ. Each of the IRQ/DATA frames has exactly three phases of one clock each:

- **Sample Phase—**During this phase, the SERIRQ device drives SERIRQ low if the corresponding interrupt signal is low. If the corresponding interrupt is high, then the SERIRQ devices tri-state the SERIRQ signal. The SERIRQ line remains high due to pull-up resistors (there is no internal pull-up resistor on this signal, an external pull-up resistor is required). A low level during the IRQ0–1 and IRQ2–15 frames indicates that an active-high ISA interrupt is not being requested, but a low level during the PCI INT[A:D], SMI#, and IOCHK# frame indicates that an active-low interrupt is being requested.
- **Recovery Phase—**During this phase, the device drives the SERIRQ line high if in the Sample Phase it was driven low. If it was not driven in the sample phase, it is tri-stated in this phase.
- **Turn-around Phase—**The device tri-states the SERIRQ line

20.9.2 Stop Frame

After all data frames, a Stop Frame is driven by the PCH. The SERIRQ signal is driven low by the PCH for two or three PCI clocks. The number of clocks is determined by the SERIRQ configuration register. The number of clocks determines the next mode.

Table 20-8. Stop Frame Explanation

Stop Frame Width	Next Mode
2 PCI clocks	**Quiet Mode.** Any SERIRQ device may initiate a Start Frame
3 PCI clocks	**Continuous Mode.** Only the host (the PCH) may initiate a Start Frame

20.9.3 Specific Interrupts Not Supported Using SERIRQ

There are three interrupts seen through the serial stream that are not supported by the PCH. These interrupts are generated internally, and are not sharable with other devices within the system. These interrupts are:

- IRQ0. Heartbeat interrupt generated off of the internal 8254 counter 0.
- IRQ8#. RTC interrupt can only be generated internally.
- IRQ13. Reserved internally.

The PCH ignores the state of these interrupts in the serial stream, and does not adjust their level based on the level seen in the serial stream.Data Frame Format

Table 20-9 shows the format of the data frames. For the PCI interrupts (A–D), the output from the PCH is AND'd with the PCI input signal. This way, the interrupt can be signaled using both the PCI interrupt input signal and using the SERIRQ signal (they are shared).

Table 20-9. Data Frame Format

Data Frame #	Interrupt	Clocks Past Start Frame	Comment
1	IRQ0	2	Ignored. IRQ0 can only be generated using the internal 8524
2	IRQ1	5	Before port 60h latch
3	SMI#	8	Causes SMI# if low. Will set the SERIRQ_SMI_STS bit.
4	IRQ3	11	
5	IRQ4	14	
6	IRQ5	17	
7	IRQ6	20	
8	IRQ7	23	
9	IRQ8	26	Ignored. IRQ8# can only be generated internally.
10	IRQ9	29	
11	IRQ10	32	
12	IRQ11	35	
13	IRQ12	38	Before port 60h latch
14	IRQ13	41	Ignored.
15	IRQ14	44	Not attached to GPIO logic
16	IRQ15	47	Not attached to GPIO logic
17	IOCHCK#	50	Same as ISA IOCHCK# going active
18	PCI INTA#	53	Drive PIRQA#
19	PCI INTB#	56	Drive PIRQB#
20	PCI INTC#	59	Drive PIRQC#
21	PCI INTD#	62	Drive PIRQD#

20.10 Interrupt Registers Summary

Table 20-10. Summary of Interrupt Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
20h	20h	Master Initialization Command Word 1 (MICW1)—Offset 20h	11h
20h	20h	Master Operational Control Word 2 (MOCW2)—Offset 20h	0h
20h	20h	Master Operational Control Word 3 (MOCW3)—Offset 20h	8h
21h	21h	Master Initialization Command Word 2 (MICW2)—Offset 21h	0h
21h	21h	Master Initialization Command Word 3 (MICW3)—Offset 21h	7h
21h	21h	Master Initialization Command Word 4 (MICW4)—Offset 21h	0h
21h	21h	Master Operational Control Word 1 (MOCW1)—Offset 21h	0h
A0h	A0h	Slave Initialization Command Word 1 (SICW1)—Offset A0h	11h
A0h	A0h	Slave Operational Control Word 2 (SOCW2)—Offset A0h	0h
A0h	A0h	Slave Operational Control Word 3 (SOCW3)—Offset A0h	8h
A1h	A1h	Slave Initialization Command Word 2 (SICW2)—Offset A1h	0h
A1h	A1h	Slave Initialization Command Word 3 (SICW3)—Offset A1h	7h
A1h	A1h	Slave Initialization Command Word 4 (SICW4)—Offset A1h	0h
A1h	A1h	Slave Operational Control Word 1 (SOCW1)—Offset A1h	0h
4D0h	4D0h	Master Edge/Level Control (ELCR1)—Offset 4D0h	0h
4D1h	4D1h	Slave Edge/Level Control (ELCR2)—Offset 4D1h	0h

20.10.1 Master Initialization Command Word 1 (MICW1)—Offset 20h

A write to Initialization Command Word 1 starts the interrupt controller initialization sequence, during which the following occurs:

1. The Interrupt Mask register is cleared.
2. IRQ7 input is assigned priority 7.
3. The slave mode address is set to 7.
4. Special Mask Mode is cleared and Status Read is set to IRR.

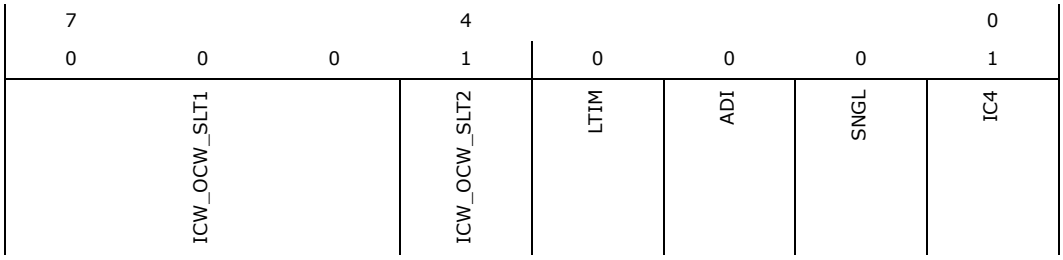
Once this write occurs, the controller expects writes to ICW2, ICW3, and ICW4 to complete the initialization sequence.

Access Method

Type: I/O Register
(Size: 8 bits)

Device:
Function:

Default: 11h



Bit Range	Default and Access	Field Name (ID): Description
7:5	0h WO	**ICW/OCW select (ICW_OCW_SLT1):** These bits are MCS-85 specific, and not needed. Should be programmed to 000
4	1h WO	**ICW/OCW select (ICW_OCW_SLT2):** This bit must be a 1 to select ICW1 and enable the ICW2, ICW3, and ICW4 sequence.
3	0h WO	**Edge/Level Bank Select (LTIM):** Disabled. Replaced by the edge/level triggered control registers (ELCR).
2	0h WO	**ADI (ADI):** Ignored for PCH. Should be programmed to 0.
1	0h WO	**Single or Cascade (SNGL):** Must be programmed to a 0 to indicate two controllers operating in cascade mode.
0	1h WO	**ICW4 Write Required (IC4):** This bit must be programmed to a 1 to indicate that ICW4 needs to be programmed.

20.10.2 Master Operational Control Word 2 (MOCW2)—Offset 20h

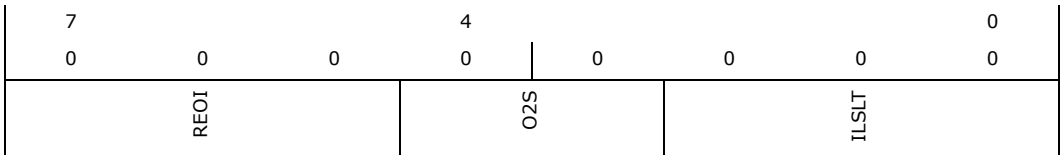
*address should be 20h
Following a part reset or ICW initialization, the controller enters the fully nested mode of operation. Non-specific EOI without rotation is the default. Both rotation mode and specific EOI mode are disabled following initialization.

Access Method

Type: I/O Register
(Size: 8 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:5	0h WO	**Rotate and EOI Codes (REOI):** R, SL, EOI - These three bits control the Rotate and End of Interrupt modes and combinations of the two. A chart of these combinations is listed above under the bit definition. 000 Rotate in Auto EOI Mode (Clear) 001 Non-specific EOI command 010 No Operation 011 *Specific EOI Command 100 Rotate in Auto EOI Mode (Set) 101 Rotate on Non-Specific EOI Command 110 *Set Priority Command 111 *Rotate on Specific EOI Command *L0 - L2 Are Used
4:3	0h WO	**OCW2 Select (O2S):** When selecting OCW2, bits 4:3 = 00
2:0	0h WO	**Interrupt Level Select (L2, L1, L0) (ILSLT):** L2, L1, and L0 determine the interrupt level acted upon when the SL bit is active. A simple binary code, outlined above, selects the channel for the command to act upon. When the SL bit is inactive, these bits do not have a defined function, programming L2, L1 and L0 to 0 is sufficient in this case.

20.10.3 Master Operational Control Word 3 (MOCW3)—Offset 20h

Note: Address should be 20h.

Access Method

Type: I/O Register
(Size: 8 bits)

Device:
Function:

Default: 8h

7			4				0
0	0	0	0	1	0	0	0
RSVD	ESMM	SMM	O3S		PMC	RRC	

Bit Range	Default and Access	Field Name (ID): Description
7	0h RO	Reserved.
6	0h WO	**Enable Special Mask Mode (ESMM):**
5	0h WO	**Special Mask Mode (SMM):** If this bit is set, the Special Mask Mode can be used by an interrupt service routine to dynamically alter the system priority structure while the routine is executing, through selective enabling/ disabling of the other channel's mask bits. Bit 6, the ESMM bit, must be set for this bit to have any meaning.
4:3	1h WO	**OCW3 Select (O3S):** When selecting OCW3, bits 4:3 = 01
2	0h WO	**Poll Mode Command (PMC):** When cleared, poll command is not issued. When set, the next I/O read to the interrupt controller is treated as an interrupt acknowledge cycle. An encoded byte is driven onto the data bus, representing the highest priority level requesting service.

Bit Range	Default and Access	Field Name (ID): Description
1:0	0h WO	**Register Read Command (RRC):** To retain the current selection (read ISR or read IRR), always write a 0 to bit 1 when programming this register. The selected register can be read repeatedly without reprogramming OCW3. To select a new status register, OCW3 must be reprogrammed prior to attempting the read. Value Command 00 No Action 01 No Action 10 Read IRQ Register 11 Read IS Register

20.10.4 Master Initialization Command Word 2 (MICW2)—Offset 21h

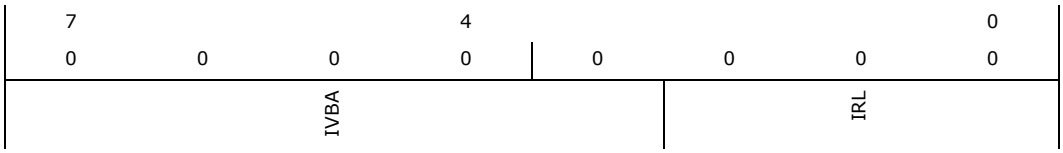
ICW2 is used to initialize the interrupt controller with the five most significant bits of the interrupt vector address. The value programmed for bits [7:3] is used by the CPU to define the base address in the interrupt vector table for the interrupt routines associated with each IRQ on the controller. Typical ISA ICW2 values are 08h for the master controller and 70h for the slave controller.

Access Method

Type: I/O Register
(Size: 8 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:3	0h WO	**Interrupt Vector Base Address (IVBA):** Bits [7:3] define the base address in the interrupt vector table for the interrupt routines associated with each interrupt request level input.
2:0	0h WO	**Interrupt Request Level (IRL):** When writing ICW2, these bits should all be 0. During an interrupt acknowledge cycle, these bits are programmed by the interrupt controller with the interrupt to be serviced. This is combined with bits [7:3] to form the interrupt vector driven onto the data bus during the second INTA# cycle. The code is a three bit binary code: Code Master Interrupt Slave Interrupt 000 IRQ0 IRQ8 001 IRQ1 IRQ9 010 IRQ2 IRQ10 011 IRQ3 IRQ11 100 IRQ4 IRQ12 101 IRQ5 IRQ13 110 IRQ6 IRQ14 111 IRQ7 IRQ15

20.10.5 Master Initialization Command Word 3 (MICW3)—Offset 21h

Note: Address should be 21h.

Access Method

Type: I/O Register
(Size: 8 bits)

Device:
Function:

Default: 7h

7			4				0
0	0	0	0	0	1	1	1
MICW3_7_3					CCC	MICW3_1_0	

Bit Range	Default and Access	Field Name (ID): Description
7:3	0h WO	**MICW3 [7:3] (MICW3_7_3):** These bits must be programmed to zero.
2	1h WO	**Cascaded Controller Connection (CCC):** This bit must always be programmed to a 1 to indicate the slave controller for interrupts 8 - 15 is cascaded on IRQ2.
1:0	3h WO	**MICW [1:0] (MICW3_1_0):** These bits must be programmed to zero.

20.10.6 Master Initialization Command Word 4 (MICW4)—Offset 21h

Note: Address should be 21h.

Access Method

Type: I/O Register
(Size: 8 bits)

Device:
Function:

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
RSVD			SFNM	BUF	MSBM	AEOI	MM

Bit Range	Default and Access	Field Name (ID): Description
7:5	0h RO	Reserved.
4	0h WO	**Special Fully Nested Mode (SFNM):** Should normally be disabled by writing a 0 to this bit. If SFNM=1, the special fully nested mode is programmed.
3	0h WO	**Buffered Mode (BUF):** Must be cleared for non-buffered mode. Writing 1 will result in undefined behavior.
2	0h WO	**Master/Slave in Buffered Mode (MSBM):** Not used. Should always be programmed to 0.
1	0h WO	**Automatic End of Interrupt (AEOI):** This bit should normally be programmed to 0. This is the normal end of interrupt. If this bit is 1, the automatic end of interrupt mode is programmed.
0	0h WO	**Microprocessor Mode (MM):** This bit must be written to 1 to indicate that the controller is operating in an Intel Architecture-based system. Writing 0 will result in undefined behavior.1

20.10.7 Master Operational Control Word 1 (MOCW1)—Offset 21h

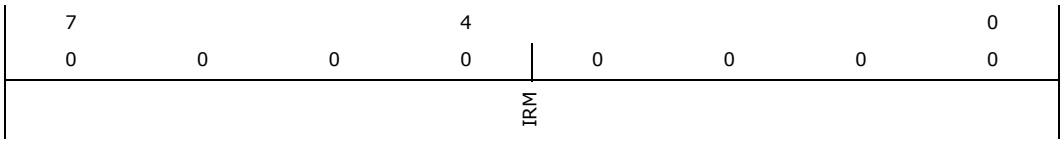
Note: Address should be 21h.

Access Method

Type: I/O Register
(Size: 8 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW	**Interrupt Request Mask (IRM):** When a 1 is written to any bit in this register, the corresponding IRQ line is masked. When a 0 is written to any bit in this register, the corresponding IRQ mask bit is cleared, and interrupt requests will again be accepted by the controller. Masking IRQ2 on the master controller will also mask the interrupt requests from the slave controller.

20.10.8 Slave Initialization Command Word 1 (SICW1)—Offset A0h

A write to Initialization Command Word 1 starts the interrupt controller initialization sequence, during which the following occurs:

1. The Interrupt Mask register is cleared.
2. IRQ7 input is assigned priority 7.
3. The slave mode address is set to 7.
4. Special Mask Mode is cleared and Status Read is set to IRR.

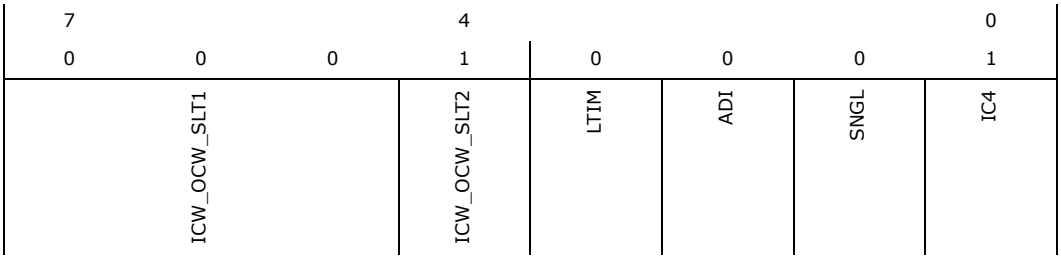
Once this write occurs, the controller expects writes to ICW2, ICW3, and ICW4 to complete the initialization sequence.

Access Method

Type: I/O Register
(Size: 8 bits)

Device:
Function:

Default: 11h



Bit Range	Default and Access	Field Name (ID): Description
7:5	0h WO	**ICW/OCW select (ICW_OCW_SLT1):** These bits are MCS-85 specific, and not needed. Should be programmed to 000
4	1h WO	**ICW/OCW select (ICW_OCW_SLT2):** This bit must be a 1 to select ICW1 and enable the ICW2, ICW3, and ICW4 sequence.
3	0h WO	**Edge/Level Bank Select (LTIM):** Disabled. Replaced by the edge/level triggered control registers (ELCR).
2	0h WO	**ADI (ADI):** Ignored for PCH. Should be programmed to 0.
1	0h WO	**Single or Cascade (SNGL):** Must be programmed to a 0 to indicate two controllers operating in cascade mode.
0	1h WO	**ICW4 Write Required (IC4):** This bit must be programmed to a 1 to indicate that ICW4 needs to be programmed.

20.10.9 Slave Operational Control Word 2 (SOCW2)—Offset A0h

Note: Address should be A0h.

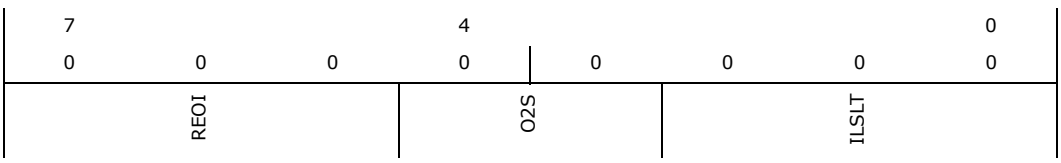
Following a part reset or ICW initialization, the controller enters the fully nested mode of operation. Non-specific EOI without rotation is the default. Both rotation mode and specific EOI mode are disabled following initialization.

Access Method

Type: I/O Register
(Size: 8 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:5	0h WO	**Rotate and EOI Codes (REOI):** R, SL, EOI - These three bits control the Rotate and End of Interrupt modes and combinations of the two. A chart of these combinations is listed above under the bit definition. 000 Rotate in Auto EOI Mode (Clear) 001 Non-specific EOI command 010 No Operation 011 *Specific EOI Command 100 Rotate in Auto EOI Mode (Set) 101 Rotate on Non-Specific EOI Command 110 *Set Priority Command 111 *Rotate on Specific EOI Command *L0 - L2 Are Used
4:3	0h WO	**OCW2 Select (O2S):** When selecting OCW2, bits 4:3 = 00
2:0	0h WO	**Interrupt Level Select (L2, L1, L0) (ILSLT):** L2, L1, and L0 determine the interrupt level acted upon when the SL bit is active. A simple binary code, outlined above, selects the channel for the command to act upon. When the SL bit is inactive, these bits do not have a defined function, programming L2, L1 and L0 to 0 is sufficient in this case.

20.10.10 Slave Operational Control Word 3 (SOCW3)—Offset A0h

Note: Address should be A0h.

Access Method

Type: I/O Register
(Size: 8 bits)

Device:
Function:

Default: 8h

7			4				0
0	0	0	0	1	0	0	0
RSVD	ESMM	SMM	O3S		PMC	RRC	

Bit Range	Default and Access	Field Name (ID): Description
7	0h RO	Reserved.
6	0h WO	**Enable Special Mask Mode (ESMM):**
5	0h WO	**Special Mask Mode (SMM):** If this bit is set, the Special Mask Mode can be used by an interrupt service routine to dynamically alter the system priority structure while the routine is executing, through selective enabling/ disabling of the other channel's mask bits. Bit 6, the ESMM bit, must be set for this bit to have any meaning.
4:3	1h WO	**OCW3 Select (O3S):** When selecting OCW3, bits 4:3 = 01
2	0h WO	**Poll Mode Command (PMC):** When cleared, poll command is not issued. When set, the next I/O read to the interrupt controller is treated as an interrupt acknowledge cycle. An encoded byte is driven onto the data bus, representing the highest priority level requesting service.

Bit Range	Default and Access	Field Name (ID): Description
1:0	0h WO	**Register Read Command (RRC):** To retain the current selection (read ISR or read IRR), always write a 0 to bit 1 when programming this register. The selected register can be read repeatedly without reprogramming OCW3. To select a new status register, OCW3 must be reprogrammed prior to attempting the read. Value Command 00 No Action 01 No Action 10 Read IRQ Register 11 Read IS Register

20.10.11 Slave Initialization Command Word 2 (SICW2)—Offset A1h

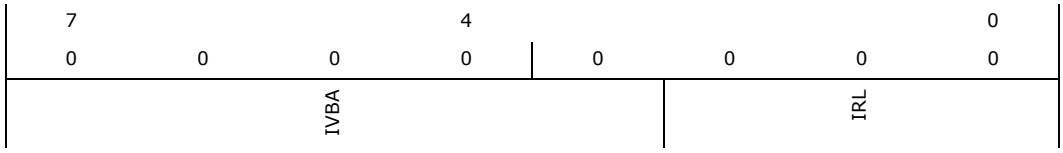
ICW2 is used to initialize the interrupt controller with the five most significant bits of the interrupt vector address. The value programmed for bits [7:3] is used by the CPU to define the base address in the interrupt vector table for the interrupt routines associated with each IRQ on the controller. Typical ISA ICW2 values are 08h for the master controller and 70h for the slave controller.

Access Method

Type: I/O Register
(Size: 8 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:3	0h WO	**Interrupt Vector Base Address (IVBA):** Bits [7:3] define the base address in the interrupt vector table for the interrupt routines associated with each interrupt request level input.
2:0	0h WO	**Interrupt Request Level (IRL):** When writing ICW2, these bits should all be 0. During an interrupt acknowledge cycle, these bits are programmed by the interrupt controller with the interrupt to be serviced. This is combined with bits [7:3] to form the interrupt vector driven onto the data bus during the second INTA# cycle. The code is a three bit binary code: Code Master Interrupt Slave Interrupt 000 IRQ0 IRQ8 001 IRQ1 IRQ9 010 IRQ2 IRQ10 011 IRQ3 IRQ11 100 IRQ4 IRQ12 101 IRQ5 IRQ13 110 IRQ6 IRQ14 111 IRQ7 IRQ15

20.10.12 Slave Initialization Command Word 3 (SICW3)—Offset A1h

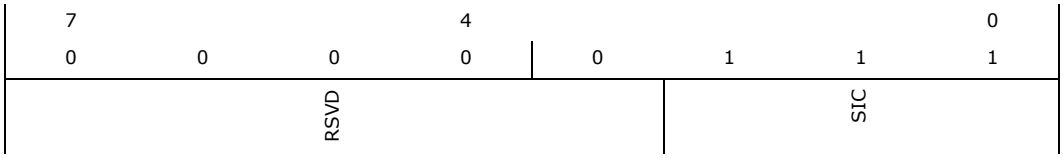
Note: Address should be A1h.

Access Method

Type: I/O Register
(Size: 8 bits)

Device:
Function:

Default: 7h



Bit Range	Default and Access	Field Name (ID): Description
7:3	0h RO	Reserved.
2:0	7h WO	**Slave Identification Code (SIC):** This field must be programmed to 02h to match the code broadcast by the master controller during the INTA# sequence.

20.10.13 Slave Initialization Command Word 4 (SICW4)—Offset A1h

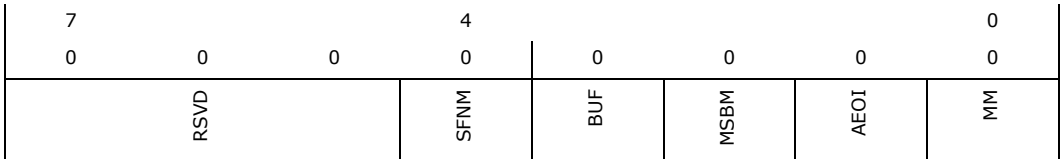
Note: Address should be A1h.

Access Method

Type: I/O Register
(Size: 8 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:5	0h RO	Reserved.
4	0h WO	**Special Fully Nested Mode (SFNM):** Should normally be disabled by writing a 0 to this bit. If SFNM=1, the special fully nested mode is programmed.
3	0h WO	**Buffered Mode (BUF):** Must be cleared for non-buffered mode. Writing 1 will result in undefined behavior.

Bit Range	Default and Access	Field Name (ID): Description
2	0h WO	**Master/Slave in Buffered Mode (MSBM):** Not used. Should always be programmed to 0.
1	0h WO	**Automatic End of Interrupt (AEOI):** This bit should normally be programmed to 0. This is the normal end of interrupt. If this bit is 1, the automatic end of interrupt mode is programmed.
0	0h WO	**Microprocessor Mode (MM):** This bit must be written to 1 to indicate that the controller is operating in an Intel Architecture-based system. Writing 0 will result in undefined behavior.1

20.10.14 Slave Operational Control Word 1 (SOCW1)—Offset A1h

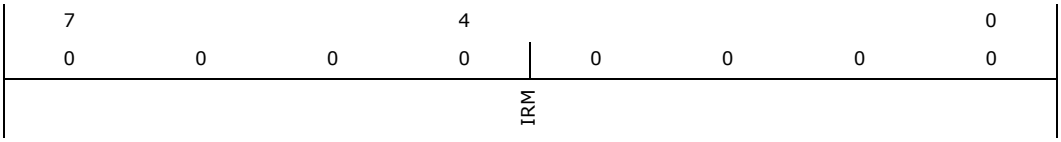
Note: Address should be A1h.

Access Method

Type: I/O Register
(Size: 8 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW	**Interrupt Request Mask (IRM):** When a 1 is written to any bit in this register, the corresponding IRQ line is masked. When a 0 is written to any bit in this register, the corresponding IRQ mask bit is cleared, and interrupt requests will again be accepted by the controller. Masking IRQ2 on the master controller will also mask the interrupt requests from the slave controller.

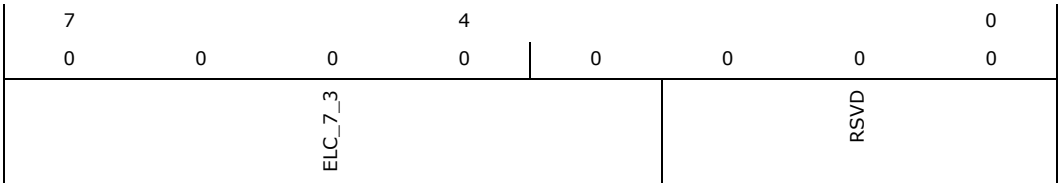
20.10.15 Master Edge/Level Control (ELCR1)—Offset 4D0h

Access Method

Type: I/O Register
(Size: 8 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:3	0h RW	**Edge Level Control (ELC_7_3):** In edge mode, (bit cleared), the interrupt is recognized by a low to high transition. In level mode (bit set), the interrupt is recognized by a high level.
2:0	0h RO	Reserved.

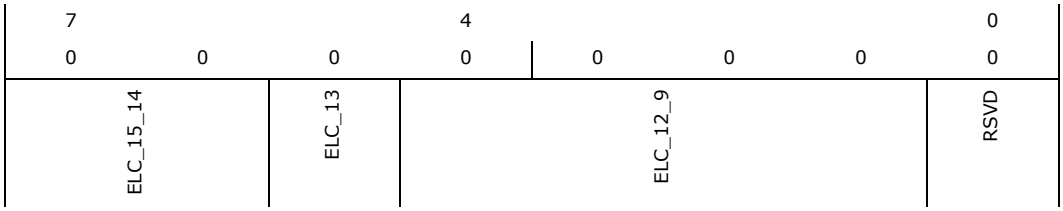
20.10.16 Slave Edge/Level Control (ELCR2)—Offset 4D1h

Access Method

Type: I/O Register
(Size: 8 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:6	0h RW	**Edge Level Control (ELC_15_14):** In edge mode, (bit cleared), the interrupt is recognized by a low to high transition. In level mode (bit set), the interrupt is recognized by a high level. Bit 7 applies to IRQ15, and bit 6 to IRQ14.
5	0h RW	**Edge Level Control (ELC_13):** In edge mode, (bit cleared), the interrupt is recognized by a low to high transition. In level mode (bit set), the interrupt is recognized by a high level. This bit applies to IRQ13.
4:1	0h RW	**Edge Level Control (ELC_12_9):** In edge mode, (bit cleared), the interrupt is recognized by a low to high transition. In level mode (bit set), the interrupt is recognized by a high level. Bit 4 applies to IRQ12, bit 3 to IRQ11, bit 2 to IRQ10, and bit 1 to IRQ9.
0	0h RO	Reserved.

20.11 Interrupt PCR Registers Summary

These registers are within the PCH Private Configuration Space which is accessible through the PCH Sideband Interface. They can be accessed via (SBREG_BAR + PortID + Register Offset).

Table 20-11. Summary of Interrupt PCR Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
3100h	3100h	PIRQA Routing Control (PARC)—Offset 3100h	80h
3101h	3101h	PIRQB Routing Control (PBRC)—Offset 3101h	80h
3102h	3102h	PIRQC Routing Control (PCRC)—Offset 3102h	80h

Table 20-11. Summary of Interrupt PCR Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
3103h	3103h	PIRQD Routing Control (PDRC)—Offset 3103h	80h
3104h	3104h	PIRQE Routing Control (PERC)—Offset 3104h	80h
3105h	3105h	PIRQF Routing Control (PFRC)—Offset 3105h	80h
3106h	3106h	PIRQG Routing Control (PGRC)—Offset 3106h	80h
3107h	3107h	PIRQH Routing Control (PHRC)—Offset 3107h	80h
3140h	3141h	PCI Interrupt Route 0 (PIR0)—Offset 3140h	3210h
3142h	3143h	PCI Interrupt Route 1 (PIR1)—Offset 3142h	0h
3144h	3145h	PCI Interrupt Route 2 (PIR2)—Offset 3144h	0h
3146h	3147h	PCI Interrupt Route 3 (PIR3)—Offset 3146h	0h
3148h	3149h	PCI Interrupt Route 4 (PIR4)—Offset 3148h	0h
31FCh	31FFh	General Interrupt Control (GIC)—Offset 31FCh	0h
3200h	3203h	Interrupt Polarity Control 0 (IPC0)—Offset 3200h	FF0000h
3204h	3207h	Interrupt Polarity Control 1 (IPC1)—Offset 3204h	0h
3208h	320Bh	Interrupt Polarity Control 2 (IPC2)—Offset 3208h	0h
320Ch	320Fh	Interrupt Polarity Control 3 (IPC3)—Offset 320Ch	0h
3300h	3303h	ITSS Power Reduction Control (ITSSPRC)—Offset 3300h	0h
3334h	3335h	Master Message Control (MMC)—Offset 3334h	0h

20.11.1 PIRQA Routing Control (PARC)—Offset 3100h

Access Method

Type: MSG Register
(Size: 8 bits)

Device:
Function:

Default: 80h

7			4				0
1	0	0	0	0	0	0	0
REN	RSVD			IR			

Bit Range	Default and Access	Field Name (ID): Description
7	1h RW	**Interrupt Routing Enable (REN):** When cleared, the corresponding PIRQ is routed to one of the legacy interrupts specified in bits[3:0]. When set, the PIRQ is not routed to the 8259.
6:4	0h RO	Reserved.
3:0	0h RW	**IRQ Routing (IR):** Bits Mapping: 1001: IRQ9 1010: IRQ10 0011: IRQ3 1011: IRQ11 0100: IRQ4 1100: IRQ12 0101: IRQ5 0110: IRQ6 1110: IRQ14 0111: IRQ7 1111: IRQ15 Others: Reserved

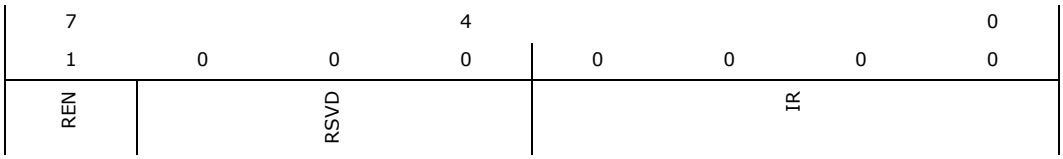
20.11.2 PIRQB Routing Control (PBRC)—Offset 3101h

Access Method

Type: MSG Register
(Size: 8 bits)

Device:
Function:

Default: 80h



Bit Range	Default and Access	Field Name (ID): Description
7	1h RW	**Interrupt Routing Enable (REN):** When cleared, the corresponding PIRQ is routed to one of the legacy interrupts specified in bits[3:0]. When set, the PIRQ is not routed to the 8259.
6:4	0h RO	Reserved.
3:0	0h RW	**IRQ Routing (IR):** Bits Mapping: 1001: IRQ9 1010: IRQ10 0011: IRQ3 1011: IRQ11 0100: IRQ4 1100: IRQ12 0101: IRQ5 0110: IRQ6 1110: IRQ14 0111: IRQ7 1111: IRQ15 Others: Reserved

20.11.3 PIRQC Routing Control (PCRC)—Offset 3102h

Access Method

Type: MSG Register
(Size: 8 bits)

Device:
Function:

Default: 80h

7			4				0
1	0	0	0	0	0	0	0
REN	RSVD			IR			

Bit Range	Default and Access	Field Name (ID): Description
7	1h RW	**Interrupt Routing Enable (REN):** When cleared, the corresponding PIRQ is routed to one of the legacy interrupts specified in bits[3:0]. When set, the PIRQ is not routed to the 8259.
6:4	0h RO	Reserved.
3:0	0h RW	**IRQ Routing (IR):** Bits Mapping: 1001: IRQ9 1010: IRQ10 0011: IRQ3 1011: IRQ11 0100: IRQ4 1100: IRQ12 0101: IRQ5 0110: IRQ6 1110: IRQ14 0111: IRQ7 1111: IRQ15 Others: Reserved

20.11.4 PIRQD Routing Control (PDRC)—Offset 3103h

Access Method

Type: MSG Register
(Size: 8 bits)

Device:
Function:

Default: 80h

7			4				0
1	0	0	0	0	0	0	0
REN	RSVD			IR			

Bit Range	Default and Access	Field Name (ID): Description
7	1h RW	**Interrupt Routing Enable (REN):** When cleared, the corresponding PIRQ is routed to one of the legacy interrupts specified in bits[3:0]. When set, the PIRQ is not routed to the 8259.
6:4	0h RO	Reserved.
3:0	0h RW	**IRQ Routing (IR):** Bits Mapping: 1001: IRQ9 1010: IRQ10 0011: IRQ3 1011: IRQ11 0100: IRQ4 1100: IRQ12 0101: IRQ5 0110: IRQ6 1110: IRQ14 0111: IRQ7 1111: IRQ15 Others: Reserved

20.11.5 PIRQE Routing Control (PERC)—Offset 3104h

Access Method

Type: MSG Register
(Size: 8 bits)

Device:
Function:

Default: 80h

7			4				0
1	0	0	0	0	0	0	0
REN	RSVD			IR			

Bit Range	Default and Access	Field Name (ID): Description
7	1h RW	**Interrupt Routing Enable (REN):** When cleared, the corresponding PIRQ is routed to one of the legacy interrupts specified in bits[3:0]. When set, the PIRQ is not routed to the 8259.
6:4	0h RO	Reserved.
3:0	0h RW	**IRQ Routing (IR):** Bits Mapping: 1001: IRQ9 1010: IRQ10 0011: IRQ3 1011: IRQ11 0100: IRQ4 1100: IRQ12 0101: IRQ5 0110: IRQ6 1110: IRQ14 0111: IRQ7 1111: IRQ15 Others: Reserved

20.11.6 PIRQF Routing Control (PFRC)—Offset 3105h

Access Method

Type: MSG Register
(Size: 8 bits)

Device:
Function:

Default: 80h

7			4				0
1	0	0	0	0	0	0	0
REN	RSVD			IR			

Bit Range	Default and Access	Field Name (ID): Description
7	1h RW	**Interrupt Routing Enable (REN):** When cleared, the corresponding PIRQ is routed to one of the legacy interrupts specified in bits[3:0]. When set, the PIRQ is not routed to the 8259.
6:4	0h RO	Reserved.
3:0	0h RW	**IRQ Routing (IR):** Bits Mapping: 1001: IRQ9 1010: IRQ10 0011: IRQ3 1011: IRQ11 0100: IRQ4 1100: IRQ12 0101: IRQ5 0110: IRQ6 1110: IRQ14 0111: IRQ7 1111: IRQ15 Others: Reserved

20.11.7 PIRQG Routing Control (PGRC)—Offset 3106h

Access Method

Type: MSG Register
(Size: 8 bits)

Device:
Function:

Default: 80h

7			4				0
1	0	0	0	0	0	0	0
REN	RSVD			IR			

Bit Range	Default and Access	Field Name (ID): Description
7	1h RW	**Interrupt Routing Enable (REN):** When cleared, the corresponding PIRQ is routed to one of the legacy interrupts specified in bits[3:0]. When set, the PIRQ is not routed to the 8259.
6:4	0h RO	Reserved.
3:0	0h RW	**IRQ Routing (IR):** Bits Mapping: 1001: IRQ9 1010: IRQ10 0011: IRQ3 1011: IRQ11 0100: IRQ4 1100: IRQ12 0101: IRQ5 0110: IRQ6 1110: IRQ14 0111: IRQ7 1111: IRQ15 Others: Reserved

20.11.8 PIRQH Routing Control (PHRC)—Offset 3107h

Access Method

Type: MSG Register
(Size: 8 bits)

Device:
Function:

Default: 80h

7			4				0
1	0	0	0	0	0	0	0
REN	RSVD			IR			

Bit Range	Default and Access	Field Name (ID): Description
7	1h RW	**Interrupt Routing Enable (REN):** When cleared, the corresponding PIRQ is routed to one of the legacy interrupts specified in bits[3:0]. When set, the PIRQ is not routed to the 8259.
6:4	0h RO	Reserved.
3:0	0h RW	**IRQ Routing (IR):** Bits Mapping: 1001: IRQ9 1010: IRQ10 0011: IRQ3 1011: IRQ11 0100: IRQ4 1100: IRQ12 0101: IRQ5 0110: IRQ6 1110: IRQ14 0111: IRQ7 1111: IRQ15 Others: Reserved

20.11.9 PCI Interrupt Route 0 (PIR0)—Offset 3140h

Access Method

Type: MSG Register
(Size: 16 bits)

Device:
Function:

Default: 3210h

15			12			8				4				0
0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
RSVD	IDR			RSVD	ICR			RSVD	IBR			RSVD	IAR		

Bit Range	Default and Access	Field Name (ID): Description
15	0h RO	Reserved.
14:12	3h RW	**Interrupt D Pin Route (IDR):** Indicates which PIRQ in PCH is connected to the INTD# pin reported for device 31 functions: Bits Pin Bits Pin 0h PIRQA# 4h PIRQE# 1h PIRQB# 5h PIRQF# 2h PIRQC# 6h PIRQG# 3h PIRQD# 7h PIRQH#
11	0h RO	Reserved.
10:8	2h RW	**Interrupt C Pin Route (ICR):** See the IDR description. This field applies to INTC#
7	0h RO	Reserved.
6:4	1h RW	**Interrupt B Pin Route (IBR):** See the IDR description. This field applies to INTB#.
3	0h RO	Reserved.
2:0	0h RW	**Interrupt A Pin Route (IAR):** See the IDR description. This field applies to INTA#.

20.11.10 PCI Interrupt Route 1 (PIR1)—Offset 3142h

Same definition as PIR0.

20.11.11 PCI Interrupt Route 2 (PIR2)—Offset 3144h

Same definition as PIR0, except this register applies to Device 29 functions.

20.11.12 PCI Interrupt Route 3 (PIR3)—Offset 3146h

Same definition as PIR0, except this register applies to Device 28 functions.

20.11.13 PCI Interrupt Route 4 (PIR4)—Offset 3148h

Same definition as PIR0, except this register applies to Device 27 functions.

20.11.14 General Interrupt Control (GIC)—Offset 31FCh

Note: FEC10000h - FEC3FFFFh is allocated to PCIe when Port I/OxApic Enable (PAE) bit is set.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD												INITVTMDIS	RSVD	AME	SDPS	MAXIRQSIZE							SERM	RSVD							CPUSDSTS

Bit Range	Default and Access	Field Name (ID): Description
31:20	0h RO	Reserved.
19	0b RW	**INIT# VLW Message Disable INITVMDIS**: When set, if other type VLW message is to be sent, the VLWValueBIt[5] under the VLW Vendor Defined Message should always be 0 - no active edge on INIT#. When cleared, the INIT# information will be conveyed as normal.
18	0b RW	Reserved
17	0h RW	**Alternate Access Mode Enable (AME):** When set, read only registers can be written, and write only registers can be read.
16	0h RW	**Shutdown Policy Select (SDPS):** When cleared (default) the PCH will update INIT# in response to the shutdown Vendor Defined Message (VDM). When set to 1, PCH will treat the shutdown VDM similar to receiving a CF9h I/O write, and will drive PLTRST# active.
15:9	0h RW	**MAX_IRQ_ENTRY_SIZE (MAXIRQSIZE):** This field indicates the size of the IOAPIC entry. The default size is 120 entries. 0000000: 120 entry size 0000001: 24 entry size (Legacy mode) 0000010 - 1111111: Reserved
8	0b RW	**Server Error Reporting Mode (SERM):** When set, the CPU Complex is the final target of all host space errors. In this mode, if the PCH detects a fatal, non-fatal, or correctable error on DMI or downstream functions from DMI, it sends one of the ERR_FATAL, ERR_NONFATAL, or ERR_CORR to CPU complex. When cleared, the PCH is the final target of all host space errors.
7:1	0h RO	Reserved.
0	0h RO/P	**CPU Shutdown Status (CPUSDSTS):** This bit is set to 1 if the CPU sends the Shutdown Special cycle message. The Shutdown Message is recognized as an INIT# event if the Shutdown Policy Select = 0, else PCH shall treat the Shutdown Special cycle as a request for CF9 Hard Reset. This is a sticky Read Only bit that is only reset by a loss of core power.

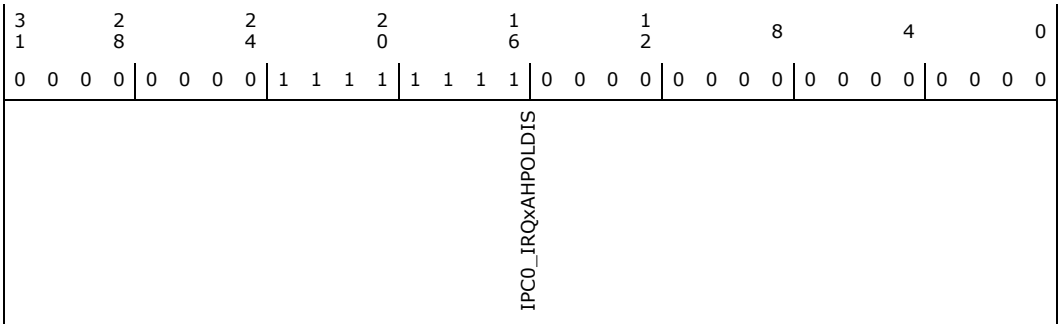
20.11.15 Interrupt Polarity Control 0 (IPC0)—Offset 3200h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: FF0000h



Bit Range	Default and Access	Field Name (ID): Description
31:0	FF0000h RW	**IRQ 31-0 Active High Polarity Disable (IPC0_IRQxAHPOLDIS):** When set to 1, the interrupt polarity associated with IRQ31 down to IRQ0 is inverted to appear as Active Low to IOAPIC. When set to 0, the interrupt is appears as Active High to IOAPIC.

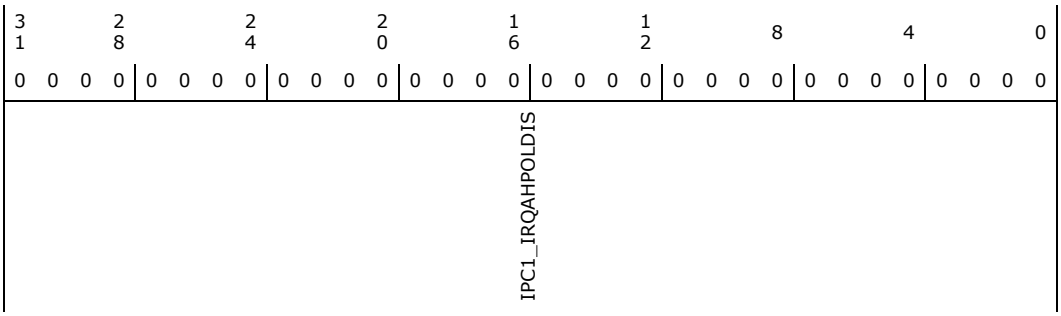
20.11.16 Interrupt Polarity Control 1 (IPC1)—Offset 3204h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**IRQ 63-32 Active High Polarity Disable (IPC1_IRQAHPOLDIS):** When set to 1, the interrupt polarity associated with IRQ63 down to IRQ32 is inverted to appear as Active Low to IOAPIC. When set to 0, the interrupt is appears as Active High to IOAPIC.

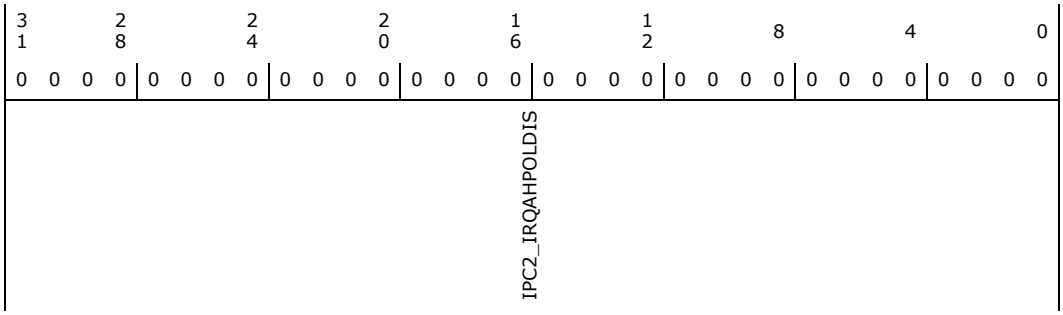
20.11.17 Interrupt Polarity Control 2 (IPC2)—Offset 3208h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**IRQ 95-64 Active High Polarity Disable (IPC2_IRQAHPOLDIS):** When set to 1, the interrupt polarity associated with IRQ95 down to IRQ64 is inverted to appear as Active Low to IOAPIC. When set to 0, the interrupt is appears as Active High to IOAPIC.

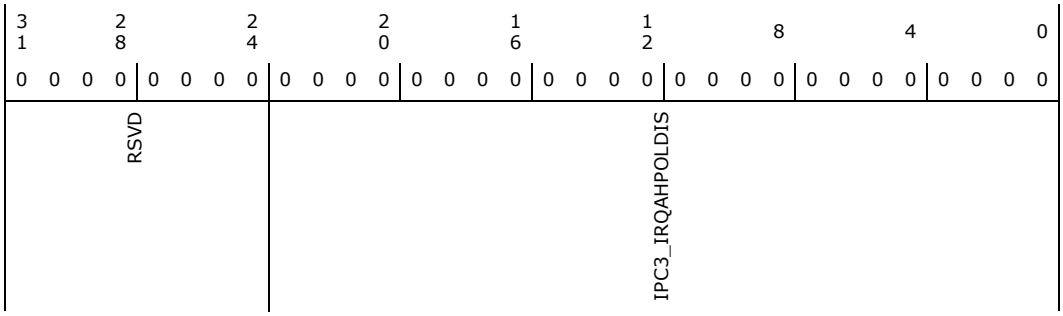
20.11.18 Interrupt Polarity Control 3 (IPC3)—Offset 320Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23:0	0h RW	**IRQ 119-96 Active High Polarity Disable (IPC3_IRQAHPOLDIS):** When set to 1, the interrupt polarity associated with IRQ119 down to IRQ96 is inverted to appear as Active Low to IOAPIC. When set to 0, the interrupt is appears as Active High to IOAPIC.

20.11.19 ITSS Power Reduction Control (ITSSPRC)—Offset 3300h

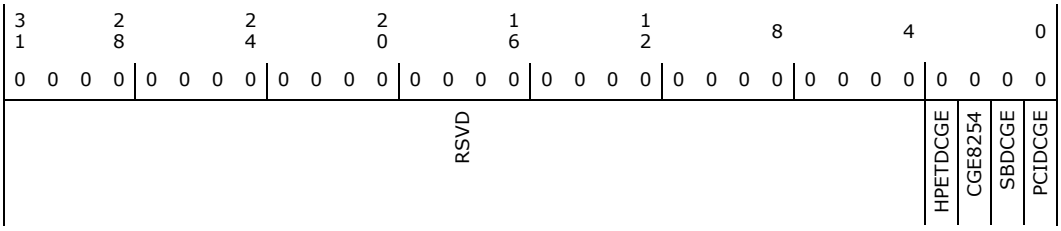
Power controls for the entire interrupt and timer subsystem.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:4	0h RO	Reserved.
3	0h RW	**HPET Dynamic Clock Gating Enable (HPETDCGE):** When set, the HPET enables dynamic clock gating.
2	0h RW	**8254 Static Clock Gating Enable (CGE8254):** When set, the 8254 timer is disabled statically. This bit shall be set by BIOS if the 8254 feature is not needed in the system or before BIOS hands off the system that supports C11. Normal operation of 8254 requires this bit to 0.
1	0h RW	**Sideband Dynamic Clock Gating Enable (SBDCGE):** Setting this bit will enable all dynamic clock gating of the Sideband Clock domain.
0	0h RW	**PCI Dynamic Clock Gating Enable (PCIDCGE):** Setting this bit will enable dynamic clock gating for the Interrupt and Timer Sub System Core Logic.

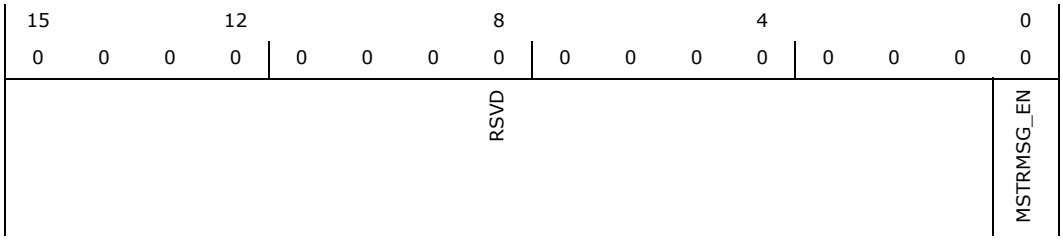
20.11.20 Master Message Control (MMC)—Offset 3334h

Access Method

Type: MSG Register
(Size: 16 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:1	0h RO	Reserved.
0	0h RW/V	**Master Message Enable (MSTRMSG_EN):** When set, allows Interrupt and Timer Subsystem (ITSS) to release any pending/in progress IOAPIC memory write, HPET memory write, virtual wire event or error messages to the I/O fabric. When cleared, ITSS prevents these messages from being issued to the I/O fabric.

20.12 APIC Indirect Registers Summary

APIC Indirect Registers lists the registers that can be accessed within the APIC using the Index Register. When accessing these registers, accesses must be done one DWord at a time. For example, software should never access byte 2 from the Data register before accessing bytes 0 and 1. The hardware will not attempt to recover from a bad programming model in this case.

Index	Mnemonic	Register Name
10-11h	RTE0	Redirection Table Entry 0
12-13h	RTE1	Redirection Table Entry 1
14-15h	RTE2	Redirection Table Entry 2
...	...	...
3E-3Fh	RTE23	Redirection Table Entry 23
40-41h	RTE24	Redirection Table Entry 24
...	...	...
FE-FFh	RTE119	Redirection Table Entry 119

Table 20-12. Summary of APIC Indirect Registers (Sheet 1 of 4)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	Identification Register (ID)—Offset 0h	0h
1h	4h	Version Register (VER)—Offset 1h	770020h
10h	17h	Redirection Table Entry 0 (RTE0)—Offset 10h	10000h
12h	19h	Redirection Table Entry 1 (RTE1)—Offset 12h	0h
14h	1Bh	Redirection Table Entry 2 (RTE2)—Offset 14h	0h

Table 20-12. Summary of APIC Indirect Registers (Sheet 2 of 4)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
16h	1Dh	Redirection Table Entry 3 (RTE3)—Offset 16h	0h
18h	1Fh	Redirection Table Entry 4 (RTE4)—Offset 18h	0h
1Ah	21h	Redirection Table Entry 5 (RTE5)—Offset 1Ah	0h
1Ch	23h	Redirection Table Entry 6 (RTE6)—Offset 1Ch	0h
1Eh	25h	Redirection Table Entry 7 (RTE7)—Offset 1Eh	0h
20h	27h	Redirection Table Entry 8 (RTE8)—Offset 20h	0h
22h	29h	Redirection Table Entry 9 (RTE9)—Offset 22h	0h
24h	2Bh	Redirection Table Entry 10 (RTE10)—Offset 24h	0h
26h	2Dh	Redirection Table Entry 11 (RTE11)—Offset 26h	0h
28h	2Fh	Redirection Table Entry 12 (RTE12)—Offset 28h	0h
2Ah	31h	Redirection Table Entry 13 (RTE13)—Offset 2Ah	0h
2Ch	33h	Redirection Table Entry 14 (RTE14)—Offset 2Ch	0h
2Eh	35h	Redirection Table Entry 15 (RTE15)—Offset 2Eh	0h
30h	37h	Redirection Table Entry 16 (RTE16)—Offset 30h	0h
32h	39h	Redirection Table Entry 17 (RTE17)—Offset 32h	0h
34h	3Bh	Redirection Table Entry 18 (RTE18)—Offset 34h	0h
36h	3Dh	Redirection Table Entry 19 (RTE19)—Offset 36h	0h
38h	3Fh	Redirection Table Entry 20 (RTE20)—Offset 38h	0h
3Ah	41h	Redirection Table Entry 21 (RTE21)—Offset 3Ah	0h
3Ch	43h	Redirection Table Entry 22 (RTE22)—Offset 3Ch	0h
3Eh	45h	Redirection Table Entry 23 (RTE23)—Offset 3Eh	0h
40h	47h	Redirection Table Entry 24 (RTE24)—Offset 40h	0h
42h	49h	Redirection Table Entry 25 (RTE25)—Offset 42h	0h
44h	4Bh	Redirection Table Entry 26 (RTE26)—Offset 44h	0h
46h	4Dh	Redirection Table Entry 27 (RTE27)—Offset 46h	0h
48h	4Fh	Redirection Table Entry 28 (RTE28)—Offset 48h	0h
4Ah	51h	Redirection Table Entry 29 (RTE29)—Offset 4Ah	0h
4Ch	53h	Redirection Table Entry 30 (RTE30)—Offset 4Ch	0h
4Eh	55h	Redirection Table Entry 31 (RTE31)—Offset 4Eh	0h
50h	57h	Redirection Table Entry 32 (RTE32)—Offset 50h	0h
52h	59h	Redirection Table Entry 33 (RTE33)—Offset 52h	0h
54h	5Bh	Redirection Table Entry 34 (RTE34)—Offset 54h	0h
56h	5Dh	Redirection Table Entry 35 (RTE35)—Offset 56h	0h
58h	5Fh	Redirection Table Entry 36 (RTE36)—Offset 58h	0h
5Ah	61h	Redirection Table Entry 37 (RTE37)—Offset 5Ah	0h
5Ch	63h	Redirection Table Entry 38 (RTE38)—Offset 5Ch	0h
5Eh	65h	Redirection Table Entry 39 (RTE39)—Offset 5Eh	0h
60h	67h	Redirection Table Entry 40 (RTE40)—Offset 60h	0h
62h	69h	Redirection Table Entry 41 (RTE41)—Offset 62h	0h
64h	6Bh	Redirection Table Entry 42 (RTE42)—Offset 64h	0h
66h	6Dh	Redirection Table Entry 43 (RTE43)—Offset 66h	0h

Table 20-12. Summary of APIC Indirect Registers (Sheet 3 of 4)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
68h	6Fh	Redirection Table Entry 44 (RTE44)—Offset 68h	0h
6Ah	71h	Redirection Table Entry 45 (RTE45)—Offset 6Ah	0h
6Ch	73h	Redirection Table Entry 46 (RTE46)—Offset 6Ch	0h
6Eh	75h	Redirection Table Entry 47 (RTE47)—Offset 6Eh	0h
70h	77h	Redirection Table Entry 48 (RTE48)—Offset 70h	0h
72h	79h	Redirection Table Entry 49 (RTE49)—Offset 72h	0h
74h	7Bh	Redirection Table Entry 50 (RTE50)—Offset 74h	0h
76h	7Dh	Redirection Table Entry 51 (RTE51)—Offset 76h	0h
78h	7Fh	Redirection Table Entry 52 (RTE52)—Offset 78h	0h
7Ah	81h	Redirection Table Entry 53 (RTE53)—Offset 7Ah	0h
7Ch	83h	Redirection Table Entry 54 (RTE54)—Offset 7Ch	0h
7Eh	85h	Redirection Table Entry 55 (RTE55)—Offset 7Eh	0h
80h	87h	Redirection Table Entry 56 (RTE56)—Offset 80h	0h
82h	89h	Redirection Table Entry 57 (RTE57)—Offset 82h	0h
84h	8Bh	Redirection Table Entry 58 (RTE58)—Offset 84h	0h
86h	8Dh	Redirection Table Entry 59 (RTE59)—Offset 86h	0h
88h	8Fh	Redirection Table Entry 60 (RTE60)—Offset 88h	0h
8Ah	91h	Redirection Table Entry 61 (RTE61)—Offset 8Ah	0h
8Ch	93h	Redirection Table Entry 62 (RTE62)—Offset 8Ch	0h
8Eh	95h	Redirection Table Entry 63 (RTE63)—Offset 8Eh	0h
90h	97h	Redirection Table Entry 64 (RTE64)—Offset 90h	0h
92h	99h	Redirection Table Entry 65 (RTE65)—Offset 92h	0h
94h	9Bh	Redirection Table Entry 66 (RTE66)—Offset 94h	0h
96h	9Dh	Redirection Table Entry 67 (RTE67)—Offset 96h	0h
98h	9Fh	Redirection Table Entry 68 (RTE68)—Offset 98h	0h
9Ah	A1h	Redirection Table Entry 69 (RTE69)—Offset 9Ah	0h
9Ch	A3h	Redirection Table Entry 70 (RTE70)—Offset 9Ch	0h
9Eh	A5h	Redirection Table Entry 71 (RTE71)—Offset 9Eh	0h
A0h	A7h	Redirection Table Entry 72 (RTE72)—Offset A0h	0h
A2h	A9h	Redirection Table Entry 73 (RTE73)—Offset A2h	0h
A4h	ABh	Redirection Table Entry 74 (RTE74)—Offset A4h	0h
A6h	ADh	Redirection Table Entry 75 (RTE75)—Offset A6h	0h
A8h	AFh	Redirection Table Entry 76 (RTE76)—Offset A8h	0h
AAh	B1h	Redirection Table Entry 77 (RTE77)—Offset AAh	0h
ACh	B3h	Redirection Table Entry 78 (RTE78)—Offset ACh	0h
AEh	B5h	Redirection Table Entry 79 (RTE79)—Offset AEh	0h
B0h	B7h	Redirection Table Entry 80 (RTE80)—Offset B0h	0h
B2h	B9h	Redirection Table Entry 81 (RTE81)—Offset B2h	0h
B4h	BBh	Redirection Table Entry 82 (RTE82)—Offset B4h	0h
B6h	BDh	Redirection Table Entry 83 (RTE83)—Offset B6h	0h
B8h	BFh	Redirection Table Entry 84 (RTE84)—Offset B8h	0h

Table 20-12. Summary of APIC Indirect Registers (Sheet 4 of 4)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
BAh	C1h	Redirection Table Entry 85 (RTE85)—Offset BAh	0h
BCh	C3h	Redirection Table Entry 86 (RTE86)—Offset BCh	0h
BEh	C5h	Redirection Table Entry 87 (RTE87)—Offset BEh	0h
C0h	C7h	Redirection Table Entry 88 (RTE88)—Offset C0h	0h
C2h	C9h	Redirection Table Entry 89 (RTE89)—Offset C2h	0h
C4h	CBh	Redirection Table Entry 90 (RTE90)—Offset C4h	0h
C6h	CDh	Redirection Table Entry 91 (RTE91)—Offset C6h	0h
C8h	CFh	Redirection Table Entry 92 (RTE92)—Offset C8h	0h
CAh	D1h	Redirection Table Entry 93 (RTE93)—Offset CAh	0h
CCh	D3h	Redirection Table Entry 94 (RTE94)—Offset CCh	0h
CEh	D5h	Redirection Table Entry 95 (RTE95)—Offset CEh	0h
D0h	D7h	Redirection Table Entry 96 (RTE96)—Offset D0h	0h
D2h	D9h	Redirection Table Entry 97 (RTE97)—Offset D2h	0h
D4h	DBh	Redirection Table Entry 98 (RTE98)—Offset D4h	0h
D6h	DDh	Redirection Table Entry 99 (RTE99)—Offset D6h	0h
D8h	DFh	Redirection Table Entry 100 (RTE100)—Offset D8h	0h
DAh	E1h	Redirection Table Entry 101 (RTE101)—Offset DAh	0h
DCh	E3h	Redirection Table Entry 102 (RTE102)—Offset DCh	0h
DEh	E5h	Redirection Table Entry 103 (RTE103)—Offset DEh	0h
E0h	E7h	Redirection Table Entry 104 (RTE104)—Offset E0h	0h
E2h	E9h	Redirection Table Entry 105 (RTE105)—Offset E2h	0h
E4h	EBh	Redirection Table Entry 106 (RTE106)—Offset E4h	0h
E6h	EDh	Redirection Table Entry 107 (RTE107)—Offset E6h	0h
E8h	EFh	Redirection Table Entry 108 (RTE108)—Offset E8h	0h
EAh	F1h	Redirection Table Entry 109 (RTE109)—Offset EAh	0h
ECh	F3h	Redirection Table Entry 110 (RTE110)—Offset ECh	0h
EEh	F5h	Redirection Table Entry 111 (RTE111)—Offset EEh	0h
F0h	F7h	Redirection Table Entry 112 (RTE112)—Offset F0h	0h
F2h	F9h	Redirection Table Entry 113 (RTE113)—Offset F2h	0h
F4h	FBh	Redirection Table Entry 114 (RTE114)—Offset F4h	0h
F6h	FDh	Redirection Table Entry 115 (RTE115)—Offset F6h	0h
F8h	FFh	Redirection Table Entry 116 (RTE116)—Offset F8h	0h
FAh	101h	Redirection Table Entry 117 (RTE117)—Offset FAh	0h
FCh	103h	Redirection Table Entry 118 (RTE118)—Offset FCh	0h
FEh	105h	Redirection Table Entry 119 (RTE119)—Offset FEh	0h

20.12.1 Identification Register (ID)—Offset 0h

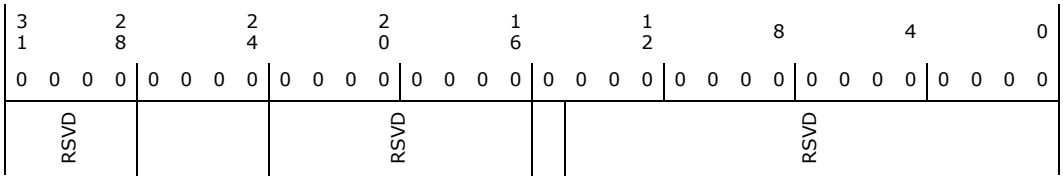
This APIC ID serves as a physical name of the APIC. The APIC bus arbitration ID for the APIC is derived from its I/O APIC ID. This register is reset to 0 on power-up reset.

Access Method

Type: APIC_IDX Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:28	0h RO	Reserved.
27:24	0h RW	**APIC ID:** Software must program this value before using the APIC.
23:16	0h RO	Reserved.
15	0h RW	**Scratchpad**
14:0	0h RO	Reserved.

20.12.2 Version Register (VER)—Offset 1h

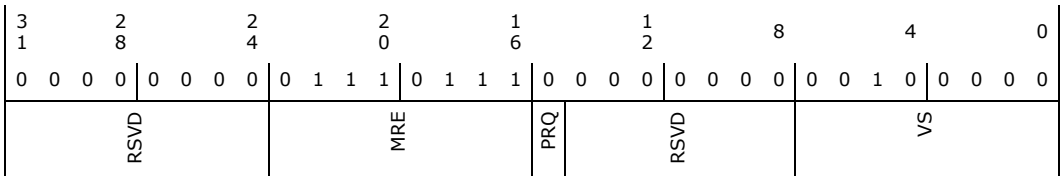
Each I/O APIC contains a hardwired Version Register that identifies different implementation of APIC and their versions. The maximum redirection entry information is also in this register to let software know how many interrupt are supported by this APIC.

Access Method

Type: APIC_IDX Register
(Size: 32 bits)

Device:
Function:

Default: 770020h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23:16	77h RW/O	**Maximum Redirection Entries (MRE):** This is the entry number (0 being the lowest entry) of the highest entry in the redirection table. It is equal to the number of interrupt input pins minus one and is in the range of 0 through 239. In PCH this field is defaulted to 17h to indicate 24 interrupts. This field is Read-Write-Once. BIOS must write to this field after PLTRST# to lock down the value. This allows BIOS to utilize some of the entries for its own purpose and thus advertising fewer IOxAPIC Redirection Entries to OS. BIOS may to program this field up to 78h (maximum 120 entries).
15	0h RO	**Pin Assertion Register Supported (PRQ):** Indicate that the IOxAPIC does not implement the Pin Assertion Register.
14:8	0h RO	Reserved.
7:0	20h RO	**Version (VS):** Identifies the implementation version as IOxAPIC.

20.12.3 Redirection Table Entry 0 (RTE0)—Offset 10h

The Redirection Table has a dedicated entry for each interrupt input pin. The information in the Redirection Table is used to translate the interrupt manifestation on the corresponding interrupt pin into an APIC message.

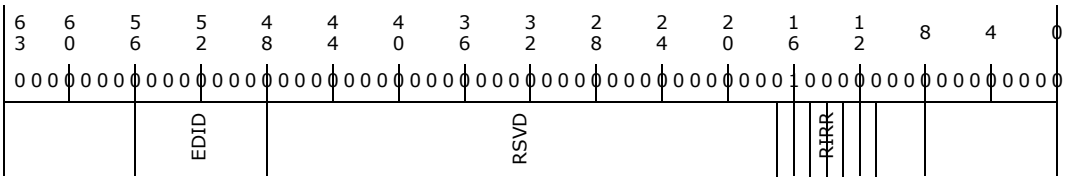
The APIC will respond to an edge triggered interrupt as long as the interrupt is held until after the acknowledge cycle has begun. Once the interrupt is detected, a delivery status bit internally to the I/O APIC is set. The state machine will step ahead and wait for an acknowledgment from the APIC unit that the interrupt message was sent. Only then will the I/O APIC be able to recognize a new edge on that interrupt pin. That new edge will only result in a new invocation of the handler if its acceptance by the destination APIC causes the Interrupt Request register bit to go from 0 to 1. (In other words, if the interrupt was not already pending a the destination.)

Access Method

Type: APIC_IDX Register
(Size: 64 bits)

Device:
Function:

Default: 10000h



Bit Range	Default and Access	Field Name (ID): Description
63:56	0h RW	**Destination:** If bit 11 of this entry is 0 (Physical), then bits 59:56 specifies an APIC ID. In this case, bits 63:59 should be programmed by software to 0. If bit 11 of this entry is 1 (Logical), then bits 63:56 specify the logical destination address of a set of processors.
55:48	0h RW	**Extended Destination ID (EDID):** These bits are sent to a local APIC only when in Processor System Bus mode. They become bits 11:4 of the address.
47:17	0h RO	Reserved.
16	1h RW	**Mask:** 0 = Not masked. An edge or level on this interrupt pin results in the delivery of the interrupt to the destination. 1 = Masked. Interrupts are not delivered nor held pending. Setting this bit after the interrupt is accepted by a local APIC has no effect on that interrupt. This behavior is identical to the device withdrawing the interrupt before it is posted to the processor. It is software's responsibility to deal with the case where the mask bit is set after the interrupt message has been accepted by a local APIC unit but before the interrupt is dispensed to the processor.
15	0h RW	**Trigger Mode:** This field indicates the type of signal on the interrupt pin that triggers an interrupt. 0 = The interrupt is edge sensitive. 1 = The interrupt is level sensitive.
14	0h RO/V	**Remote IRR (RIRR):** This is used for level triggered interrupts; its meaning is undefined for edge triggered interrupts. 0 = Reset when an EOI message is received that matches the VCT field. 1 = Set when IOxAPIC sends the level interrupt message to the CPU. Note, this bit is never set for SMI, NMI, INIT, or ExtINT delivery modes.
13	0h RW	**Interrupt Input Pin Polarity:** This bit specifies the polarity of each interrupt signal connected to the interrupt pins. 0 = Signal is active high. 1 = Signal is active low.
12	0h RO/V	**Delivery Status:** This field contains the current status of the delivery of this interrupt. 0 = Idle. There is no activity for this interrupt. 1 = Pending. An interrupt has been injected, but delivery is not complete. Note, writes to this bit have no effect.
11	0h RW	**Destination Mode:** This field is used by the local Apic to determine whether it is the destination of the message. 0 = Physical. Destination APIC ID is identified by bits 59:56. 1 = Logical. Destinations are identified by matching bit 63:56 with Logical Destination in the Destination Format register and Logical Destination register in each Local APIC.
10:8	0h RW	**Delivery Mode:** This field specifies how the APICs listed in the destination field should act upon reception of this signal. Certain Delivery Modes will only operate as intended when used in conjunction with a specific trigger mode. These encodings are: Val Name Notes 000 Fixed 001 Lowest Priority 010 SMI Not supported 011 Reserved 100 NMI Not supported 101 INIT Not supported 110 Reserved 111 ExtINT
7:0	0h RW	**Vector:** This field contains the interrupt vector for this interrupt. Values range between 10h and FEh.

20.12.4 Redirection Table Entry 1 (RTE1)—Offset 12h

This register has the same bit definition as RTE0.

20.12.5 Redirection Table Entry 2 (RTE2)—Offset 14h

This register has the same bit definition as RTE0.

20.12.6 Redirection Table Entry 3 (RTE3)—Offset 16h

This register has the same bit definition as RTE0.

20.12.7 Redirection Table Entry 4 (RTE4)—Offset 18h

This register has the same bit definition as RTE0.

20.12.8 Redirection Table Entry 5 (RTE5)—Offset 1Ah

This register has the same bit definition as RTE0.

20.12.9 Redirection Table Entry 6 (RTE6)—Offset 1Ch

This register has the same bit definition as RTE0.

20.12.10 Redirection Table Entry 7 (RTE7)—Offset 1Eh

This register has the same bit definition as RTE0.

20.12.11 Redirection Table Entry 8 (RTE8)—Offset 20h

This register has the same bit definition as RTE0.

20.12.12 Redirection Table Entry 9 (RTE9)—Offset 22h

This register has the same bit definition as RTE0.

20.12.13 Redirection Table Entry 10 (RTE10)—Offset 24h

This register has the same bit definition as RTE0.

20.12.14 Redirection Table Entry 11 (RTE11)—Offset 26h

This register has the same bit definition as RTE0.

20.12.15 Redirection Table Entry 12 (RTE12)—Offset 28h

This register has the same bit definition as RTE0.

20.12.16 Redirection Table Entry 13 (RTE13)—Offset 2Ah

This register has the same bit definition as RTE0.

20.12.17 Redirection Table Entry 14 (RTE14)—Offset 2Ch

This register has the same bit definition as RTE0.

20.12.18 Redirection Table Entry 15 (RTE15)—Offset 2Eh

This register has the same bit definition as RTE0.

20.12.19 Redirection Table Entry 16 (RTE16)—Offset 30h

This register has the same bit definition as RTE0.

20.12.20 Redirection Table Entry 17 (RTE17)—Offset 32h

This register has the same bit definition as RTE0.

20.12.21 Redirection Table Entry 18 (RTE18)—Offset 34h

This register has the same bit definition as RTE0.

20.12.22 Redirection Table Entry 19 (RTE19)—Offset 36h

This register has the same bit definition as RTE0.

20.12.23 Redirection Table Entry 20 (RTE20)—Offset 38h

This register has the same bit definition as RTE0.

20.12.24 Redirection Table Entry 21 (RTE21)—Offset 3Ah

This register has the same bit definition as RTE0.

20.12.25 Redirection Table Entry 22 (RTE22)—Offset 3Ch

This register has the same bit definition as RTE0.

20.12.26 Redirection Table Entry 23 (RTE23)—Offset 3Eh

This register has the same bit definition as RTE0.

20.12.27 Redirection Table Entry 24 (RTE24)—Offset 40h

This register has the same bit definition as RTE0.

20.12.28 Redirection Table Entry 25 (RTE25)—Offset 42h

This register has the same bit definition as RTE0.

20.12.29 Redirection Table Entry 26 (RTE26)—Offset 44h

This register has the same bit definition as RTE0.

20.12.30 Redirection Table Entry 27 (RTE27)—Offset 46h

This register has the same bit definition as RTE0.

20.12.31 Redirection Table Entry 28 (RTE28)—Offset 48h

This register has the same bit definition as RTE0.

20.12.32 Redirection Table Entry 29 (RTE29)—Offset 4Ah

This register has the same bit definition as RTE0.

20.12.33 Redirection Table Entry 30 (RTE30)—Offset 4Ch

This register has the same bit definition as RTE0.

20.12.34 Redirection Table Entry 31 (RTE31)—Offset 4Eh

This register has the same bit definition as RTE0.

20.12.35 Redirection Table Entry 32 (RTE32)—Offset 50h

This register has the same bit definition as RTE0.

20.12.36 Redirection Table Entry 33 (RTE33)—Offset 52h

This register has the same bit definition as RTE0.

20.12.37 Redirection Table Entry 34 (RTE34)—Offset 54h

This register has the same bit definition as RTE0.

20.12.38 Redirection Table Entry 35 (RTE35)—Offset 56h

This register has the same bit definition as RTE0.

20.12.39 Redirection Table Entry 36 (RTE36)—Offset 58h

This register has the same bit definition as RTE0.

20.12.40 Redirection Table Entry 37 (RTE37)—Offset 5Ah

This register has the same bit definition as RTE0.

20.12.41 Redirection Table Entry 38 (RTE38)—Offset 5Ch

This register has the same bit definition as RTE0.

20.12.42 Redirection Table Entry 39 (RTE39)—Offset 5Eh

This register has the same bit definition as RTE0.

20.12.43 Redirection Table Entry 40 (RTE40)—Offset 60h

This register has the same bit definition as RTE0.

20.12.44 Redirection Table Entry 41 (RTE41)—Offset 62h

This register has the same bit definition as RTE0.

20.12.45 Redirection Table Entry 42 (RTE42)—Offset 64h

This register has the same bit definition as RTE0.

20.12.46 Redirection Table Entry 43 (RTE43)—Offset 66h

This register has the same bit definition as RTE0.

20.12.47 Redirection Table Entry 44 (RTE44)—Offset 68h

This register has the same bit definition as RTE0.

20.12.48 Redirection Table Entry 45 (RTE45)—Offset 6Ah

This register has the same bit definition as RTE0.

20.12.49 Redirection Table Entry 46 (RTE46)—Offset 6Ch

This register has the same bit definition as RTE0.

20.12.50 Redirection Table Entry 47 (RTE47)—Offset 6Eh

This register has the same bit definition as RTE0.

20.12.51 Redirection Table Entry 48 (RTE48)—Offset 70h

This register has the same bit definition as RTE0.

20.12.52 Redirection Table Entry 49 (RTE49)—Offset 72h

This register has the same bit definition as RTE0.

20.12.53 Redirection Table Entry 50 (RTE50)—Offset 74h

This register has the same bit definition as RTE0.

20.12.54 Redirection Table Entry 51 (RTE51)—Offset 76h

This register has the same bit definition as RTE0.

20.12.55 Redirection Table Entry 52 (RTE52)—Offset 78h

This register has the same bit definition as RTE0.

20.12.56 Redirection Table Entry 53 (RTE53)—Offset 7Ah

This register has the same bit definition as RTE0.

20.12.57 Redirection Table Entry 54 (RTE54)—Offset 7Ch

This register has the same bit definition as RTE0.

20.12.58 Redirection Table Entry 55 (RTE55)—Offset 7Eh

This register has the same bit definition as RTE0.

20.12.59 Redirection Table Entry 56 (RTE56)—Offset 80h

This register has the same bit definition as RTE0.

20.12.60 Redirection Table Entry 57 (RTE57)—Offset 82h

This register has the same bit definition as RTE0.

20.12.61 Redirection Table Entry 58 (RTE58)—Offset 84h

This register has the same bit definition as RTE0.

20.12.62 Redirection Table Entry 59 (RTE59)—Offset 86h

This register has the same bit definition as RTE0.

20.12.63 Redirection Table Entry 60 (RTE60)—Offset 88h

This register has the same bit definition as RTE0.

20.12.64 Redirection Table Entry 61 (RTE61)—Offset 8Ah

This register has the same bit definition as RTE0.

20.12.65 Redirection Table Entry 62 (RTE62)—Offset 8Ch

This register has the same bit definition as RTE0.

20.12.66 Redirection Table Entry 63 (RTE63)—Offset 8Eh

This register has the same bit definition as RTE0.

20.12.67 Redirection Table Entry 64 (RTE64)—Offset 90h

This register has the same bit definition as RTE0.

20.12.68 Redirection Table Entry 65 (RTE65)—Offset 92h

This register has the same bit definition as RTE0.

20.12.69 Redirection Table Entry 66 (RTE66)—Offset 94h

This register has the same bit definition as RTE0.

20.12.70 Redirection Table Entry 67 (RTE67)—Offset 96h

This register has the same bit definition as RTE0.

20.12.71 Redirection Table Entry 68 (RTE68)—Offset 98h

This register has the same bit definition as RTE0.

20.12.72 Redirection Table Entry 69 (RTE69)—Offset 9Ah

This register has the same bit definition as RTE0.

20.12.73 Redirection Table Entry 70 (RTE70)—Offset 9Ch

This register has the same bit definition as RTE0.

20.12.74 Redirection Table Entry 71 (RTE71)—Offset 9Eh

This register has the same bit definition as RTE0.

20.12.75 Redirection Table Entry 72 (RTE72)—Offset A0h

This register has the same bit definition as RTE0.

20.12.76 Redirection Table Entry 73 (RTE73)—Offset A2h

This register has the same bit definition as RTE0.

20.12.77 Redirection Table Entry 74 (RTE74)—Offset A4h

This register has the same bit definition as RTE0.

20.12.78 Redirection Table Entry 75 (RTE75)—Offset A6h

This register has the same bit definition as RTE0.

20.12.79 Redirection Table Entry 76 (RTE76)—Offset A8h

This register has the same bit definition as RTE0.

20.12.80 Redirection Table Entry 77 (RTE77)—Offset AAh

This register has the same bit definition as RTE0.

20.12.81 Redirection Table Entry 78 (RTE78)—Offset ACh

This register has the same bit definition as RTE0.

20.12.82 Redirection Table Entry 79 (RTE79)—Offset AEh

This register has the same bit definition as RTE0.

20.12.83 Redirection Table Entry 80 (RTE80)—Offset B0h

This register has the same bit definition as RTE0.

20.12.84 Redirection Table Entry 81 (RTE81)—Offset B2h

This register has the same bit definition as RTE0.

20.12.85 Redirection Table Entry 82 (RTE82)—Offset B4h

This register has the same bit definition as RTE0.

20.12.86 Redirection Table Entry 83 (RTE83)—Offset B6h

This register has the same bit definition as RTE0.

20.12.87 Redirection Table Entry 84 (RTE84)—Offset B8h

This register has the same bit definition as RTE0.

20.12.88 Redirection Table Entry 85 (RTE85)—Offset BAh

This register has the same bit definition as RTE0.

20.12.89 Redirection Table Entry 86 (RTE86)—Offset BCh

This register has the same bit definition as RTE0.

20.12.90 Redirection Table Entry 87 (RTE87)—Offset BEh

This register has the same bit definition as RTE0.

20.12.91 Redirection Table Entry 88 (RTE88)—Offset C0h

This register has the same bit definition as RTE0.

20.12.92 Redirection Table Entry 89 (RTE89)—Offset C2h

This register has the same bit definition as RTE0.

20.12.93 Redirection Table Entry 90 (RTE90)—Offset C4h

This register has the same bit definition as RTE0.

20.12.94 Redirection Table Entry 91 (RTE91)—Offset C6h

This register has the same bit definition as RTE0.

20.12.95 Redirection Table Entry 92 (RTE92)—Offset C8h

This register has the same bit definition as RTE0.

20.12.96 Redirection Table Entry 93 (RTE93)—Offset CAh

This register has the same bit definition as RTE0.

20.12.97 Redirection Table Entry 94 (RTE94)—Offset CCh

This register has the same bit definition as RTE0.

20.12.98 Redirection Table Entry 95 (RTE95)—Offset CEh

This register has the same bit definition as RTE0.

20.12.99 Redirection Table Entry 96 (RTE96)—Offset D0h

This register has the same bit definition as RTE0.

20.12.100 Redirection Table Entry 97 (RTE97)—Offset D2h

This register has the same bit definition as RTE0.

20.12.101 Redirection Table Entry 98 (RTE98)—Offset D4h

This register has the same bit definition as RTE0.

20.12.102 Redirection Table Entry 99 (RTE99)—Offset D6h

This register has the same bit definition as RTE0.

20.12.103 Redirection Table Entry 100 (RTE100)—Offset D8h

This register has the same bit definition as RTE0.

20.12.104 Redirection Table Entry 101 (RTE101)—Offset DAh

This register has the same bit definition as RTE0.

20.12.105 Redirection Table Entry 102 (RTE102)—Offset DCh

This register has the same bit definition as RTE0.

20.12.106 Redirection Table Entry 103 (RTE103)—Offset DEh

This register has the same bit definition as RTE0.

20.12.107 Redirection Table Entry 104 (RTE104)—Offset E0h

This register has the same bit definition as RTE0.

20.12.108 Redirection Table Entry 105 (RTE105)—Offset E2h

This register has the same bit definition as RTE0.

20.12.109 Redirection Table Entry 106 (RTE106)—Offset E4h

This register has the same bit definition as RTE0.

20.12.110 Redirection Table Entry 107 (RTE107)—Offset E6h

This register has the same bit definition as RTE0.

20.12.111 Redirection Table Entry 108 (RTE108)—Offset E8h

This register has the same bit definition as RTE0.

20.12.112 Redirection Table Entry 109 (RTE109)—Offset EAh

This register has the same bit definition as RTE0.

20.12.113 Redirection Table Entry 110 (RTE110)—Offset ECh

This register has the same bit definition as RTE0.

20.12.114 Redirection Table Entry 111 (RTE111)—Offset EEh

This register has the same bit definition as RTE0.

20.12.115 Redirection Table Entry 112 (RTE112)—Offset F0h

This register has the same bit definition as RTE0.

20.12.116 Redirection Table Entry 113 (RTE113)—Offset F2h

This register has the same bit definition as RTE0.

20.12.117 Redirection Table Entry 114 (RTE114)—Offset F4h

This register has the same bit definition as RTE0.

20.12.118 Redirection Table Entry 115 (RTE115)—Offset F6h

This register has the same bit definition as RTE0.

20.12.119 Redirection Table Entry 116 (RTE116)—Offset F8h

This register has the same bit definition as RTE0.

20.12.120 Redirection Table Entry 117 (RTE117)—Offset FAh

This register has the same bit definition as RTE0.

20.12.121 Redirection Table Entry 118 (RTE118)—Offset FCh

This register has the same bit definition as RTE0.

20.12.122 Redirection Table Entry 119 (RTE119)—Offset FEh

This register has the same bit definition as RTE0.

20.13 Advanced Programmable Interrupt Controller (APIC) Registers Summary

The APIC is accessed using an indirect addressing scheme. Two registers are visible by software for manipulation of most of the APIC registers. These registers are mapped into memory space. The address bits [19:12] of the address range are programmable through bits [7:0] of OIC register (Chipset Configuration Register: Offset 31FEh). The registers are shown below.

Table 20-13. Summary of Advanced Programmable Interrupt Controller (APIC) Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	Index Register (IDX)—Offset 0h	0h
10h	13h	Data Register (DAT)—Offset 10h	0h
40h	43h	EOI Register (EOIR)—Offset 40h	0h

20.13.1 Index Register (IDX)—Offset 0h

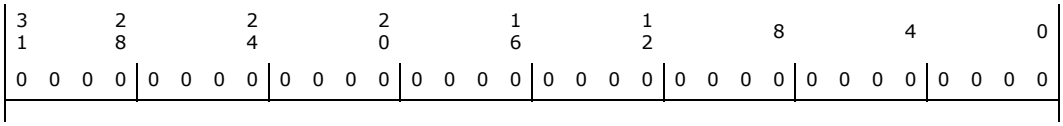
The Index Register will select which APIC indirect register to be manipulated by software. Software will program this register to select the desired APIC internal register.

Access Method

Type: MEM Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**APIC Index:** This is an 8 bit pointer into the I/O APIC register table.

20.13.2 Data Register (DAT)—Offset 10h

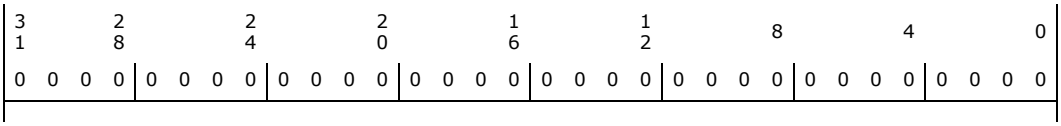
This 32-bit register specifies the data to be read or written to the register pointed to by the Index register. This register can be accessed only in DW quantities.

Access Method

Type: MEM Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**APIC Data:** This is a 32-bit register for the data to be read or written to the APIC indirect register pointed to by the Index register (Memory Address FEC0_0000h).

20.13.3 EOI Register (EOIR)—Offset 40h

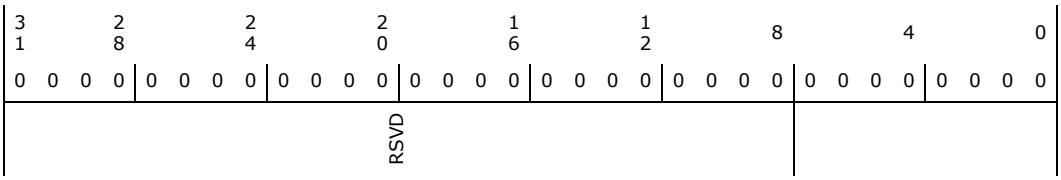
When a write is issued to this register, the IOxAPIC will check the lower 8 bits written to this register, and compare it with the vector field for each entry in the I/O Redirection Table. When a match is found, RTE.RIRR for that entry will be cleared. If multiple entries have the same vector, each of those entries will have RTE.RIRR cleared. Only bits [7:0] are used. Bits [31:8] are ignored.

Access Method

Type: MEM Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:8	0h RO	Reserved.
7:0	0h WO	**Redirection Entry Clear:** When a write is issued to this register, the I/O APIC will check this field and compare it with the vector field for each entry in the I/O Redirection Table. When a match is found, the Remote_IRR bit for that I/O Redirection Entry will be cleared.

§

21 Low Pin Count

21.1 Acronyms

Acronyms	Description
LPC	Low Pin Count

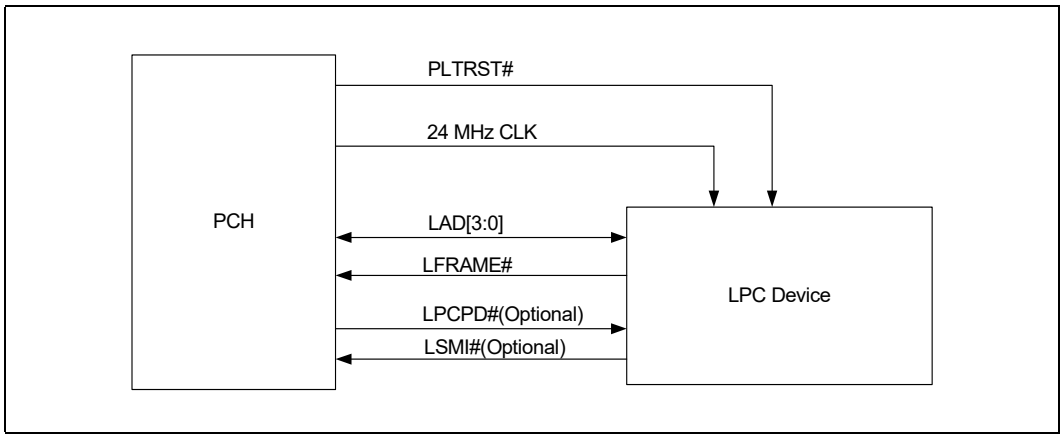
21.2 References

Specification	Location
Intel® Low Pin Count Interface Specification Revision 1.1	http://developer.intel.com/design/chipsets/industry/lpc.htm

21.3 Overview

The PCH implements an LPC interface as described in the *Low Pin Count Interface Specification,* Revision 1.1. The LPC interface to the PCH is shown in the following figure.

Table 21-1. LPC Interface Diagram



The PCH supports all of the signals that are shown as optional, but peripherals are not required to do so.

LSMI# can be connected to any of the PCH's SMI capable GPIO signals.

Note: The LPC bridge cannot be configured as a subtractive decode agent.

21.4 Signal Description

Name	Type	Description
GPP_A1_LAD0_ESPI_IO0	I/O	**LPC Multiplexed Command, Address, Data.** For LAD0, internal pull-up is provided.
GPP_A2_LAD1_ESPI_IO1	I/O	**LPC Multiplexed Command, Address, Data.** For LAD1, internal pull-up is provided.
GPP_A3_LAD2_ESPI_IO2	I/O	**LPC Multiplexed Command, Address, Data.** For LAD2, internal pull-up is provided.
GPP_A4_LAD3_ESPI_IO3	I/O	**LPC Multiplexed Command, Address, Data.** For LAD3, internal pull-up is provided.
GPP_A5_LFRAME_N_ESPI_CS1_N	O	**LPC Frame:** LFRAME# indicates the start of an LPC cycle, or an abort.

21.5 Integrated Pull-ups and Pull-downs

Signal	Resistor Type	Value	Notes
LAD[3:0]	Pull-up	15 - 40 kΩ	
Note: Integrated pull-ups and pull-downs will not be valid until all the voltages have reached a valid level.			

21.6 I/O Signal Planes and States

Signal Name	Power Plane	During Reset	Immediately after Reset	S3/S4/S5	Deep Sx
LAD[3:0]	Primary	Internal Pull-up	Internal Pull-up	Undriven	Off
LFRAME_N	Primary	Driven High	Driven High	Driven Low	Off

21.7 Functional Description

The PCH LPC interface supports the *Low Pin Count Interface Specification,* Revision 1.1. The bus is operating at 24 MHz clock frequency.

One change in the C620 series chipset from previous chipsets is the removal of the LDRQ_N signals. This results in LPC DMA/bus mastering not being supported.

21.7.1 LPC Cycle Types

The PCH implements the cycle types shown in Table 21-2.

Table 21-2. LPC Cycle Types Supported (Sheet 1 of 2)

Cycle Type	Comment
Memory Read	One byte only—(See Note 1 below)
Memory Write	One byte only—(See Note 1 below)
I/O Read	One byte only—The PCH breaks up 16-bit and 32-bit processor cycles into multiple 8-bit transfers.
I/O Write	One byte only—The PCH breaks up 16-bit and 32-bit processor cycles into multiple 8-bit transfers.
Bus Master Read	Can be one, two or four bytes—(See Note 2 below)
Bus Master Write	Can be 1, 2 or 4 bytes—(See Note 2 below)

Table 21-2. LPC Cycle Types Supported (Sheet 2 of 2)

Cycle Type	Comment
Notes: 1. The PCH provides a single generic memory range (LGMR) for decoding memory cycles and forwarding them as LPC Memory cycles on the LPC bus. The LGMR memory decode range is 64 KB in size and can be defined as being anywhere in the 4-GB memory space. This range needs to be configured by BIOS during POST to provide the necessary memory resources. BIOS should advertise the LPC Generic Memory Range as Reserved to the OS in order to avoid resource conflict. For larger transfers, the PCH performs multiple 8-bit transfers. If the cycle is not claimed by any peripheral, it is subsequently aborted, and the PCH returns a value of all 1s to the processor. This is done to maintain compatibility with ISA memory cycles where pull-up resistors would keep the bus high if no device responds. 2. Bus Master Read or Write cycles must be naturally aligned. For example, a 1-byte transfer can be to any address. However, the 2-byte transfer must be word-aligned (that is, with an address where A0=0). A DWord transfer must be DWord-aligned (that is, with an address where A1 and A0 are both 0)	

21.7.2 Start Field Definition

Table 21-3. Start Field Bit Definitions

Bits[3:0] Encoding	Definition
0000	Start of cycle for a generic target
1111	Stop/Abort: End of a cycle for a target.
Note: All other encodings are RESERVED.	

21.7.3 Cycle Type/Direction

The PCH always drives Bit 0 of this field to 0. Table 21-4 shows the valid bit encodings.

Table 21-4. Cycle Type Bit Definitions

Bits[3:2]	Bit1	Definition
00	0	I/O Read
00	1	I/O Write
01	0	Memory Read
01	1	Memory Read
11	x	Reserved. If a peripheral performing a bus master cycle generates this value, the PCH aborts the cycle.
Note: All other encodings are RESERVED.		

21.7.4 Size

Bits [3:2] are reserved. The PCH always drives them to 00. Bits [1:0] are encoded as listed in Table 21-5.

Table 21-5. Transfer Size Bit Definition

Bits[1:0]	Size
00	8-bit transfer (one byte)
01	16-bit transfer (two bytes)
10	Reserved—The PCH never drives this combination.
11	32-bit transfer (four bytes)
Note: All other combination are RESERVED.	

21.7.4.1 SYNC

Valid values for the SYNC field are shown in Table 21-6.

21.7.5 SYNC Timeout

Table 21-6. SYNC Bit Definition

Bits[3:0]	Indication
0000	**Ready:** SYNC achieved with no error.
0101	**Short Wait:** Part indicating wait-states. For bus master cycles, the PCH does not use this encoding. Instead, the PCH uses the Long Wait encoding (see next encoding below).
0110	**Long Wait:** Part indicating wait-states, and many wait-states will be added. This encoding driven by the PCH for bus master cycles, rather than the Short Wait (0101).
1010	**Error:** Sync achieved with error. This is generally used to replace the SERR# or IOCHK# signal on the PCI/ISA bus. It indicates that the data is to be transferred, but there is a serious error in this transfer.
Notes: 1. All other combinations are RESERVED. 2. If the LPC controller receives any SYNC returned from the device other than short (0101), long wait (0110), or ready (0000) when running a FWH cycle, indeterminate results may occur. A FWH device is not allowed to assert an Error SYNC.	

There are several error cases that can occur on the LPC interface. The PCH responds as defined in Section 4.2.1.9 of the *Low Pin Count Interface Specification*, Revision 1.1 (http://www.intel.com/design/chipsets/industry/lpc.htm) to the stimuli described therein. There may be other peripheral failure conditions; however, these are not handled by the PCH.

21.7.6 SYNC Error Indication

The PCH responds as defined in Section 4.2.1.10 of the *Low Pin Count Interface Specification,* Revision 1.1.

Upon recognizing the SYNC field indicating an error, the PCH treats this as a SERR by reporting this into the Device 31 Error Reporting Logic.

21.7.7 LFRAME# Usage

The PCH follows the usage of LFRAME# as defined in the *Low Pin Count Interface Specification,* Revision 1.1.

The PCH performs an abort for the following cases (possible failure cases):

- The PCH starts a Memory or I/O cycle, but no device drives a valid SYNC after four consecutive clocks.
- The PCH starts a Memory or I/O and the peripheral drives an invalid SYNC pattern.
- A peripheral drives an invalid value.

21.7.8 I/O Cycles

For I/O cycles targeting registers specified in the PCH's decode ranges, the PCH performs I/O cycles as defined in the *Low Pin Count Interface Specification,* Revision 1.1. These are 8-bit transfers. If the processor attempts a 16-bit or 32-bit transfer, the PCH breaks the cycle up into multiple 8-bit transfers to consecutive I/O addresses.

Note: If the cycle is not claimed by any peripheral (and subsequently aborted), the PCH returns a value of all 1s (FFh) to the processor. This is to maintain compatibility with ISA I/O cycles where pull-up resistors would keep the bus high if no device responds.

21.7.9 Configuration and PCH Implications

21.7.9.1 LPC I/F Decoders

To allow the I/O cycles and memory mapped cycles to go to the LPC interface, the PCH includes several decoders. During configuration, the PCH must be programmed with the same decode ranges as the peripheral. The decoders are programmed using the D 31:F0 configuration space.

Note: The PCH cannot accept PCI write cycles from PCI-to-PCI bridges or devices with similar characteristics (specifically those with a "Retry Read" feature which is enabled) to an LPC device if there is an outstanding LPC read cycle towards the same PCI device or bridge. These cycles are not part of normal system operation, but may be encountered as part of platform validation testing using custom test fixtures.

21.8 LPC Configuration Registers Summary

Table 21-7. Summary of LPC Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	Identifiers (ID)—Offset 0h	8086h
4h	5h	Device Command (CMD)—Offset 4h	7h
6h	7h	Status (STS)—Offset 6h	210h
8h	8h	Revision ID (RID)—Offset 8h	0h
9h	Ch	Class Code (CC)—Offset 9h	60100h
Eh	Eh	Header Type (HTYPE)—Offset Eh	80h
2Ch	2Fh	Sub System Identifiers (SS)—Offset 2Ch	0h
34h	34h	Capability List Pointer (CAPP)—Offset 34h	E0h
64h	64h	Serial IRQ Control (SCNT)—Offset 64h	10h
80h	81h	I/O Decode Ranges (IOD)—Offset 80h	0h
82h	83h	I/O Enables (IOE)—Offset 82h	0h
84h	87h	LPC Generic I/O Range #1 (LGIR1)—Offset 84h	0h
88h	8Bh	LPC Generic I/O Range #2 (LGIR2)—Offset 88h	0h
8Ch	8Fh	LPC Generic I/O Range #3 (LGIR3)—Offset 8Ch	0h
90h	93h	LPC Generic I/O Range #4 (LGIR4)—Offset 90h	0h
94h	97h	USB Legacy Keyboard/Mouse Control (ULKMC)—Offset 94h	0h
98h	9Bh	LPC Generic Memory Range (LGMR)—Offset 98h	0h
D0h	D3h	FWH ID Select #1 (FS1)—Offset D0h	112233h
D4h	D5h	FWH ID Select #2 (FS2)—Offset D4h	4567h
D8h	D9h	BIOS Decode Enable (BDE)—Offset D8h	FFCFh
DCh	DCh	BIOS Control (BC)—Offset DCh	20h
E0h	E3h	PCI Clock Control (PCCTL)—Offset E0h	0h

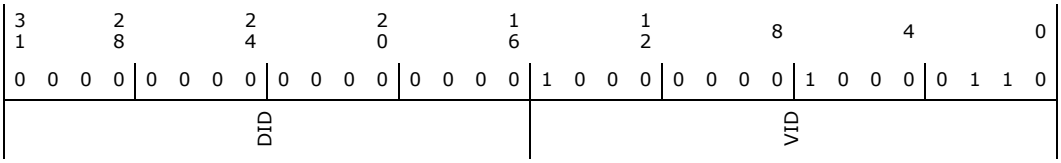
21.8.1 Identifiers (ID)—Offset 0h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 8086h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	**Device Identification (DID):** This is a 16-bit value assigned to the PCH LPC bridge.
15:0	8086h RO	**Vendor Identification (VID):** Indicates Intel.

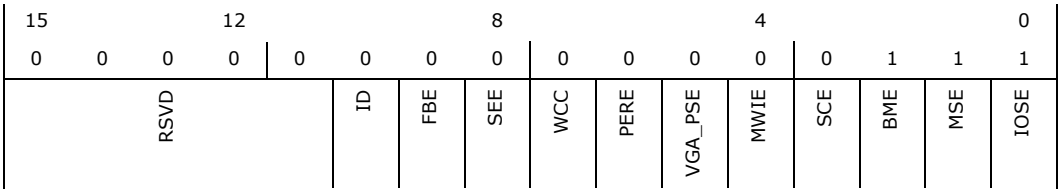
21.8.2 Device Command (CMD)—Offset 4h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 0

Default: 7h



Bit Range	Default and Access	Field Name (ID): Description
15:11	0h RO	Reserved.
10	0h RO	**Interrupt Disable (ID):** The LPC bridge has no interrupts to disable.
9	0h RO	**Fast Back to Back Enable (FBE):** Reserved as 0 per PCI Express Specification.
8	0h RW	**SERR# Enable (SEE):** The LPC bridge generates SERR# if this bit is set.
7	0h RO	**Wait Cycle Control (WCC):** Reserved as 0 per PCI Express Specification.

Bit Range	Default and Access	Field Name (ID): Description
6	0h RW	**Parity Error Response Enable (PERE):** When this bit is set to 1, it enables the LPC bridge to response to parity errors detected on backbone interface.
5	0h RO	**VGA Palette Snoop (VGA_PSE):** Reserved as 0 per *PCI Express Specification*.
4	0h RO	**Memory Write and Invalidate Enable (MWIE):** Reserved as 0 per *PCI Express Specification*.
3	0h RO	**Special Cycle Enable (SCE):** Reserved as 0 per *PCI Express Specification*.
2	1h RO	**Bus Master Enable (BME):** Bus Masters cannot be disabled.
1	1h RO	**Memory Space Enable (MSE):** Memory space cannot be disabled on LPC.
0	1h RO	**I/O Space Enable (IOSE):** I/O space cannot be disabled on LPC.

21.8.3 Status (STS)—Offset 6h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 0

Default: 210h

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0
DPE	SSE	RMA	RTA	STA	DTS		DPD	FBC	RSVD	C66	CLIST	IS	RSVD		

Bit Range	Default and Access	Field Name (ID): Description
15	0h RW/1C	**Detected Parity Error (DPE):** Set when the bridge detects a parity error on the internal backbone. This bit gets set even if CMD.PERE is not set.
14	0h RW/1C	**Signaled System Error (SSE):** Set when the LPC bridge signals a system error to the internal SERR# logic.
13	0h RO	**Received Master Abort (RMA):** Set when the bridge receives a completion with unsupported request status from the backbone. LPC is a target only controller.
12	0h RO	**Received Target Abort (RTA):** Set when the bridge receives a completion with completer abort status from the backbone. LPC is a target only controller.
11	0h RW/1C	**Signaled Target Abort (STA):** Set when the bridge generates a completion packet with target abort status on the backbone.
10:9	1h RO	**DEVSEL# Timing Status (DTS):** Indicates medium timing, although this has no meaning on the backbone.
8	0h RW/1C	**Data Parity Error Detected (DPD):** Set when the bridge receives a completion packet from the backbone from a previous request, and detects a parity error, and CMD.PERE is set.
7	0h RO	**Fast Back to Back Capable (FBC):** Reserved - bit has no meaning on internal backbone.

Bit Range	Default and Access	Field Name (ID): Description
6	0h RO	Reserved.
5	0h RO	**66 MHz Capable (C66):** Reserved - bit has no meaning on internal backbone.
4	1h RO	**Capabilities List (CLIST):** There is a capabilities list in the LPC bridge.
3	0h RO	**Interrupt Status (IS):** The LPC bridge does not generate interrupts.
2:0	0h RO	Reserved.

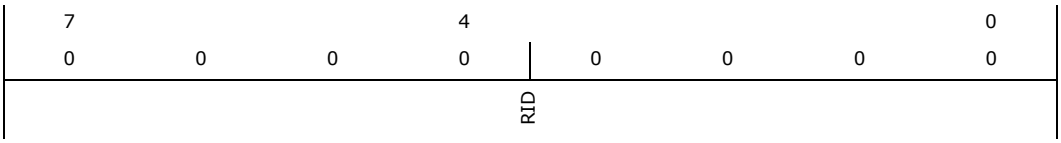
21.8.4 Revision ID (RID)—Offset 8h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO	**Revision ID (RID):** Indicates the PCH revision.

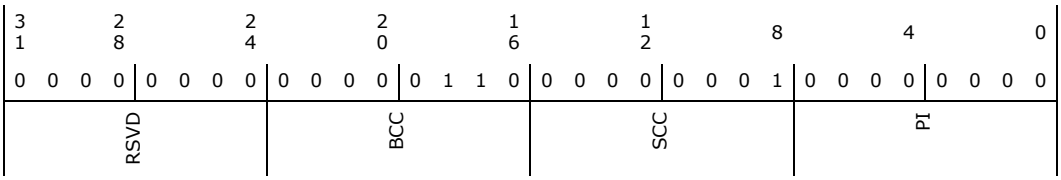
21.8.5 Class Code (CC)—Offset 9h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 60100h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23:16	6h RO	**Base Class Code (BCC):** Indicates the device is a bridge device.
15:8	1h RO	**Sub-Class Code (SCC):** Indicates the device a PCI to ISA bridge.
7:0	0h RO	**Programming Interface (PI):** The LPC bridge has no programming interface.

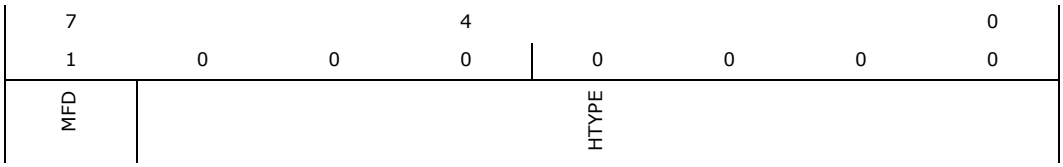
21.8.6 Header Type (HTYPE)—Offset Eh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 0

Default: 80h



Bit Range	Default and Access	Field Name (ID): Description
7	1h RO	**Multi-function Device (MFD):** This bit is 1 to indicate a multifunction device.
6:0	0h RO	**Header Type (HTYPE):** Identifies the header layout of the configuration space, which is a generic device.

21.8.7 Sub System Identifiers (SS)—Offset 2Ch

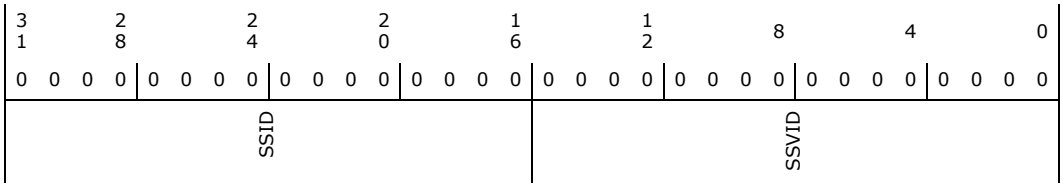
This register is initialized to logic 0 by the assertion of PLTRST#. This register can be written only once after PLTRST# de-assertion.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RW/O	**Subsystem ID (SSID):** This is written by BIOS. No hardware action taken on this value.
15:0	0h RW/O	**Subsystem Vendor ID (SSVID):** This is written by BIOS. No hardware action taken on this value.

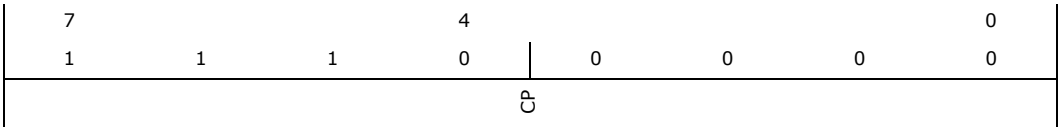
21.8.8 Capability List Pointer (CAPP)—Offset 34h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 0

Default: E0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	E0h RO	**Capability Pointer (CP):** Indicates the offset of the first Capability Item.

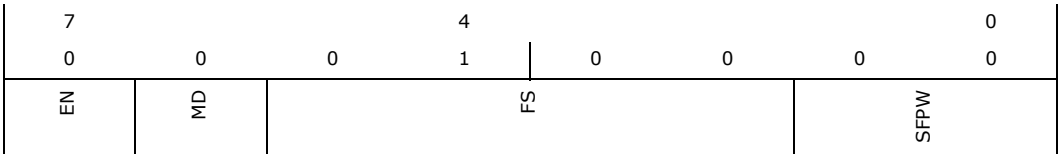
21.8.9 Serial IRQ Control (SCNT)—Offset 64h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 0

Default: 10h



Bit Range	Default and Access	Field Name (ID): Description
7	0h RW	**Enable (EN):** When set, serial IRQs will be recognized.

Bit Range	Default and Access	Field Name (ID): Description
6	0h RW	**Mode (MD):** When set, the serial IRQ machine will be in continuous mode. When cleared, the serial IRQ machine will be in quiet mode. When setting the EN bit, this bit must also be written as a one to guarantee that the first action of the serial IRQ machine will be a start frame.
5:2	4h RO	**Frame Size (FS):** Fixed field that indicates the size of the SERIRQ frame as 21 frames.
1:0	0h RW	**Start Frame Pulse Width (SFPW):** This is the number of 33 MHz clocks that the SERIRQ pin will be driven low by the Serial IRQ controller to signal a start frame. In continuous mode, the controller will drive the start frame for the number of clocks specified. In quiet mode, the controller will drive the start frame for the number of clocks specified minus one, as the first clock was driven by the peripheral. Bits Clocks 00 4 01 6 10 8 11 Reserved

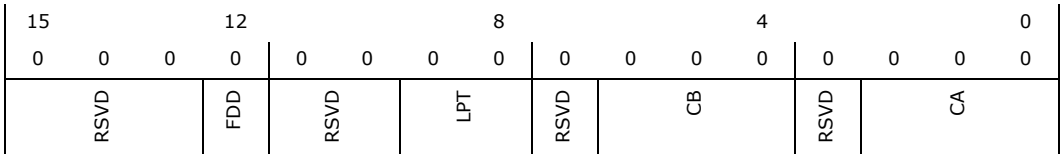
21.8.10 I/O Decode Ranges (IOD)—Offset 80h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:13	0h RO	Reserved.
12	0h RW	**FDD Range (FDD):** The following table describes which range to decode for the FDD Port Bits Decode Range 0 3F0h - 3F5h, 3F7h (Primary) 1 370h - 375h, 377h (Secondary)
11:10	0h RO	Reserved.
9:8	0h RW	**LPT Range (LPT):** The following table describes which range to decode for the LPT Port: Bits Decode Range 00 378h - 37Fh and 778h - 77Fh 01 278h - 27Fh (port 279h is read only) and 678h - 67Fh 10 3BCh - 3BEh and 7BCh - 7BEh 11 Reserved
7	0h RO	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
6:4	0h RW	**ComB Range (CB):** The following table describes which range to decode for the COMB Port Bits Decode Range 000 3F8h - 3FFh (COM 1) 001 2F8h - 2FFh (COM 2) 010 220h - 227h 011 228h - 22Fh 100 238h - 23Fh 101 2E8h - 2EFh (COM 4) 110 338h - 33Fh 111 3E8h - 3EFh (COM 3)
3	0h RO	Reserved.
2:0	0h RW	**ComA Range (CA):** The following table describes which range to decode for the COMA Port Bits Decode Range 000 3F8h - 3FFh (COM 1) 001 2F8h - 2FFh (COM 2) 010 220h - 227h 011 228h - 22Fh 100 238h - 23Fh 101 2E8h - 2EFh (COM 4) 110 338h - 33Fh 111 3E8h - 3EFh (COM 3)

21.8.11 I/O Enables (IOE)—Offset 82h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 0

Default: 0h

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD		ME2	SE	ME1	KE	HGE	LGE	RSVD				FDE	PPE	CBE	CAE

Bit Range	Default and Access	Field Name (ID): Description
15:14	0h RO	Reserved.
13	0h RW	**Microcontroller Enable #2 (ME2):** Enables decoding of I/O locations 4Eh and 4Fh to LPC.
12	0h RW	**SuperI/O Enable (SE):** Enables decoding of I/O locations 2Eh and 2Fh to LPC.
11	0h RW	**Microcontroller Enable #1 (ME1):** Enables decoding of I/O locations 62h and 66h to LPC.
10	0h RW	**Keyboard Enable (KE):** Enables decoding of the keyboard I/O locations 60h and 64h to LPC.
9	0h RW	**High Gameport Enable (HGE):** Enables decoding of the I/O locations 208h to 20Fh to LPC.
8	0h RW	**Low Gameport Enable (LGE):** Enables decoding of the I/O locations 200h to 207h to LPC.

Bit Range	Default and Access	Field Name (ID): Description
7:4	0h RO	Reserved.
3	0h RW	**Floppy Drive Enable (FDE):** Enables decoding of the FDD range to LPC. Range is selected by LIOD.FDE
2	0h RW	**Parallel Port Enable (PPE):** Enables decoding of the LPT range to LPC. Range is selected by LIOD.LPT.
1	0h RW	**Com Port B Enable (CBE):** Enables decoding of the COMB range to LPC. Range is selected LIOD.CB.
0	0h RW	**Com Port A Enable (CAE):** Enables decoding of the COMA range to LPC. Range is selected LIOD.CA.

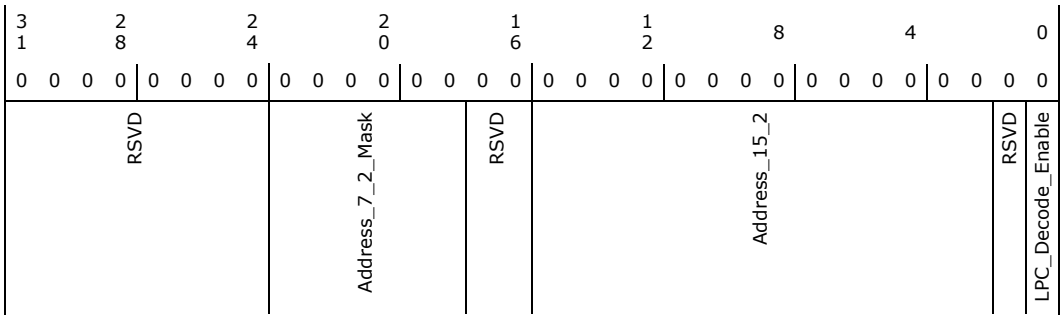
21.8.12 LPC Generic I/O Range #1 (LGIR1)—Offset 84h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23:18	0h RW	**Address[7:2] Mask (Address_7_2_Mask):** A 1 in any bit position indicates that any value in the corresponding address bit in a received cycle will be treated as a match. The corresponding bit in the Address field, below, is ignored. The mask is only provided for the lower 6 bits of the DWord address, allowing for decoding blocks up to 256 bytes in size.
17:16	0h RO	Reserved.
15:2	0h RW	**Address[15:2] (Address_15_2):** DWord-aligned address. Note that PCH does not provide decode down to the word or byte level.
1	0h RO	Reserved.
0	0h RW	**LPC Decode Enable (LPC_Decode_Enable):** When this bit is set to 1, then the range specified in this register is enabled for decoding to LPC.

21.8.13 LPC Generic I/O Range #2 (LGIR2)—Offset 88h

Same bit definition as Generic I/O Range #1 (LGIR1).

21.8.14 LPC Generic I/O Range #3 (LGIR3)—Offset 8Ch

Same bit definition as Generic I/O Range #1 (LGIR1).

21.8.15 LPC Generic I/O Range #4 (LGIR4)—Offset 90h

Same bit definition as Generic I/O Range #1 (LGIR1).

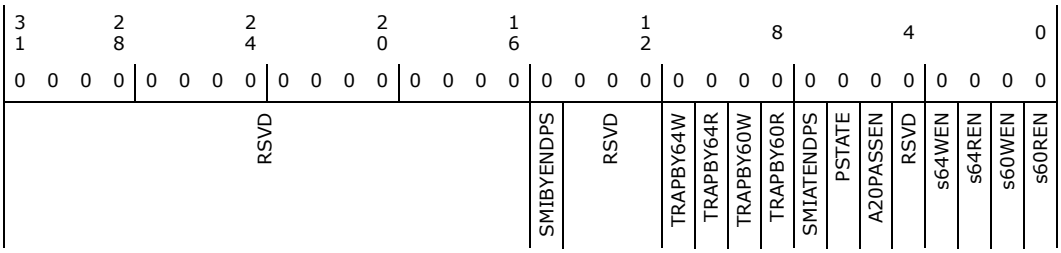
21.8.16 USB Legacy Keyboard/Mouse Control (ULKMC)—Offset 94h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15	0h RW/1C	**SMI Caused by End of Pass-through (SMIBYENDPS):** Indicates if the event occurred. Note that even if the corresponding enable bit is not set in the Bit 7, then this bit will still be active. It is up to the SMM code to use the enable bit to determine the exact cause of the SMI#. Writing a 1 to this bit will clear the latch.
14:12	0h RO	Reserved.
11	0h RW/1C	**SMI Caused by Port 64 Write (TRAPBY64W):** Indicates if the event occurred. Note that even if the corresponding enable bit is not set in the Bit 3, then this bit will still be active. It is up to the SMM code to use the enable bit to determine the exact cause of the SMI#. Writing a 1 to this bit will clear the latch. Note that the A20Gate Pass-Through Logic allows specific port 64h Writes to complete without setting this bit.
10	0h RW/1C	**SMI Caused by Port 64 Read (TRAPBY64R):** Indicates if the event occurred. Note that even if the corresponding enable bit is not set in the Bit 2, then this bit will still be active. It is up to the SMM code to use the enable bit to determine the exact cause of the SMI#. Writing a 1 to this bit will clear the latch.
9	0h RW/1C	**SMI Caused by Port 60 Write (TRAPBY60W):** Indicates if the event occurred. Note that even if the corresponding enable bit is not set in the Bit 1, then this bit will still be active. It is up to the SMM code to use the enable bit to determine the exact cause of the SMI#. Writing a 1 to this bit will clear the latch. Note that the A20Gate Pass-Through Logic allows specific port 60h Writes to complete without setting this bit.

Bit Range	Default and Access	Field Name (ID): Description
8	0h RW/1C	**SMI Caused by Port 60 Read (TRAPBY60R):** Indicates if the event occurred. Note that even if the corresponding enable bit is not set in the bit 0, then this bit will still be active. It is up to the SMM code to use the enable bit to determine the exact cause of the SMI#. Writing a 1 to this bit will clear the latch.
7	0h RW	**SMI at End of Pass-through Enable (SMIATENDPS):** May need to cause SMI at the end of a pass-through. Can occur if an SMI is generated in the middle of a pass through, and needs to be serviced later.
6	0h RO	**Pass Through State (PSTATE):** This read-only bit indicates that the state machine is in the middle of an A20GATE pass-through sequence. If software needs to reset this bit, it should set Bit 5 0.
5	0h RW	**A20Gate Pass-Through Enable (A20PASSEN):** When enabled, allows A20GATE sequence Pass-Through function. When enabled, a specific cycle sequence involving writes to port 60h and port 64h does not result in the setting of the SMI status bits.SMI# will not be generated, even if the various enable bits are set.
4	0h RO	Reserved.
3	0h RW	**SMI on Port 64 Writes Enable (s64WEN):** When set, a 1 in bit 11 will cause an SMI event.
2	0h RW	**SMI on Port 64 Reads Enable (s64REN):** When set, a 1 in bit 10 will cause an SMI event.
1	0h RW	**SMI on Port 60 Writes Enable (s60WEN):** When set, a 1 in bit 9 will cause an SMI event.
0	0h RW	**SMI on Port 60 Reads Enable (s60REN):** When set, a 1 in bit 8 will cause an SMI event.

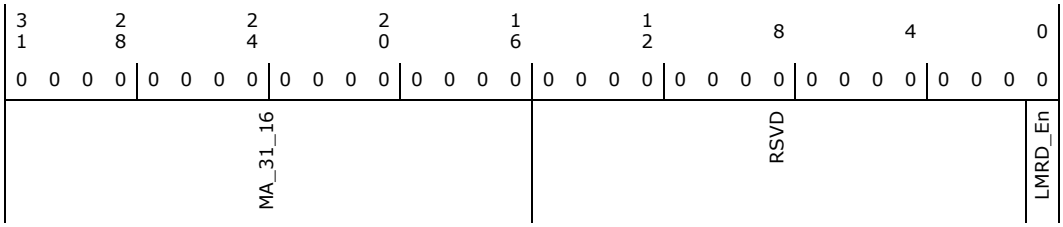
21.8.17 LPC Generic Memory Range (LGMR)—Offset 98h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RW	**Memory Address[31:16] (MA_31_16):** This field specifies a 64 KB memory block anywhere in the 4GB memory space that will be decoded to LPC as standard LPC Memory Cycle if enabled.
15:1	0h RO	Reserved.
0	0h RW	**LPC Memory Range Decode Enable (LMRD_En):** When this bit is set to 1, then the range specified in this register is enabled for decoding to LPC.

21.8.18 FWH ID Select #1 (FS1)—Offset D0h

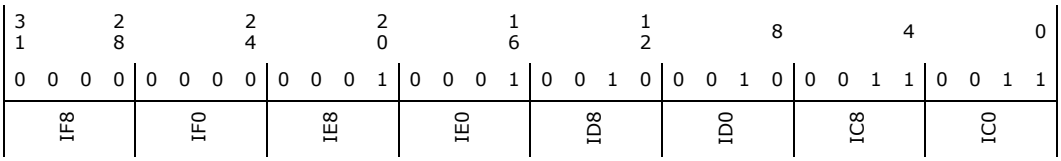
This register contains the IDSEL fields the LPC Bridge uses for memory cycles going to the FWH.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 112233h



Bit Range	Default and Access	Field Name (ID): Description
31:28	0h RO	**F8-FF IDSEL (IF8):** IDSEL to use in FWH cycle for range enabled by BDE.EF8.
27:24	0h RW	**F0-F7 IDSEL (IF0):** IDSEL to use in FWH cycle for range enabled by BDE.EF0.
23:20	1h RW	**E8-EF IDSEL (IE8):** IDSEL to use in FWH cycle for range enabled by BDE.EE8.
19:16	1h RW	**E0-E7 IDSEL (IE0):** IDSEL to use in FWH cycle for range enabled by BDE.EE0.
15:12	2h RW	**D8-DF IDSEL (ID8):** IDSEL to use in FWH cycle for range enabled by BDE.ED8.
11:8	2h RW	**D0-D7 IDSEL (ID0):** IDSEL to use in FWH cycle for range enabled by BDE.ED0.
7:4	3h RW	**C8-CF IDSEL (IC8):** IDSEL to use in FWH cycle for range enabled by BDE.EC8.
3:0	3h RW	**C0-C7 IDSEL (IC0):** IDSEL to use in FWH cycle for range enabled by BDE.EC0.

21.8.19 FWH ID Select #2 (FS2)—Offset D4h

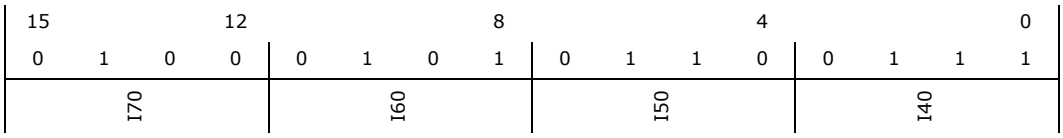
This register contains the additional IDSEL fields the LPC Bridge uses for memory cycles going to the FWH.

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 0

Default: 4567h



Bit Range	Default and Access	Field Name (ID): Description
15:12	4h RW	**70-7F IDSEL (I70):** IDSEL to use in FWH cycle for range enabled by BDE.E70.
11:8	5h RW	**60-6F IDSEL (I60):** IDSEL to use in FWH cycle for range enabled by BDE.E60.
7:4	6h RW	**50-5F IDSEL (I50):** IDSEL to use in FWH cycle for range enabled by BDE.E50.
3:0	7h RW	**40-4F IDSEL (I40):** IDSEL to use in FWH cycle for range enabled by BDE.E40.

21.8.20 BIOS Decode Enable (BDE)—Offset D8h

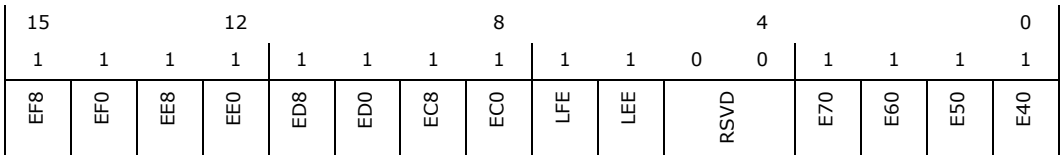
Note that this register effects the BIOS decode regardless of whether the BIOS is resident on LPC or SPI. The concept of Feature Space does not apply to SPI-based flash. PCH simply decodes these ranges as memory accesses when enabled for the SPI flash interface.

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 0

Default: FFCFh



Bit Range	Default and Access	Field Name (ID): Description
15	1h RO	**F8-FF Enable (EF8):** Enables decoding of 512K of the following BIOS range: Data space: FFF80000h - FFFFFFFFh Feature space: FFB80000h - FFBFFFFFh
14	1h RW	**F0-F8 Enable (EF0):** Enables decoding of 512K of the following BIOS range: Data space: FFF00000h - FFF7FFFFh Feature space: FFB00000h - FFB7FFFFh
13	1h RW	**E8-EF Enable (EE8):** Enables decoding of 512K of the following BIOS range: Data space: FFE80000h - FFEFFFFFh Feature space: FFA80000h - FFAFFFFFh
12	1h RW	**E0-E8 Enable (EE0):** Enables decoding of 512K of the following BIOS range: Data space: FFE00000h - FFE7FFFFh Feature Space: FFA00000h - FFA7FFFFh
11	1h RW	**D8-DF Enable (ED8):** Enables decoding of 512K of the following BIOS range: Data space: FFD80000h - FFDFFFFFh Feature space: FF980000h - FF9FFFFFh
10	1h RW	**D0-D7 Enable (ED0):** Enables decoding of 512K of the following BIOS range: Data space: FFD00000h - FFD7FFFFh Feature space: FF900000h - FF97FFFFh
9	1h RW	**C8-CF Enable (EC8):** Enables decoding of 512K of the following BIOS range: Data space: FFC80000h - FFCFFFFFh Feature space: FF880000h - FF8FFFFFh
8	1h RW	**C0-C7 Enable (EC0):** Enables decoding of 512K of the following BIOS range: Data space: FFC00000h - FFC7FFFFh Feature space: FF800000h - FF87FFFFh
7	1h RW	**Legacy F Segment Enable (LFE):** This enables the decoding of the legacy 64KB range at F0000h - FFFFFh Note that decode for the BIOS legacy F segment is enabled by the LFE bit only, it is not affected by the GEN_PMCON_1.iA64_EN bit.
6	1h RW	**Legacy E Segment Enable (LEE):** This enables the decoding of the legacy 64KB range at E0000h - EFFFFh Note that decode for the BIOS legacy E segment is enabled by the LEE bit only, it is not affected by the GEN_PMCON_1.iA64_EN bit.
5:4	0h RO	Reserved.
3	1h RW	**70-7F Enable (E70):** Enables decoding of 1MB of the following BIOS range: Data space: FF700000h - FF7FFFFFh Feature space: FF300000h - FF3FFFFFh
2	1h RW	**60-6F Enable (E60):** Enables decoding of 1MB of the following BIOS range: Data space: FF600000h - FF6FFFFFh Feature Space: FF200000h - FF2FFFFFh
1	1h RW	**50-5F Enable (E50):** Enables decoding of 1MB of the following BIOS range: Data space: FF500000h - FF5FFFFFh Feature space: FF100000h - FF1FFFFFh
0	1h RW	**40-4F Enable (E40):** Enables decoding of 1MB of the following BIOS range: Data space: FF400000h - FF4FFFFFh Feature space: FF000000h - FF0FFFFFh

21.8.21 BIOS Control (BC)—Offset DCh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 0

Default: 20h

7			4				0
0	0	1	0	0	0	0	0
BILD	BBS	EISS	TS	SRC		LE	WPD

Bit Range	Default and Access	Field Name (ID): Description
7	0h RW/1L	**BIOS Interface Lock-Down (BILD):** When set, prevents BC.TS and BC.BBS from being changed. This bit can only be written from 0 to 1 once. **BIOS Note:** This bit is not backed up in the RTC well. This bit should also be set in the BUC register in the RTC device to record the last state of this value following a cold reset.
6	0h RW/L	**Boot BIOS Strap (BBS):** This field determines the destination of accesses to the BIOS memory range. For the default, Functional Strap section of Signal Description chapter for details. Bits Description 0 SPI 1 LPC When SPI or LPC is selected, the range that is decoded is further qualified by other configuration bits described in the respective sections. The value in this field can be overwritten by software as long as the BIOS Interface Lock-Down is not set.
5	1h RW/L	**Enable InSMM.STS (EISS):** When this bit is set, the BIOS region is not writable until SMM sets the InSMM.STS bit. Today BIOS Flash is writable if WPD is a 1. If this bit [5] is set, then WPD must be a 1 and InSMM.STS (0xFED3_0880[0]) must be 1 also. If this bit [5] is clear, then BIOS is writable based only on WPD = 1 and the InSMM.STS is a don't care.
4	0h RO	**Top Swap (TS):** When set, PCH will invert either A16, A17, or A18 for cycles going to the BIOS space (but not the feature space) in the FWH. When cleared, PCH will not invert A16. If booting from LPC (FWH), then the Boot Blook size is 64 KB and A16 is inverted if Top Swap is enabled. If booting from SPI, then the BOOT_BLOCK_SIZE soft strap determines if A16, A17, or A18 should be inverted if Top Swap is enabled. *If PCH is strapped for Top-Swap (GNT[3]# is low at rising edge of PWROK), then this bit cannot be cleared by software. The strap jumper should be removed and the system rebooted. **BIOS Note:** 1. This bit provides a read-only path to view the state of the Top Swap strap. It is backed up and driven from the RTC well. BIOS will need to program the corresponding register in the RTC Controller (in RTC well), which will be reflected in this register. 2. The Register portion of the Top Swap is lockable by the BIOS Interface Lockdown Bit (BC.BILD) but unlockable by SPI Flash Protected Range and Top Swap Override (uCode.PRR_TS_OVR).
3:2	0h RW	LPC or eSPI Strap (LPC_eSPI). Bit 3 is reserved. Bit 2 is read only and reflects whether eSPI or LPC is enabled. If 0, then D31:F0 is LPC. If 1, then D31:F0 is eSPI.
1	0h RW/1L	**Lock Enable (LE):** When set, setting the WP bit will cause SMI. When cleared, setting the WP bit will not cause SMI. Once set, this bit can only be cleared by a PLTRST#. When this bit is set, EISS - bit [5] of this register is locked down.
0	0h RW	**Write Protect Disable (WPD):** When set, access to the BIOS space is enabled for both read and write cycles to BIOS. When cleared, only read cycles are permitted to the FWH or SPI flash. When this bit is written from a 0 to a 1 and the LE bit is also set, an SMI# is generated. This ensures that only SMM code can update BIOS.

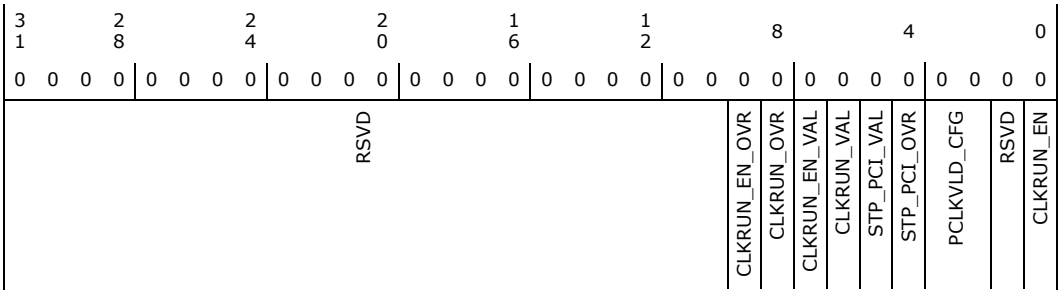
21.8.22 PCI Clock Control (PCCTL)—Offset E0h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:10	0h RO	Reserved.
9	0h RO	**CLKRUN# Buffer Enable Override (CLKRUN_EN_OVR):** When set to '1', SW is in control of the CLKRUN# buffer enable and the value in CLKRUN_EN_VAL will be propagated to the output buffer enable. When this bit is '0', HW will determine the value of the buffer enable.
8	0h RO	**CLKRUN# Override (CLKRUN_OVR):** When set to '1', SW is in control of the CLKRUN# pin and the value in CLKRUN_VAL will be propagated to the output pin. When this bit is '0', HW will determine the value of the pin.
7	0h RO	**CLKRUN# Buffer Enable Value (CLKRUN_EN_VAL):** Either HW or SW may own control of the CLKRUN# pin. This bit provides the value to drive on the active low CLKRUN# buffer enable if CLKRUN_EN_OVR is set to '1'.
6	0h RO	**CLKRUN# Pin Output Value (CLKRUN_VAL):** Either HW or SW may own control of the CLKRUN# pin. This bit provides the value to drive on the pin if CLKRUN_OVR is set to '1'.
5	0h RO	**Stop PCI# Value (STP_PCI_VAL):** Either Hardware or Software may own control of the internal STP_PCI#. This bit provides the value to drive on the STP_PCI# if STP_PCI_OVR is set to 1. ***Note:*** SW cannot control the STP_PCI# pin while PLTRST# is asserted (the pin will be at its reset default value).
4	0h RO	**Stop PCI# Override (STP_PCI_OVR):** This field determines the relationship between the internally broadcast indication of the external PCI clock being valid vs. the STP_PCI# pin. Encodings: 00: 1 flop stage of delay from STP_PCI# (default). Expected setting for Full Integration Mode 01: No delay (edges match STP_PCI#). For CK505 legacy mode 10: 2 flop stages of delay from STP_PCI# 11: Tie high (indicate that PCI clock is always valid)

Bit Range	Default and Access	Field Name (ID): Description
3:2	0h RW	**PCI Clock Valid Configuration (PCLKVLD_CFG):** This field determines the relationship between the internally broadcast indication of the external PCI clock being valid vs. the STP_PCI# pin. Encodings: 00: 1 flop stage of delay from STP_PCI# (default) 01: No delay (edges match STP_PCI# 10: 2 flop stages of delay from STP_PCI# 11: Tie high (indicate that PCI clock is always valid)
1	0h RO	Reserved.
0	0h RW	**Clock Run Enable (CLKRUN_EN):** Enables the CLKRUN# logic to stop the PCI clocks. If the SLP_EN bit is set, then the Intel PCH will drive CLKRUN# low. This will keep the PCI and LPC clocks running on the way to the sleeping state. This is required to meet an LPC specification. This does not necessarily mean that the CLKRUN_EN bit is forced low when SLP_EN is set. Even though the CLKRUN# signal will be low when SLP_EN is set, the state of the CLKRUN_EN bit is ignored when SLP_EN bit is set. This gives flexibility in the implementation.

21.9 LPC PCR Registers Summary

These registers are within the PCH Private Configuration Space which is accessible through the PCH Sideband Interface. They can be accessed via (SBREG_BAR + PortID + Register Offset).

Table 21-8. Summary of LPC PCR Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
3418h	341Bh	General Control and Function Disable (GCFD)—Offset 3418h	0h

21.9.1 General Control and Function Disable (GCFD)—Offset 3418h

Access Method

Type: MSG Register
(Size: 32 bits)

Device: 31
Function: 0

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																														eSPI	LPC_BD

Bit Range	Default and Access	Field Name (ID): Description
31:2	0h RO	Reserved.
1	0h RO	**eSPI Enable Pin Strap (eSPI):** This field determines the destination of accesses to the D31:F0 and related Fixed and Variable I/O and Memory decode ranges, including BIOS memory range. 0: LPC is the D31:F0 target. 1: eSPI is the D31:F0 target. ***Note:*** This field, along with BC.BBS strap setting determines the final BIOS Boot Location.
0	0h RW	**LPC Bridge Disable (LPC_BD):** When set, the LPC bridge is disabled. When disabled the following spaces will no longer be decoded by the LPC bridge: 1) D31:F0 PCI Configuration space 2) Memory cycles below 16MB (1000000h) 3) I/O cycles below 64kB (10000h)

§

22 PCI Express*

22.1 Acronyms and Register Attributes/Modifiers

Acronyms	Description
LTR	Latency tolerance Reporting
SSC	Spread Spectrum Clocking
SRIS	Separate Reference Clocking with Independent SSC
PCIe*	PCI Express
Intel® RSTe	Intel® Rapid Storage Technology Enterprise

Attribute	Description
RO	**Read Only:** These bits can be read by Software, but writes have no effect. The value of the bit is determined by hardware only.
RW	**Read/Write:** These register bits can be read by software and modified by software
RW1C	**Read/Write 1Clear**. These bits can be read from and written to by software. Writing a "0" to these bits has no effect. Writing a "1" clears the bit.
RV	**Reserved**: These bits are reserved. Their value must be maintained by software. When writing these bits, software must preserve the value read.
Modifier	**Description**
S	**Sticky**: These bits are only reinitialized to their default state by a PWRGD reset.
-K	**Key:** This bit is used to lock other bits
-L	**Lock:** This bit can will become read only once locked by another bit or logic
-O	**Once:** After reset, these bits can only be written to once. After that they are read only.
-V	**Variant:** The value of the bit can be updated by hardware.

22.2 References

Specification	Location
PCI Express Base Specification*	http://www.pcisig.com/specifications
PCI Local Bus Specification	
PCI Power Management Specification	

22.3 Overview

The PCIe ports on the Intel® C620 Series Chipset are arranged in three groups with separate functions:

- One group of PCIe ports are root ports, that are used to connect the Intel® C620 Series Chipset to downstream PCIe devices outside the chip (down devices, slots, etc.) These PCIe ports are always available and are selected by strapping options. Data from these PCIe ports is transferred to the CPU over the DMI bus.
- The second set of PCIe functions are endpoints. Their job is to provide a connection to the 10/1 GbE Ethernet controller and Intel® QuickAssist Technology engines. These downstream ports come in two blocks. One block is a x16 port. All 10 GbE traffic is sent to the CPU over this port. Data from the Intel QuickAssist Technology

engines is also sent over this port to the CPU. In addition to this x16 uplink, 8 of the HSIO pins have a PCIe endpoint function multiplexed on them, too. When enabled, this permits the PCH to have x24 PCIe Gen3 lanes connecting the PCH to the CPU. These extra x8 PCIe uplinks are solely for the use of the Intel QuickAssist Technology engines.

- The third set of PCIe functions are Virtual Switch Ports (VSP). Their job is to provide connection between the PCIe Upstream Ports and the 10/1 GbE Ethernet controller and Intel QuickAssist Technology endpoints.

22.4 PCIe Root Port

22.4.1 PCIe Root Port overview

- Up to 16 PCIe Ports and 20 PCIe Lanes, with transfer rates up to 8 GT/s (Gen3)
- ExpressCard 1.0 module-based hot-plug supported
- Dynamic Link Throttling
- Port 8xh Decode
- Separate Reference Clock with Independent Spread Spectrum Clocking (SSC)
- Latency Tolerance Reporting
- x4 static end-to-end Lane Reversal
- Access Control Services
- Alternative Routing ID
- Autonomous Link Width Negotiation as a target
- Advanced Error Reporting
- Polarity Inversion
- Configurable 128B or 256B Maximum Data Payload
- Subtractive Decode is not Supported
- Intel® Rapid Storage Technology enterprise (Intel® RSTe) for PCIe Gen3 Storage Support

22.4.2 Signal Description

Name	Type	Description
USB3_7_PCIE0_TXN/P	O	PCI Express Differential Transmit Pair 0 PCI Express transmit pair 0, muxed with USB3 port 7
USB3_8_PCIE1_TXN/P	O	PCI Express Differential Transmit Pair 1 PCI Express transmit pair 1, muxed with USB3 port 8
USB3_9_PCIE2_TXN/P	O	PCI Express Differential Transmit Pair 2 PCI Express transmit pair 2, muxed with USB3 port 9
USB3_10_PCIE3_TXN/P	O	PCI Express Differential Transmit Pair 3 PCI Express transmit pair 3, muxed with USB3 port 10
PCIE4_GBE_TXN/P	O	PCI Express Differential Transmit Pair 4 PCI Express transmit pair 4, muxed with the output of the 1 GbE MAC
PCIE5_GBE_TXN/P	O	PCI Express Differential Transmit Pair 5 PCI Express transmit pair 5, muxed with the output of the 1 GbE MAC

Name	Type	Description
PCIE6_SSATA0_TXN/P	O	PCI Express Differential Transmit Pair 6 PCI Express transmit pair 6, muxed with SSATA port 0
PCIE7_SSATA1_TXN/P	O	PCI Express Differential Transmit Pair 7 PCI Express transmit pair 7, muxed with SSATA port 1
PCIE8_SSATA2_GBE_TXN/P	O	PCI Express Differential Transmit Pair 8 PCI Express Transmit pair 7, muxed with SSATA port 2 and the output of the 1 GbE MAC
PCIE9_SSATA3_TXN/P	O	PCI Express Differential Transmit Pair 9 PCI Express transmit pair 9, muxed with SSATA port 3
PCIE10_SSATA4_TXN/P	O	PCI Express Differential Transmit Pair 10 PCI Express transmit pair 10, muxed with SSATA port 4
PCIE11_SSATA5_GBE_TXN/P	O	PCI Express Differential Transmit Pair 11 PCI Express Transmit pair 11, muxed with SSATA port 5and the output of the 1GbE MAC
PCIE12_UP0_SATA0_TXN/P	O	PCI Express Differential Transmit Pair 12 PCI Express Transmit pair 12, muxed with PCI Express Uplink Port 0 and SATA port 0
PCIE13_UP1_SATA1_TXN/P	O	PCI Express Differential Transmit Pair 13 PCI Express Transmit pair 13, muxed with PCI Express Uplink Port 1 and SATA port 1
PCIE14_UP2_SATA2_TXN/P	O	PCI Express Differential Transmit Pair 14 PCI Express Transmit pair 14, muxed with PCI Express Uplink Port 2 and SATA port 2
PCIE15_UP3_SATA3_TXN/P	O	PCI Express Differential Transmit Pair 15 PCI Express Transmit pair 15, muxed with PCI Express Uplink Port 3 and SATA port 3
PCIE16_UP4_SATA4_TXN/P	O	PCI Express Differential Transmit Pair 16 PCI Express Transmit pair 16, muxed with PCI Express Uplink Port 4 and SATA port 4
PCIE17_UP5_SATA5_TXN/P	O	PCI Express Differential Transmit Pair 17 PCI Express Transmit pair 17, muxed with PCI Express Uplink Port 5 and SATA port 5
PCIE18_UP6_SATA6_TXN/P	O	PCI Express Differential Transmit Pair 18 PCI Express Transmit pair 18, muxed with PCI Express Uplink Port 6 and SATA port 6
PCIE19_UP7_SATA7_TXN/P	O	PCI Express Differential Transmit Pair 19 PCI Express Transmit pair 19, muxed with PCI Express Uplink Port 7 and SATA port 7
USB3_7_PCIE0_RXN/P	I	PCI Express Differential Receive Pair 0 PCI Express Receive pair 0, muxed with USB3 port 7
USB3_8_PCIE1_RXN/P	I	PCI Express Differential Receive Pair 1 PCI Express Receive pair 1, muxed with USB3 port 8
USB3_9_PCIE2_RXN/P	I	PCI Express Differential Receive Pair 2 PCI Express Receive pair 2, muxed with USB3 port 9
USB3_10_PCIE3_RXN/P	I	PCI Express Differential Receive Pair 3 PCI Express Receive pair 3, muxed with USB3 port 10
PCIE4_GBE_RXN/P	I	PCI Express Differential Receive Pair 4 PCI Express Receive pair 4, muxed with the output of the 1 GbE MAC
PCIE5_GBE_RXN/P	I	PCI Express Differential Receive Pair 5 PCI Express Receive pair 5, muxed with the output of the 1 GbE MAC
PCIE6_SSATA0_RXN/P	I	PCI Express Differential Receive Pair 6 PCI Express Receive pair 6, muxed with SSATA port 0

Name	Type	Description
PCIE7_SSATA1_RXN/P	I	PCI Express Differential Receive Pair 7 PCI Express Receive pair 7, muxed with SSATA port 1
PCIE8_SSATA2_GBE_RXN/P	I	PCI Express Differential Receive Pair 8 PCI Express Receive pair 7, muxed with SSATA port 2 and the output of the 1GbE MAC
PCIE9_SSATA3_RXN/P	I	PCI Express Differential Receive Pair 9 PCI Express Receive pair 9, muxed with SSATA port 3
PCIE10_SSATA4_RXN/P	I	PCI Express Differential Receive Pair 10 PCI Express Receive pair 10, muxed with SSATA port 4
PCIE11_SSATA5_GBE_RXN/P	I	PCI Express Differential Receive Pair 11 PCI Express Receive pair 11, muxed with SSATA port 5and the output of the 1 GbE MAC
PCIE12_UP0_SATA0_RXN/P	I	PCI Express Differential Receive Pair 12 PCI Express Receive pair 12, muxed with PCI Express Uplink Port 0 and SATA port 0
PCIE13_UP1_SATA1_RXN/P	I	PCI Express Differential Receive Pair 13 PCI Express Receive pair 13, muxed with PCI Express Uplink Port 1 and SATA port 1
PCIE14_UP2_SATA2_RXN/P	O	PCI Express Differential Receive Pair 14 PCI Express Receive pair 14, muxed with PCI Express Uplink Port 2 and SATA port 2
PCIE15_UP3_SATA3_RXN/P	O	PCI Express Differential Receive Pair 15 PCI Express Receive pair 15, muxed with PCI Express Uplink Port 3 and SATA port 3
PCIE16_UP4_SATA4_RXN/P	O	PCI Express Differential Receive Pair 16 PCI Express Receive pair 16, muxed with PCI Express Uplink Port 4 and SATA port 4
PCIE17_UP5_SATA5_RXN/P	O	PCI Express Differential Receive Pair 17 PCI Express Receive pair 17, muxed with PCI Express Uplink Port 5 and SATA port 5
PCIE18_UP6_SATA6_RXN/P	O	PCI Express Differential Receive Pair 18 PCI Express Receive pair 18, muxed with PCI Express Uplink Port 6 and SATA port 6
PCIE19_UP7_SATA7_RXN/P	O	PCI Express Differential Receive Pair 19 PCI Express Receive pair 19, muxed with PCI Express Uplink Port 7 and SATA port 7
PCIE_RCOMPP PCIE_RCOMPN	I	Impedance Compensation Inputs

22.4.3 I/O Signal Planes and States

Table 22-1. Power Plane and States for PCI Express* Signals

Signal Name	Type	Power Plane	During Reset	Immediately After Reset	S3/S4/S5	Deep Sx
PCIE[20:1]_TXP/N	O	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	Off
PCIE[20:1]_RXP/N	I	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	Off
PCIE_RCOMPP PCIE_RCOMPN	I	Primary	Un-driven	Un-driven	Un-driven	Off

Note: For brevity, the PCIe root port names have been shortened.

PCIE1_RXP\N pins transition from un-driven to internal pull-down during reset.

22.4.4 PCI Express Port Support Feature Details

Table 22-2. PCI Express Port Feature Details

Max Device (Ports)	Max Lanes	PCIe Gen Type	Encoding	Transfer Rate (MT/s)	Theoretical Max Bandwidth (GB/s)		
					x1	x2	x4
16	20	1	8b/10b	2500	0.25	0.50	1.00
		2	8b/10b	5000	0.50	1.00	2.00
		3	128b/130b	8000	1.00	2.00	3.94

Notes:
1. Theoretical Maximum Bandwidth (GB/s) = ((Transfer Rate * Encoding * # PCIe Lanes) /8) / 1000
   — Gen3 Example: = ((8000 * 128/130* 4) / 8) / 1000 = 3.94 GB/s
2. When GbE is enabled on a PCIe Root Port, the Max Device (Ports) value listed is reduced by a factor of 1

Table 22-3. PCI Express Link Configurations Supported

PCIe Link Config	PCI Express Lanes																			
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1x4	P1				P5				P9				P13				P17			
2x2	P1		P3		P5		P7		P9		P11		P13		P15		P17		P19	
1x2 + 2x1	P1		P3	P4	P5		P7	P8	P9		P11	P12	P13		P15	P16	P17		P19	P20
4x1	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20

Notes:
1. P# refers to a specific PCH PCI Express Root Port #; for example P3 = PCH PCI Express Root Port 3
2. A PCIe Lane is composed of a single pair of Transmit (TX) and Receive (RX) differential pairs, for a total of four data wires per PCIe Lane (such as, PCIE[3]_TXP/ PCIE[3]_TXN and PCIE[3]_RXP/ PCIE[3]_RXN make up PCIe Lane 3). A connection between two PCIe devices is known as a PCIe Link, and is built up from a collection of one or more PCIe Lanes which make up the width of the link (such as bundling 2 PCIe Lanes together would make a x2 PCIe Link). A PCIe Link is addressed by the lowest number PCIe Port it connects to in the PCH (such as a x2 PCIe Link connected to PCIe Ports 3 and 4 would be called x2 PCIe Port 3). This lowest number PCIe Port in the PCIe Link is known as the PCIe Root Port.
3. The PCIe Ports can be configured independently from one another but the max number of configured Devices (Ports) must not be exceeded.
4. Unidentified Ports within a PCIe Link Configuration are disabled but their physical lanes are used for the identified Port.
5. GbE can be mapped to PCIe Root Ports 4, 5, 9, 12, or 13 within their respective PCIe Link configurations. When GbE is enabled on a PCIe Root Port, there can be at most up to a max of 15 Device (Ports) enabled
6. The PCH PCIe Root Ports can be configured to map to any of the SRCCLKRQ# PCIe clock request signals and the CLKOUT_SRC_P/N PCIe differential signal pairs covered in the “PCH and System Clocks” Chapter
7. Reference and understand the PCIe High Speed I/O Muxing details covered in the “Flexible I/O” Chapter

22.4.5 Interrupt Generation

The root port generates interrupts on behalf of hot plug, power management, link bandwidth management, Link Equalization Request and link error events, when enabled. These interrupts can either be pin-based, or can be MSI, when enabled.

When an interrupt is generated via the legacy pin, the pin is internally routed to the PCH interrupt controllers. The pin that is driven is based upon the setting of the STRPFUSECFG.PXIP configuration registers.

Table 22-4 summarizes interrupt behavior for MSI and wire-modes. In the table “bits” refers to the hot-plug and PME interrupt bits.

Table 22-4. MSI Versus PCI IRQ Actions

Interrupt Register	Wire-Mode Action	MSI Action
All bits 0	Wire inactive	No action
One or more bits set to 1	Wire active	Send message
One or more bits set to 1, new bit gets set to 1	Wire active	Send message
One or more bits set to 1, software clears some (but not all) bits	Wire active	Send message
One or more bits set to 1, software clears all bits	Wire inactive	No action
Software clears one or more bits, and one or more bits are set on the same clock	Wire active	Send message

22.4.6 Power Management

22.4.6.1 S3/S4/S5 Support

Software initiates the transition to S3/S4/S5 by performing an I/O write to the Power Management Control register in the PCH. After the I/O write completion has been returned to the CPU the Power Management Controller will signal each root port to send a PME_Turn_Off message on the downstream link. The device attached to the link will eventually respond with a PME_TO_Ack followed by sending a PM_Enter_L23 DLLP (Data Link Layer Packet) request to enter L23. The Express ports and Power Management Controller take no action upon receiving a PME_TO_Ack. When all the Express port links are in state L23, the Power Management Controller will proceed with the entry into S3/S3/S5.

Prior to entering S3, software is required to put each device into $D3_{HOT.}$ When a device is put into $D3_{HOT,}$ it will initiate entry into a L1 link state by sending a PM_Enter_L1 DLLP. Under normal operating conditions when the root ports sends the PME_Turn_Off message, the link will be in state L1. However, when the root port is instructed to send the PME_Turn_Off message, it will send it whether or not the link was in L1. Endpoints attached to the PCH can make no assumptions about the state of the link prior to receiving a PME_Turn_Off message.

22.4.6.2 Resuming from Suspended State

The root port contains enough circuitry in the suspend well to detect a wake event through the WAKE# signal and to wake the system. When WAKE# is detected asserted, an internal signal is sent to the power management controller of the PCH to cause the system to wake up. This internal message is not logged in any register, nor is an interrupt/GPE generated due to it.

22.4.6.3 Device Initiated PM_PME Message

When the system has returned to a working state from a previous low power state, a device requesting service will send a PM_PME message continuously, until acknowledged by the root port. The root port will take different actions depending upon whether this is the first PM_PME that has been received, or whether a previous message has been received but not yet serviced by the operating system.

If this is the first message received (RSTS.PS), the root port will set RSTS.PS, and log the PME Requester ID into RSTS.RID. If an interrupt is enabled using RCTL.PIE, an interrupt will be generated. This interrupt can be either a pin or an MSI if MSI is enabled using MC.MSIE. See Section 22.4.6.4 for SMI/SCI generation.

If this is a subsequent message received (RSTS.PS is already set), the root port will set RSTS.PP. No other action will be taken.

When the first PME event is cleared by software clearing RSTS.PS, the root port will set RSTS.PS, clear RSTS.PP, and move the requester ID into RSTS.RID.

If RCTL.PIE is set, an interrupt will be generated. If RCTL.PIE is not set, a message will be sent to the power management controller so that a GPE can be set. If messages have been logged (RSTS.PS is set), and RCTL.PIE is later written from a 0b to a 1b, an interrupt will be generated. This last condition handles the case where the message was received prior to the operating system re-enabling interrupts after resuming from a low power state.

22.4.6.4 SMI/SCI Generation

Interrupts for power management events are not supported on legacy operating systems. To support power management on non-PCI Express aware operating systems, PM events can be routed to generate SCI. To generate SCI, MPC.PMCE must be set. When set, a power management event will cause SMSCS.PMCS to be set.

Additionally, BIOS workarounds for power management can be supported by setting MPC.PMME. When this bit is set, power management events will set SMSCS.PMMS, and SMI# will be generated. This bit will be set regardless of whether interrupts or SCI is enabled. The SMI# may occur concurrently with an interrupt or SCI.

When operating at PCIe 8 Gb/s, Link Equalization Request can also be routed to generate SCI or SMI. The intention8 for the SCI/SMI is to invoke the proprietary software to diagnose the reason behind the Link Equalization Request interrupt and take the proper link recovery path, which may include software re-performing link equalization. Root Ports do not support the hardware mechanism to service the Link Equalization Request from the device.

22.4.6.5 Latency Tolerance Reporting

The root port supports the extended Latency Tolerance Reporting capability. LTR provides a means for device endpoints to dynamically report their service latency requirements for memory access to the root port. Endpoint devices should transmit a new LTR message to the root port each time its latency tolerance changes (and initially during boot). The PCH uses the information to make better power management decisions. The processor uses the worst case tolerance value communicated by the PCH to optimize C-state transitions. This results in better platform power management without impacting endpoint functionality.

Note: Endpoint devices that support LTR must implement the reporting and enable mechanism detailed in the PCIe Latency Tolerance Reporting Engineering Change Notice.

22.4.7 Dynamic Link Throttling

Root Port supports dynamic link throttling as a mechanism to help lower the overall component power, ensuring that the component never operates beyond the thermal limit of the package. Dynamic link throttling is also used as a mechanism for ensuring that the ICC_{max} current rating of the voltage regulator is never exceeded. The target response time for this particular usage model is < 100 us.

If dynamic link throttling is enabled, the link will be induced by the Root Port to enter TxL0s and RxL0s based on the throttle severity indication received. To induce the link into TxL0s, new TLP requests and opportunistic flow control update will be blocked. Eventually, in the absence of TLP and DLLP requests, the transmitter side of the link will enter TxL0s.

The periodic flow control update, as required by the PCI Express Base Specification is not blocked. However, the flow control credit values advertised to the component on the other side of the link will not be incremented, even if the periodic flow control update packet is sent. Once the other component runs out of credits, it will eventually enter TxL0s, resulting in the local receiver entering RxL0s.

Each of the Root Ports receives four throttle severity indications; T0, T1, T2 and T3. The throttling response for each of the four throttle severity levels can be independently configured in the Root Port TNPT.TSLxM register fields. This allows the duty cycle of the Throttling Window to be varied based on the severity levels, when dynamic link throttling is enabled.

A Throttling Window is defined as a period of time where the duty cycle of throttling can be specified. A Throttling Window is sub-divided into a Throttling Zone and a Non-Throttling Zone. The period of the Throttling Zone is configurable through the TNPT.TT field. Depending on the throttle severity levels, the throttling duration specified by the TNPT.TT field will be multiplied by the multipliers configurable through TNPT.TSLxM.

The period of the Throttling Window is configurable through the TNPT.TP field. The Throttling Window is always referenced from the time a new Throttle State change indication is received by the Root Port or from the time the throttling is enabled by the configuration register. The Throttling Window and Throttling Zone timers continue to behave the same as in L0 or L0s even if the link transitions to other LTSSM states, except for L1, L23_Rdy and link down. For L1 case, the timer is allowed to be stopped and hardware is allowed to re-start the Throttling Window and the corresponding Throttling Zone timers on exit from L1.

22.4.8 Port 8xh Decode

The PCIe root ports will explicitly decode and claim I/O cycles within the 80h – 8Fh range when MPC.P8XDE is set. The claiming of these cycles are not subjected to standard PCI I/O Base/Limit and I/O Space Enable fields. This allows a POST-card to be connected to the Root Port either directly as a PCI Express device or through a PCI Express to PCI bridge as a PCI card.

Any I/O reads or writes will be forwarded to the link as it is. The device will need to be able to return the previously written value, on I/O read to these ranges. BIOS must ensure that at any one time, no more than one Root Port is enabled to claim Port 8xh cycles.

22.4.9 Separate Reference Clock with Independent SSC

The current PCI-SIG PCIe External Cabling form factor specification defines the reference clock as part of the signals delivered through the cable. Inclusion of the reference clock in the cable requires an expensive shielding solution to meet EMI requirements.

The need for an inexpensive PCIe cabling solution for PCIe SSDs requires a new cabling form factor that supports non-common clock mode with spread spectrum enabled, such that the reference clock does not need to be part of the signals delivered through the

cable. This new clock mode requires the components on both sides of a link to tolerate a much higher ppm tolerance of ~5600 ppm compared to the PCIe Base Specification defined as 600 ppm.

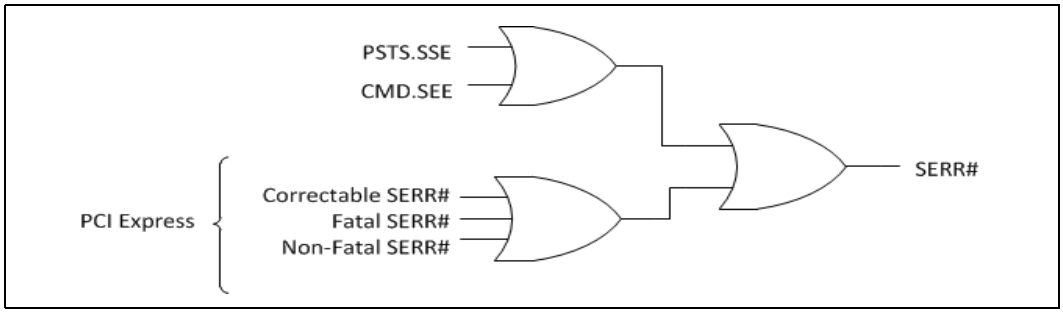
Soft straps are needed as a method to configure the port statically to operate in the new mode. This new mode is only enabled if the SSD connector is present on the motherboard, where the SSD connector does not include the reference clock. No change is being made to PCIe add-in card form factors and solutions.

ASPM L0s is not supported in this new form factor. The L1 exit latency advertised to software would be increased to 10 us. The root port does not support Lower SKP Ordered Set generation and reception feature defined in SRIS ECN.

22.4.10 SERR# Generation

SERR# may be generated using two paths—through PCI mechanisms involving bits in the PCI header, or through PCI Express* mechanisms involving bits in the PCI Express capability structure.

Figure 22-1. Generation of SERR# to Platform



22.4.11 Hot-Plug

All PCIe Root Ports support Express Card 1.0 based hot-plug that performs the following:

- Presence Detect and Link Active Changed Support
- Interrupt Generation Support

22.4.11.1 Presence Detection

When a module is plugged in and power is supplied, the physical layer will detect the presence of the device, and the root port sets SLSTS.PDS and SLSTS.PDC. If SLCTL.PDE and SLCTL.HPE are both set, the root port will also generate an interrupt.

When a module is removed (using the physical layer detection), the root port clears SLSTS.PDS and sets SLSTS.PDC. If SLCTL.PDE and SLCTL.HPE are both set, the root port will also generate an interrupt.

22.4.11.2 SMI/SCI Generation

Interrupts for power-management events are not supported on legacy operating systems. To support power-management on non-PCI Express aware operating systems, power management events can be routed to generate SCI. To generate SCI, MPC.HPCE must be set. When set, enabled hot-plug events will cause SMSCS.HPCS to be set.

Additionally, BIOS workarounds for hot-plug can be supported by setting MPC.HPME. When this bit is set, hot-plug events can cause SMI status bits in SMSCS to be set. Supported hot-plug events and their corresponding SMSCS bit are:

- Presence Detect Changed – SMSCS.HPPDM
- Link Active State Changed – SMSCS.HPLAS

When any of these bits are set, SMI# will be generated. These bits are set regardless of whether interrupts or SCI is enabled for hot-plug events. The SMI# may occur concurrently with an interrupt or SCI.

22.5 PCIe Upstream Port

22.5.1 Features

22.5.1.1 Supported Features

- PCI Express 3.0 specification Compliant
- Support for PCIe Gen3, Gen2 and Gen1 speeds
- VC0 support only
- 512B maximum payload
- 2048 maximum read request
- Malformed packet checks
- ARI support
- Advanced Error Reporting
- Role Based Error Reporting
- ASPM L1
- L1, L3
- D0, $D3_{HOT,}$ and $D3_{COLD}$
- Polarity Inversion
- Lane reversal
- Support of legacy INTx Interrupt messaging over PCIe
- Support for MSI, MSI-X interrupt messaging over PCIe
- NPx16 supports 1x16, 1x8, 1x4 and 1x1 widths
- NPx8 supports 1x8, 1x4 and 1x1 widths.

22.5.1.2 Features Not Supported

- 1x2 link width
- L0s
- Isochronous services
- P2P support
- Atomic ops
- Lock cycles
- Multicast

22.5.2 Signal List

Table 22-5. NPx16 Port

Name	Type	Description
PCIEUP_0_TXN/P	O	PCI Express Uplink Differential Transmit Pair 0
PCIEUP_1_TXN/P	O	PCI Express Uplink Differential Transmit Pair 1
PCIEUP_2_TXN/P	O	PCI Express Uplink Differential Transmit Pair 2
PCIEUP_3_TXN/P	O	PCI Express Uplink Differential Transmit Pair 3
PCIEUP_4_TXN/P	O	PCI Express Uplink Differential Transmit Pair 4
PCIEUP_5_TXN/P	O	PCI Express Uplink Differential Transmit Pair 5
PCIEUP_6_TXN/P	O	PCI Express Uplink Differential Transmit Pair 6
PCIEUP_7_TXN/P	O	PCI Express Uplink Differential Transmit Pair 7
PCIEUP_8_TXN/P	O	PCI Express Uplink Differential Transmit Pair 8
PCIEUP_9_TXN/P	O	PCI Express Uplink Differential Transmit Pair 9
PCIEUP_10_TXN/P	O	PCI Express Uplink Differential Transmit Pair 10
PCIEUP_11_TXN/P	O	PCI Express Uplink Differential Transmit Pair 11
PCIEUP_12_TXN/P	O	PCI Express Uplink Differential Transmit Pair 12
PCIEUP_13_TXN/P	O	PCI Express Uplink Differential Transmit Pair 13
PCIEUP_14_TXN/P	O	PCI Express Uplink Differential Transmit Pair 14
PCIEUP_15_TXN/P	O	PCI Express Uplink Differential Transmit Pair 15
PCIEUP_0_RXN/P	I	PCI Express Uplink Differential Receive Pair 0
PCIEUP_1_RXN/P	I	PCI Express Uplink Differential Receive Pair 1
PCIEUP_2_RXN/P	I	PCI Express Uplink Differential Receive Pair 2
PCIEUP_3_RXN/P	I	PCI Express Uplink Differential Receive Pair 3
PCIEUP_4_RXN/P	I	PCI Express Uplink Differential Receive Pair 4
PCIEUP_5_RXN/P	I	PCI Express Uplink Differential Receive Pair 5
PCIEUP_6_RXN/P	I	PCI Express Uplink Differential Receive Pair 6
PCIEUP_7_RXN/P	I	PCI Express Uplink Differential Receive Pair 7
PCIEUP_8_RXN/P	I	PCI Express Uplink Differential Receive Pair 8
PCIEUP_9_RXN/P	I	PCI Express Uplink Differential Receive Pair 9
PCIEUP_10_RXN/P	I	PCI Express Uplink Differential Receive Pair 10
PCIEUP_11_RXN/P	I	PCI Express Uplink Differential Receive Pair 11
PCIEUP_12_RXN/P	I	PCI Express Uplink Differential Receive Pair 12
PCIEUP_13_RXN/P	I	PCI Express Uplink Differential Receive Pair 13
PCIEUP_14_RXN/P	I	PCI Express Uplink Differential Receive Pair 14
PCIEUP_15_RXN/P	I	PCI Express Uplink Differential Receive Pair 15
PCIEUP_RCOMPP **PCIEUP_RCOMPN**	I	Impedance Compensation Inputs

Table 22-6. NPx8

Name	Type	Description
PCIE12_UP0_SATA0_TXN/P	O	PCI Express Uplink Differential Transmit Pair 0 PCI Express Uplink Transmit pair 0, muxed with PCI Express Root Port 12 and SATA port 0
PCIE13_UP1_SATA1_TXN/P	O	PCI Express Uplink Differential Transmit Pair 1 PCI Express Uplink Transmit pair 1, muxed with PCI Express Root Port 13 and SATA port 1
PCIE14_UP2_SATA2_TXN/P	O	PCI Express Uplink Differential Transmit Pair 2 PCI Express Uplink Transmit pair 2, muxed with PCI Express Root Port 14 and SATA port 2
PCIE15_UP3_SATA3_TXN/P	O	PCI Express Uplink Differential Transmit Pair 3 PCI Express Uplink Transmit pair 3, muxed with PCI Express Root Port 15 and SATA port 3
PCIE16_UP4_SATA4_TXN/P	O	PCI Express Uplink Differential Transmit Pair 4 PCI Express Uplink Transmit pair 4, muxed with PCI Express Root Port 16 and SATA port 4
PCIE17_UP5_SATA5_TXN/P	O	PCI Express Uplink Differential Transmit Pair 5 PCI Express Uplink Transmit pair 5, muxed with PCI Express Root Port 17 and SATA port 5
PCIE18_UP6_SATA6_TXN/P	O	PCI Express Uplink Differential Transmit Pair 6 PCI Express Uplink Transmit pair 6, muxed with PCI Express Root Port 18 and SATA port 6
PCIE19_UP7_SATA7_TXN/P	O	PCI Express Uplink Differential Transmit Pair 7 PCI Express Uplink Transmit pair 7, muxed with PCI Express Root Port 19 and SATA port 7
PCIE12_UP0_SATA0_RXN/P	I	PCI Express Uplink Differential Receive Pair 0 PCI Express Uplink Receive pair 0, muxed with PCI Express Root Port 12 and SATA port 0
PCIE13_UP1_SATA1_RXN/P	I	PCI Express Uplink Differential Receive Pair 1 PCI Express Uplink Receive pair 1, muxed with PCI Express Root Port 13 and SATA port 1
PCIE14_UP2_SATA2_RXN/P	I	PCI Express Uplink Differential Receive Pair 2 PCI Express Uplink Receive pair 2, muxed with PCI Express Root Port 14 and SATA port 2
PCIE15_UP3_SATA3_RXN/P	I	PCI Express Uplink Differential Receive Pair 3 PCI Express Uplink Receive pair 3, muxed with PCI Express Root Port 15 and SATA port 3
PCIE16_UP4_SATA4_RXN/P	I	PCI Express Uplink Differential Receive Pair 4 PCI Express Uplink Receive pair 4, muxed with PCI Express Root Port 16 and SATA port 4
PCIE17_UP5_SATA5_TXN/P	I	PCI Express Uplink Differential Receive Pair 5 PCI Express Uplink Receive pair 5, muxed with PCI Express Root Port 17 and SATA port 5
PCIEUP_RCOMPP PCIEUP_RCOMPN	I	Impedance Compensation Inputs

22.5.3 PCIe Upstream Ports - Configuration

The PCIe Upstream ports consist of two groups, a x16 port that will refer to as NPX16, and an x8 port, referred to as NPx8. The availability and configuration of these ports is based on a mixture of fusing and soft straps. NPx16 port is always enabled by default. Fusing will enabled this as x16, x8 or x1 PCIe uplink depending on SKU. This sets the maximum width, but with soft straps it's possible to either 1) make the uplink smaller or 2) completely disable the uplink. The impact of #2 is that there is no Intel QuickAssist Technology or 10/1GbE functionality.

The PCH has two PCIe Uplink Configurations, Configuration 1 and Configuration 2, associated with how the Intel QuickAssist Technology engines route to the NPx8 and NPx16. Two of the three engines are routed to the NPx16 port, while the 3rd engine can either be routed to NPx8 or NPx16. Selection of which uplink configuration is used is done via softstrap. Refer to Section 22.5.3.1, "PCIe Uplink Configuration 1" and Section 22.5.3.2, "PCIe Uplink Configuration 2" for additional details. Refer to Section 22.6.1.3, "Virtual Switch Ports Overview" for additional details on the related VSP.

It must be noted that for maximum bandwidth of data transfer, if routing up NPx8 is an option, then you must route the PCIe traffic from the Intel QuickAssist Technology engine up that uplink.

Note: Not all PCH SKUs support the secondary PCIe x8 Uplink. On those SKUs, the Uplink has been fused to Configuration 2.

Note: For SKUs that do not have Intel QuickAssist Technology, there is only the NPx16 uplink.

Table 22-7. PCI Express Link Configurations Supported

PCIe NBX16 Link Config	PCI Express Lanes															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1x16	P1															
1x8	P1								NA	NA	NA	NA	NA	NA	NA	NA
1x4	P1				NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
1x1	P1	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA

Notes:
1. The up link has only one controller, and there is only one Port. Unused lanes are not available.

PCIe* NBX8 Link Config	PCI Express* Lanes							
	0	1	2	3	4	5	6	7
1x8	P1							
1x4	P1				NA	NA	NA	NA
1x1	P1	NA	NA	NA	NA	NA	NA	NA

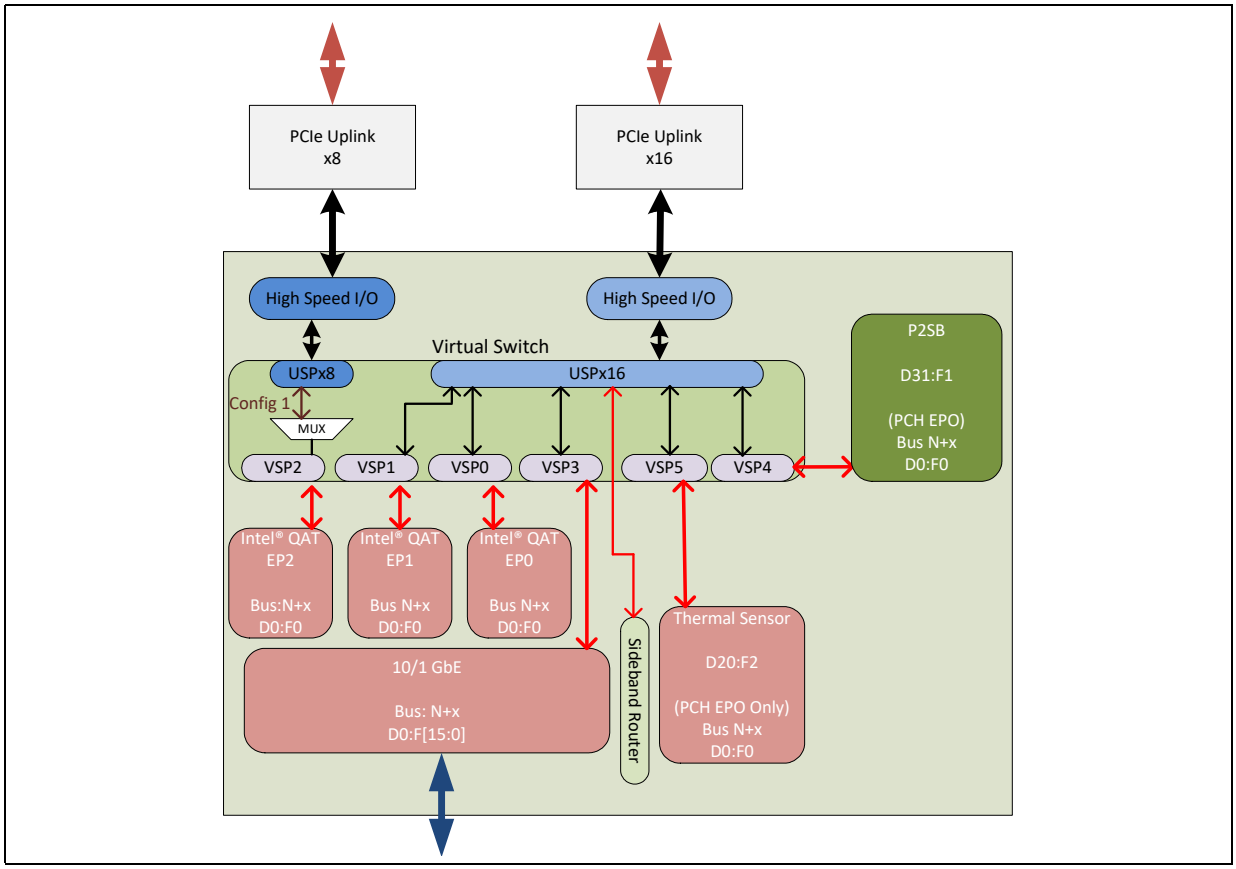
Notes:
1. The up link has only one controller, and there is only one Port. Unused lanes are not available.

22.5.3.1 PCIe Uplink Configuration 1

In PCIe Uplink Configuration 1, the primary PCIe x16 Uplink is used for transactions to the 10 GbE controller, Intel QAT Endpoints [1:0], P2SB, and the PCH thermal sensor (PCH EPO mode only) via the Virtual Switch Ports [5:3,1:0]. The secondary PCIe x8 Uplink is connected and dedicated to Intel QAT Endpoint 2 via Virtual Switch Port 2.

The following is a block diagram of PCIe Uplink configuration #1.

Figure 22-2. PCIe Uplink Configuration 1

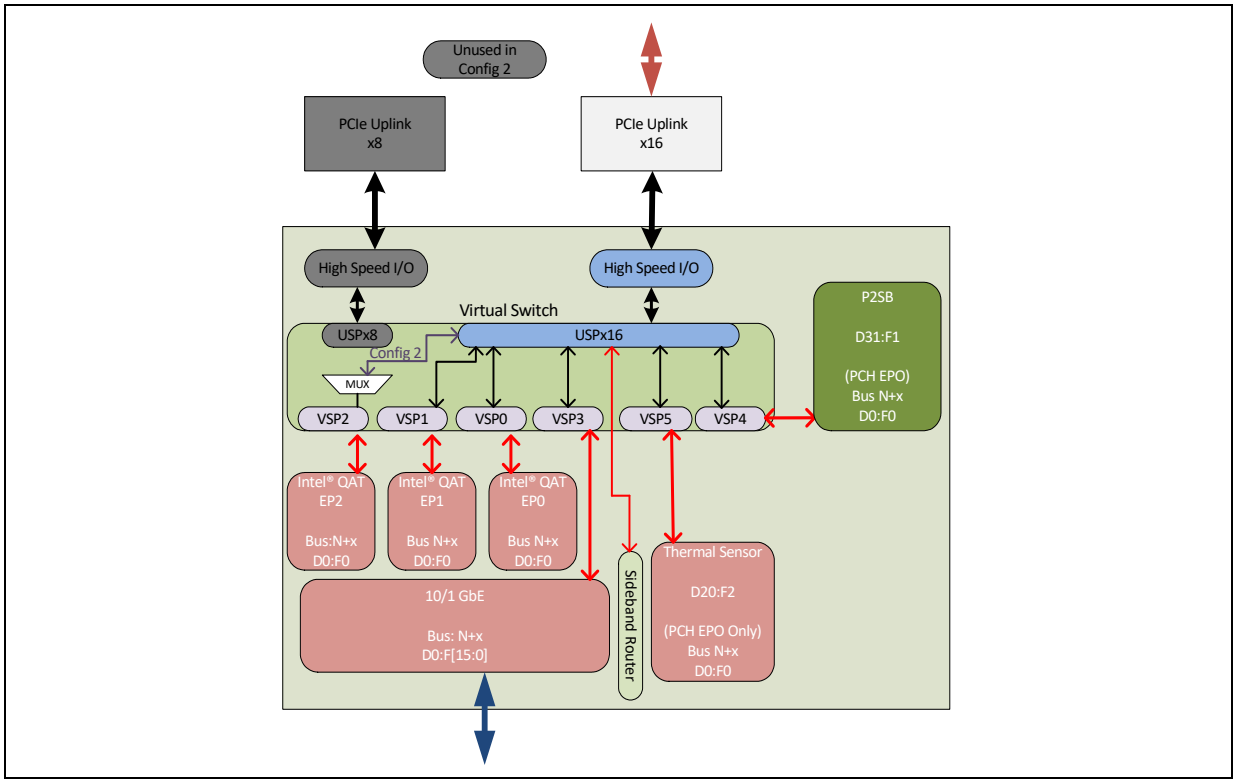


22.5.3.2 PCIe Uplink Configuration 2

In PCIe Uplink Configuration 2, the primary PCIe x16 Uplink is used for transactions to the 10 GbE controller, Intel QAT Endpoints [2:0], P2SB, and the PCH thermal sensor (PCH EPO Only) via the Virtual Switch Ports [5:0]. The secondary PCIe x8 Uplink is unused and is power gated.

The following is a block diagram of PCIe Uplink configuration #2.

Figure 22-3. NP8 PCIe Uplink Configuration 2



22.6 PCIe Virtual Switch Ports

22.6.1 Features

22.6.1.1 Supported Features

- PCI Express 3.0 specification Compliant
- VC0 support only
- 512B maximum payload
- 2048 maximum read request
- Malformed packet checks
- Allows passing of SR-IOV transactions through the PCIe Switch
- ARI support
- Advanced Error Reporting
- Role Based Error Reporting
- L1, L3
- D0, $D3_{HOT,}$ and $D3_{COLD}$
- Support of legacy INTx Interrupt messaging over PCIe
- Support for MSI, MSI-X interrupt messaging over PCIe

22.6.1.2 Features Not Supported

- 1x2 link width
- L0s
- Isochronous services
- P2P support
- Atomic ops
- Lock cycles
- Multicast

22.6.1.3 Virtual Switch Ports Overview

The PCH PCIe Upstream Ports and the Virtual Switch Ports together provide the capabilities of an integrated PCIe Switch. The VSP does not implement the data and link layer functionality relative to a typical PCIe downstream port.

The PCH supports independent VSPs with Type 1 headers for each PCIe endpoint device. Each of the endpoints have a separate VSP that allows the endpoint to show up as a Type 0 device.

The six VSPs are:

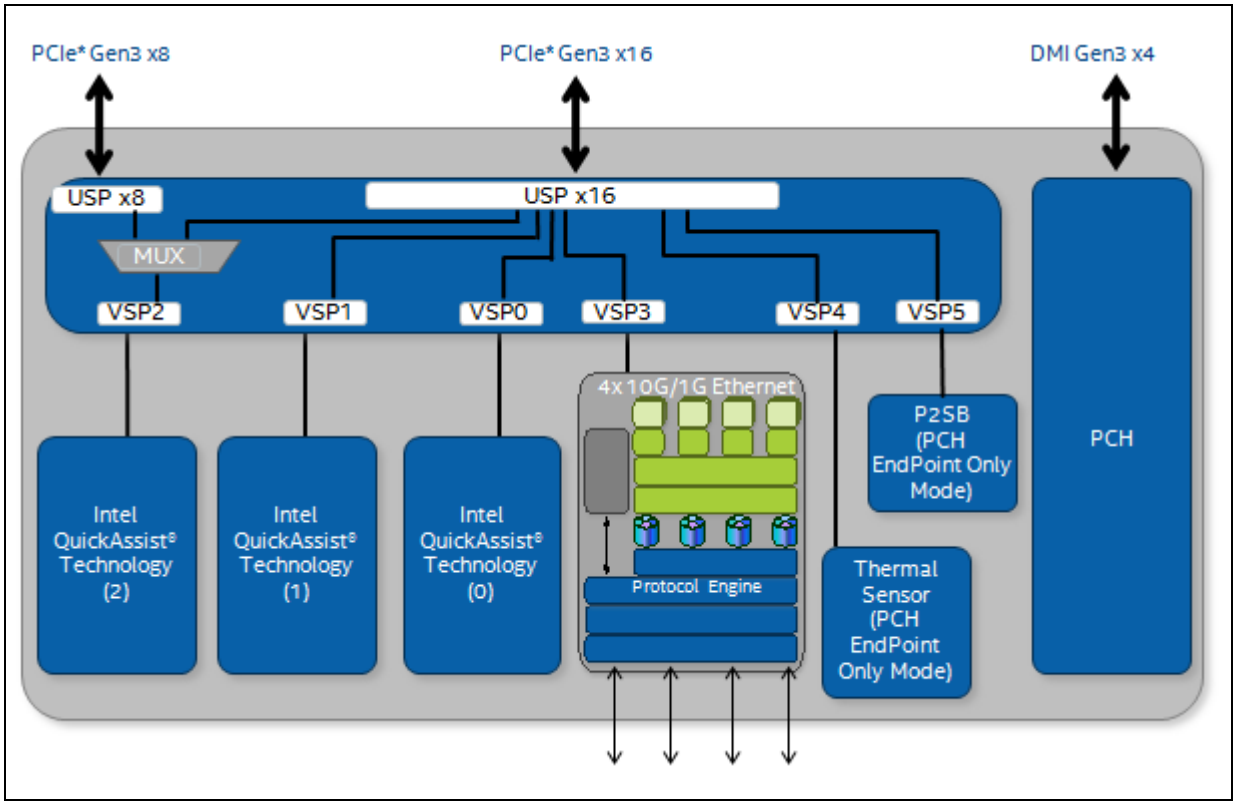
- VSP0 - associated with Intel QuickAssist Endpoint 0
- VSP1- associated with Intel QuickAssist Endpoint 1
- VSP2 - associated with Intel QuickAssist Endpoint 2
- VSP3 - associated with integrated 10/1 GbE
- VSP4 - associated with the PCH Thermal Sensor in Endpoint Only (EPO) mode

- VSP5 - associated with PCH P2SB in Endpoint Only (EPO) mode

Note: VSP[2:0] are only available on PCH SKUs that support Intel QuickAssist Technology.

Note: VSP[5:4] are only available when the PCH is configured for EndPoint Only (EPO) mode.

Figure 22-4. Virtual Switch Ports Block Diagram



22.7 PCI Express Root Port Configuration Registers Register Summary

There are twenty sets of the following configuration registers used for PCH PCI Express Port Configuration. Each PCH PCI Express* Configuration Register set covers a single PCI Express* Port and maps out as the following Device/Function: D28/F0 = Port1, D28/F1 = Port2, D28/F2 = Port3, D28/F3 = Port4 D28/F4 = Port5, D28/F5 = Port6, D28/F6 = Port7, D28/F7 = Port8 D29/F0 = Port9, D29/F1 = Port10, D29/F2 = Port11, D29/F3 = Port12 D29/F4 = Port13, D29/F5 = Port14, D29/F6 = Port15, D29/F7 = Port16 D27/F0 = Port17, D27/F1 = Port18, D27/F2 = Port19, D27/F3 = Port20

Table 22-8. Summary of PCI Express Port Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	Identifiers (ID)—Offset 0h	8086h
4h	7h	Device Command; Primary Status (CMD_PSTS)—Offset 4h	100000h
8h	Bh	Revision ID;Class Code (RID_CC)—Offset 8h	60400F0h
Ch	Fh	Cache Line Size; Primary Latency Timer; Header Type (CLS_PLT_HTYPE)—Offset Ch	810000h

Table 22-8. Summary of PCI Express Port Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
18h	1Bh	Bus Numbers; Secondary Latency Timer (BNUM_SLT)—Offset 18h	0h
1Ch	1Fh	I/O Base and Limit; Secondary Status (IOBL_SSTS)—Offset 1Ch	0h
20h	23h	Memory Base and Limit (MBL)—Offset 20h	0h
24h	27h	Prefetchable Memory Base and Limit (PMBL)—Offset 24h	10001h
28h	2Bh	Prefetchable Memory Base Upper 32 Bits (PMBU32)—Offset 28h	0h
2Ch	2Fh	Prefetchable Memory Limit Upper 32 Bits (PMLU32)—Offset 2Ch	0h
34h	37h	Capabilities List Pointer (CAPP)—Offset 34h	40h
3Ch	3Fh	Interrupt Information; Bridge Control (INTR_BCTRL)—Offset 3Ch	0h
40h	43h	Capabilities List; PCI Express Capabilities (CLIST_XCAP)—Offset 40h	428010h
44h	47h	Device Capabilities (DCAP)—Offset 44h	8001h
48h	4Bh	Device Control; Device Status (DCTL_DSTS)—Offset 48h	100000h
4Ch	4Fh	Link Capabilities (LCAP)—Offset 4Ch	710C00h
50h	53h	Link Control; Link Status (LCTL_LSTS)—Offset 50h	10000h
54h	57h	Slot Capabilities (SLCAP)—Offset 54h	40060h
58h	5Bh	Slot Control; Slot Status (SLCTL_SLSTS)—Offset 58h	0h
5Ch	5Fh	Root Control (RCTL)—Offset 5Ch	0h
60h	63h	Root Status (RSTS)—Offset 60h	0h
64h	67h	Device Capabilities 2 (DCAP2)—Offset 64h	80837h
68h	6Bh	Device Control 2; Device Status 2 (DCTL2_DSTS2)—Offset 68h	0h
6Ch	6Fh	Link Capabilities 2 (LCAP2)—Offset 6Ch	0h
70h	73h	Link Control 2; Link Status 2 (LCTL2_LSTS2)—Offset 70h	1h
80h	83h	Message Signaled Interrupt Identifiers; Message Signaled Interrupt Message Control (MID_MC)—Offset 80h	9005h
84h	87h	Message Signaled Interrupt Message Address (MA)—Offset 84h	0h
88h	8Bh	Message Signaled Interrupt Message Data (MD)—Offset 88h	0h
90h	93h	Subsystem Vendor Capability (SVCAP)—Offset 90h	A00Dh
94h	97h	Subsystem Vendor IDs (SVID)—Offset 94h	0h
A0h	A3h	Power Management Capability; PCI Power Management Capabilities (PMCAP_PMC)—Offset A0h	C8030001h
A4h	A7h	PCI Power Management Control and Status (PMCS)—Offset A4h	8h
100h	103h	Advanced Error Extended Reporting Capability Header (AECH)—Offset 100h	0h
104h	107h	Uncorrectable Error Status (UES)—Offset 104h	0h
108h	10Bh	Uncorrectable Error Mask (UEM)—Offset 108h	0h
10Ch	10Fh	Uncorrectable Error Severity (UEV)—Offset 10Ch	60011h
110h	113h	Correctable Error Status (CES)—Offset 110h	0h
114h	117h	Correctable Error Mask (CEM)—Offset 114h	2000h
118h	11Bh	Advanced Error Capabilities and Control (AECC)—Offset 118h	0h
12Ch	12Fh	Root Error Command (REC)—Offset 12Ch	0h
130h	133h	Root Error Status (RES)—Offset 130h	0h
134h	137h	Error Source Identification (ESID)—Offset 134h	0h
144h	147h	ACS Capability Register (ACSCAPR)—Offset 144h	Fh

Table 22-8. Summary of PCI Express Port Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
148h	14Bh	ACS Control Register (ACSCTLR)—Offset 148h	0h
200h	203h	L1 Sub-States Extended Capability Header (L1SECH)—Offset 200h	0h
204h	207h	L1 Sub-States Capabilities (L1SCAP)—Offset 204h	28281Fh
208h	20Bh	L1 Sub-States Control 1 (L1SCTL1)—Offset 208h	0h
20Ch	20Fh	L1 Sub-States Control 2 (L1SCTL2)—Offset 20Ch	28h
220h	223h	Secondary PCI Express Extended Capability Header (SPEECH)—Offset 220h	0h
224h	227h	Link Control 3 (LCTL3)—Offset 224h	0h
22Ch	22Fh	Lane 0 and Lane 1 Equalization Control (L01EC)—Offset 22Ch	7F7F7F7Fh
230h	233h	Lane 2 and Lane 3 Equalization Control (L23EC)—Offset 230h	7F7F7F7Fh

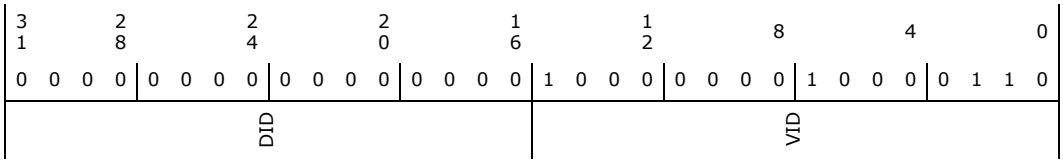
22.7.1 Identifiers (ID)—Offset 0h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 8086h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO/V	**Device Identification (DID):** The value of this ID is product specific.
15:0	8086h RO	**Vendor Identification (VID):** Indicates Intel

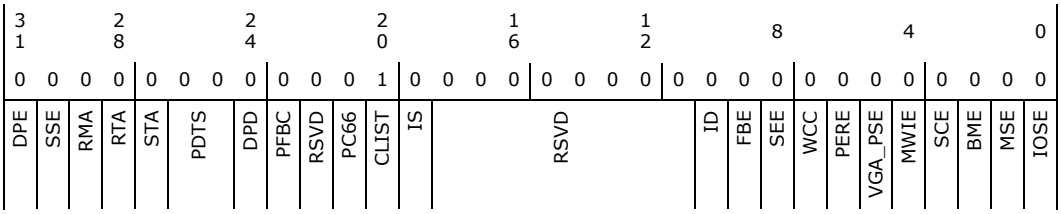
22.7.2 Device Command; Primary Status (CMD_PSTS)—Offset 4h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 100000h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RW/1C/V	**DPE - Detected Parity Error (DPE):** Set when the root port receives a command or data from the backbone with a parity error. This is set even if CMD.PERE is not set.
30	0h RW/1C/V	**Signaled System Error (SSE):** Set when the root port signals a system error to the internal SERR# logic.
29	0h RW/1C/V	**Received Master Abort (RMA):** Set when the root port receives a completion with unsupported request status from the backbone.
28	0h RW/1C/V	**Received Target Abort (RTA):** Set when the root port receives a completion with completer abort from the backbone.
27	0h RW/1C/V	**Signaled Target Abort (STA):** Set whenever the root port forwards a target abort received from the downstream device onto the backbone.
26:25	0h RO	**Primary DEVSEL# Timing Status (PDTS):** Reserved per *PCI Express Specification*.
24	0h RW/1C/V	**Master Data Parity Error Detected (DPD):** Set when the root port receives a completion with a data parity error on the backbone and CMD.PERE is set.
23	0h RO	**Primary Fast Back to Back Capable (PFBC):** Reserved per *PCI Express Specification*.
22	0h RO	Reserved.
21	0h RO	**Primary 66 MHz Capable (PC66):** Reserved per *PCI Express Specification*.
20	1h RO	**Capabilities List (CLIST):** Indicates the presence of a capabilities list.
19	0h RO/V	**Interrupt Status (IS):** Indicates status of hot plug and power management interrupts on the root port that result in INTx# message generation. This bit is not set if MSI is enabled. If MSI is not enabled, this bit is set regardless of the state of CMD.ID.
18:11	0h RO	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
10	0h RW/V2	**Interrupt Disable (ID):** This disables pin-based INTx# interrupts on enabled hot plug and power management events. This bit has no effect on MSI operation. When set, internal INTx# messages will not be generated. When cleared, internal INTx# messages are generated if there is an interrupt for hot plug or power management and MSI is not enabled. This bit does not affect interrupt forwarding from devices connected to the root port. Assert_INTx and Deassert_INTx messages will still be forwarded to the internal interrupt controllers if this bit is set. For PCI Bus Emulation Mode compatibility, if the PCIBEM register is set, this register is RO and returns a value of 0 when read, else it is RW with the functionality described above.
9	0h RO	**Fast Back to Back Enable (FBE):** Reserved per *PCI Express Specification*.
8	0h RW	**SERR# Enable (SEE):** When set, enables the root port to generate an SERR# message when PSTS.SSE is set.
7	0h RO	**Wait Cycle Control (WCC):** Reserved per *PCI Express Specification*.
6	0h RW	**Parity Error Response Enable (PERE):** Indicates that the device is capable of reporting parity errors as a master on the backbone.
5	0h RO	**VGA Palette Snoop (VGA_PSE):** Reserved per *PCI Express Specification*.
4	0h RO	**Memory Write and Invalidate Enable (MWIE):** Reserved per *PCI Express Specification*.
3	0h RO	**Special Cycle Enable (SCE):** Reserved per *PCI Express Specification* and *PCI Bridge Specification*.
2	0h RW	**Bus Master Enable (BME):** When set, allows the root port to forward Memory and I/O Read/Write cycles onto the backbone from a PCI-Express device. When this bit is 0b, Memory and I/O requests received at a Root Port must be handled as Unsupported Requests (UR). This bit does not affect forwarding of Completions in either the Upstream or Downstream direction. The forwarding of Requests other than Memory or I/O requests is not controlled by this bit.
1	0h RW	**Memory Space Enable (MSE):** When set, memory cycles within the range specified by the memory base and limit registers can be forwarded to the PCI-Express device. When cleared, these memory cycles are master aborted on the backbone.
0	0h RW	**I/O Space Enable (IOSE):** When set, I/O cycles within the range specified by the I/O base and limit registers can be forwarded to the PCI-Express device. When cleared, these cycles are master aborted on the backbone.

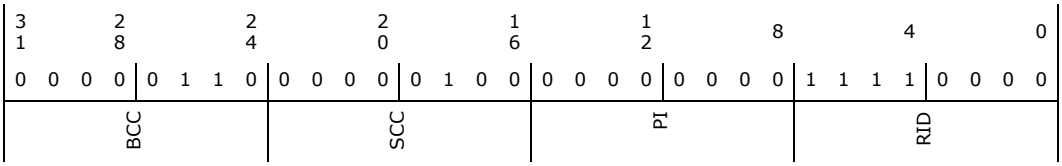
22.7.3 Revision ID;Class Code (RID_CC)—Offset 8h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 60400F0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	6h RO	**Base Class Code (BCC):** Indicates the device is a bridge device.
23:16	4h RO/V	**Sub-Class Code (SCC):** The default indicates the device is a PCI-to-PCI bridge. If the MPC.BT register is set to a '1' for a Host Bridge, this register reads 00h.
15:8	0h RO/V	**Programming Interface (PI):** The value reported in this register is a function of the Decode Control.Subtractive Decode Enable (SDE) register. SDE Value reported in this register 0: 00h 1: 01h
7:0	F0h RO/V	**Revision ID (RID):** Indicates the revision of the bridge.

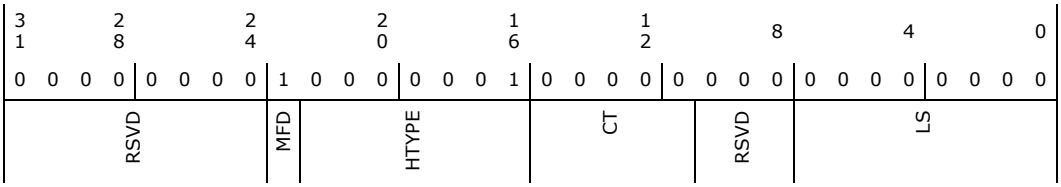
22.7.4 Cache Line Size; Primary Latency Timer; Header Type (CLS_PLT_HTYPE)—Offset Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 810000h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23	1h RO	**Multi-function Device (MFD):** This bit is '1' to indicate a multi-function device.
22:16	1h RO/V	**Header Type (HTYPE):** The default mode identifies the header layout of the configuration space, which is a PCI-to-PCI bridge. If the MPC.BT register is set to a '1' for a Host Bridge, this register reads 00h.
15:11	0h RO	**Latency Count (CT):** Reserved per *PCI Express Specification*.
10:8	0h RO	Reserved.
7:0	0h RW	**Line Size (LS):** This is read/write but contains no functionality, per *PCI Express Specification*.

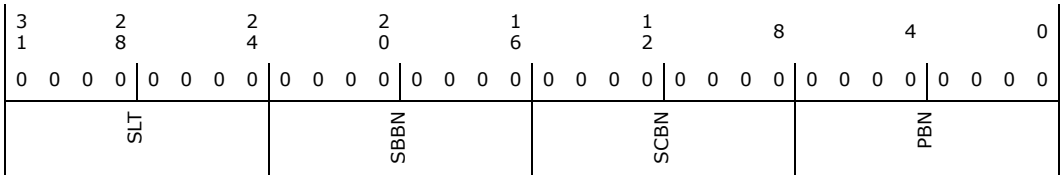
22.7.5 Bus Numbers; Secondary Latency Timer (BNUM_SLT)—Offset 18h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RW/V2	**Secondary Latency Timer (SLT):** For PCI Bus Emulation Mode compatibility, if the DC.PCIBEM register is set, this register is a RW register; else this register is RO and returns 0. This register does not affect the behavior of any HW logic.
23:16	0h RW	**Subordinate Bus Number (SBBN):** Indicates the highest PCI bus number below the bridge.
15:8	0h RW	**Secondary Bus Number (SCBN):** Indicates the bus number the port.
7:0	0h RW	**Primary Bus Number (PBN):** Indicates the bus number of the backbone.

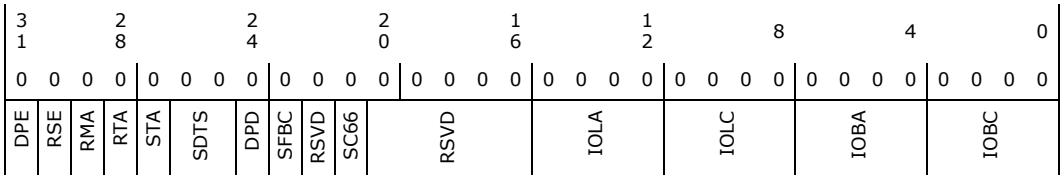
22.7.6 I/O Base and Limit; Secondary Status (IOBL_SSTS)—Offset 1Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RW/1C/V	**Detected Parity Error (DPE):** Set when the port receives a poisoned TLP.
30	0h RW/1C/V	**Received System Error (RSE):** Set when the port receives an ERR_FATAL or ERR_NONFATAL message from the device.
29	0h RW/1C/V	**Received Master Abort (RMA):** Set when the port receives a completion with 'Unsupported Request' status from the device.
28	0h RW/1C/V	**Received Target Abort (RTA):** Set when the port receives a completion with 'Completion Abort' status from the device.
27	0h RW/1C/V	**Signaled Target Abort (STA):** Set when the port generates a completion with 'Completion Abort' status to the device.
26:25	0h RO/V	**Secondary DEVSEL# Timing Status (SDTS):** Reserved per *PCI Express Specification*. For PCI Bus Emulation Mode compatibility, if the PCIBEM register is set, this register returns a value of 01b when read, else this register returns a value of 00b.
24	0h RW/1C/V	**Data Parity Error Detected (DPD):** Set when the BCTRL.PERE, and either of the following two conditions occurs: Port receives completion marked poisoned. Port poisons a write request to the secondary side.
23	0h RO/V	**Secondary Fast Back to Back Capable (SFBC):** Reserved per *PCI Express Specification*. For PCI Bus Emulation Mode compatibility, if the PCIBEM register is set, this register returns a value of 1b when read, else this register returns a value of 0b.
22	0h RO	Reserved.
21	0h RO	**Secondary 66 MHz Capable (SC66):** Reserved per *PCI Express Specification*.
20:16	0h RO	Reserved.
15:12	0h RW	**I/O Address Limit (IOLA):** I/O Base bits corresponding to address lines 15:12 for 4KB alignment. Bits 11:0 are assumed to be padded to FFFh.
11:8	0h RO	**I/O Limit Address Capability (IOLC):** Indicates that the bridge does not support 32-bit I/O addressing.
7:4	0h RW	**I/O Base Address (IOBA):** I/O Base bits corresponding to address lines 15:12 for 4KB alignment. Bits 11:0 are assumed to be padded to 000h.
3:0	0h RO	**I/O Base Address Capability (IOBC):** Indicates that the bridge does not support 32-bit I/O addressing.

22.7.7 Memory Base and Limit (MBL)—Offset 20h

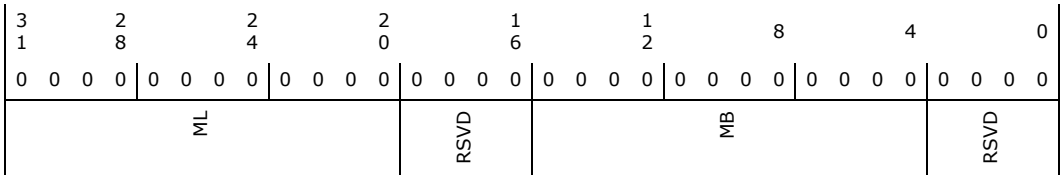
Accesses that are within the ranges specified in this register will be sent to the attached device if CMD.MSE is set. Accesses from the attached device that are outside the ranges specified will be forwarded to the backbone if CMD.BME is set. The comparison performed is MB [gt]= AD[lb]31:20[rb] [lt]= ML.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:20	0h RW	**Memory Limit (ML):** These bits are compared with bits [31:20] of the incoming address to determine the upper 1MB aligned value of the range.
19:16	0h RO	Reserved.
15:4	0h RW	**Memory Base (MB):** These bits are compared with bits [31:20] of the incoming address to determine the lower 1MB aligned value of the range.
3:0	0h RO	Reserved.

22.7.8 Prefetchable Memory Base and Limit (PMBL)—Offset 24h

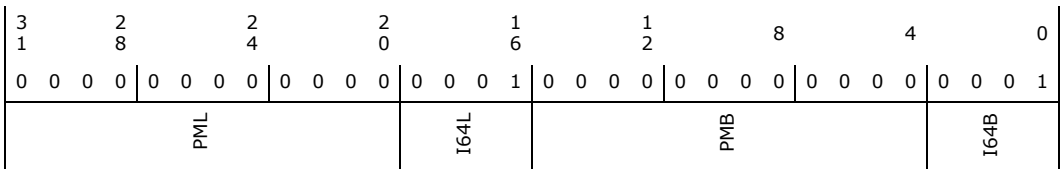
Accesses that are within the ranges specified in this register will be sent to the device if CMD.MSE is set. Accesses from the device that are outside the ranges specified will be forwarded to the backbone if CMD.BME is set. The comparison performed is PMBU32:PMB [gt]= AD[lb]63:32[rb]:AD[lb]31:20[rb] [lt]= PMLU32:PML.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 10001h



Bit Range	Default and Access	Field Name (ID): Description
31:20	0h RW	**Prefetchable Memory Limit (PML):** These bits are compared with bits [31:20] of the incoming address to determine the upper 1MB aligned value of the range.
19:16	1h RO	**64-bit Indicator (I64L):** Indicates support for 64-bit addressing.
15:4	0h RW	**Prefetchable Memory Base (PMB):** These bits are compared with bits [31:20] of the incoming address to determine the lower 1MB aligned value of the range.
3:0	1h RO	**64-bit Indicator (I64B):** Indicates support for 64-bit addressing.

22.7.9 Prefetchable Memory Base Upper 32 Bits (PMBU32)—Offset 28h

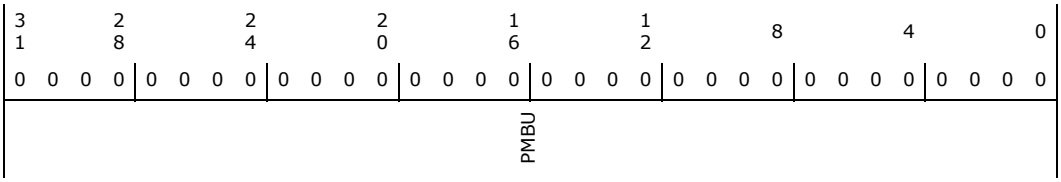
Size:32 bits

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Prefetchable Memory Base Upper Portion (PMBU):** Upper 32-bits of the prefetchable address base.

22.7.10 Prefetchable Memory Limit Upper 32 Bits (PMLU32)—Offset 2Ch

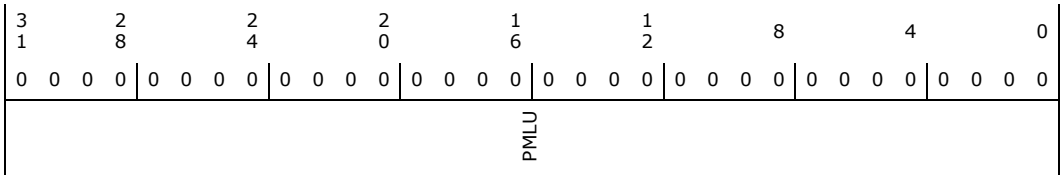
Size:32 bits

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Prefetchable Memory Limit Upper Portion (PMLU):** Upper 32-bits of the prefetchable address limit.

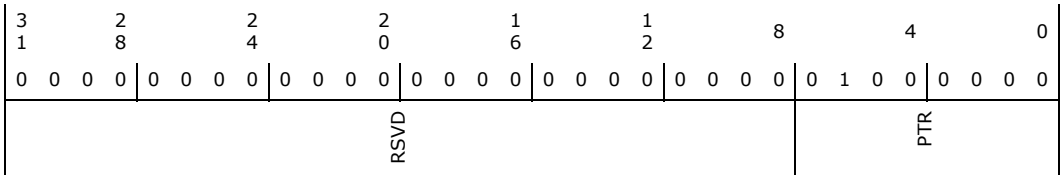
22.7.11 Capabilities List Pointer (CAPP)—Offset 34h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 40h



Bit Range	Default and Access	Field Name (ID): Description
31:8	0h RO	Reserved.
7:0	40h RW/O	**Capabilities Pointer (PTR):** Indicates that the pointer for the first entry in the capabilities list. BIOS can determine which capabilities will be exposed by including or removing them from the capability linked list. As this register is RWO, BIOS must write a value to this register, even if it is to re-write the default value. Capability Linked List (Default Settings) OffsetCapability Next Pointer 40h PCI Express 80h 80h Message Signaled Interrupt (MSI) 90h 90h Subsystem Vendor A0h A0h PCI Power Management 00h Extended PCIe Capability Linked List OffsetCapability Next Pointer 100h Advanced Error Reporting 000h

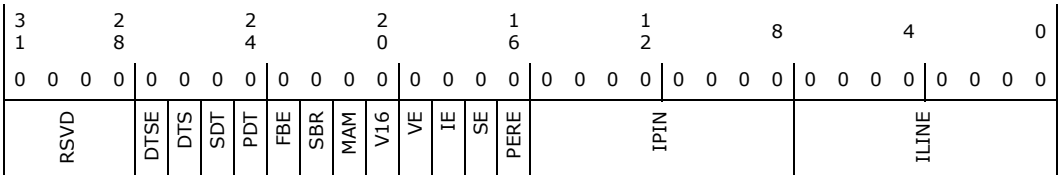
22.7.12 Interrupt Information; Bridge Control (INTR_BCTRL)—Offset 3Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:28	0h RO	Reserved.
27	0h RW/V2	**Discard Timer SERR# Enable (DTSE):** Reserved per *PCI Express Specification*. For PCI Bus Emulation Mode compatibility, if the DC.PCIBEM register is set, this register is RW else it is RO. This register is only maintained for SW compatibility and has no functionality within the port.
26	0h RO	**Discard Timer Status (DTS):** Reserved per *PCI Express Specification*. For PCI Bus Emulation Mode compatibility, this register can remain RO as no secondary discard timer exists that will ever cause it to be set.
25	0h RW/V2	**Secondary Discard Timer (SDT):** Reserved per *PCI Express Specification*. For PCI Bus Emulation Mode compatibility, if the DC.PCIBEM register is set, this register is RW else it is RO. This register is only maintained for SW compatibility and has no functionality within the port.
24	0h RW/V2	**Primary Discard Timer (PDT):** Reserved per *PCI Express Specification*. For PCI Bus Emulation Mode compatibility, if the DC.PCIBEM register is set, this register is RW else it is RO. This register is only maintained for SW compatibility and has no functionality within the port.
23	0h RO	**Fast Back to Back Enable (FBE):** Reserved per *PCI Express Specification*.

Bit Range	Default and Access	Field Name (ID): Description
22	0h RW	**Secondary Bus Reset (SBR):** Triggers a hot reset on the PCI-Express port.
21	0h RW/V2	**Master Abort Mode (MAM):** Reserved per *PCI Express Specification*.For PCI Bus Emulation Mode compatibility, if the DC.PCIBEM register is set, this register is RW else it is RO. This register is only maintained for SW compatibility and has no functionality within the port.
20	0h RW	**VGA 16-Bit Decode (V16):** When set, indicates that the I/O aliases of the VGA range (see BCTRL:VE definition below), are not enabled. 0: Execute 10-bit address decode on VGA I/O accesses. 1: Execute 16-bit address decode on VGA I/O accesses.
19	0h RW	**VGA Enable (VE):** When set, the following ranges will be claimed off the backbone by the root port: Memory ranges A0000h-BFFFFh I/O ranges 3B0h 3BBh and 3C0h 3DFh, and all aliases of bits 15:10 in any combination of 1's
18	0h RW	**ISA Enable (IE):** This bit only applies to I/O addresses that are enabled by the I/O Base and I/O Limit registers and are in the first 64KB of PCI I/O space. If this bit is set, the root port will block any forwarding from the backbone to the device of I/O transactions addressing the last 768 bytes in each 1KB block (offsets 100h to 3FFh).
17	0h RW	**SERR# Enable (SE):** When set, ERR_COR, ERR_NONFATAL, and ERR_FATAL messages received are forwarded to the backbone. When cleared, they are not.
16	0h RW	**Parity Error Response Enable (PERE):** When set, poisoned write TLPs and completions indicating poisoned TLPs will set the SSTS.DPD.
15:8	0h RO/V	**Interrupt Pin (IPIN):** Indicates the interrupt pin driven by the root port. At reset, this register takes on the following values, which reflect the reset state of the STRPFUSECFG register in chipset config space: Port Bits[lb]15:12[rb] Bits[lb]11:08[rb] 1 0h STRPFUSECFG.P1IP 2 0h STRPFUSECFG.P2IP 3 0h STRPFUSECFG.P3IP 4 0h STRPFUSECFG.P4IP 5 0h STRPFUSECFG.P5IP 6 0h STRPFUSECFG.P6IP 7 0h STRPFUSECFG.P7IP 8 0h STRPFUSECFG.P8IP The value that is programmed into STRPFUSECFG.PxIP is always reflected in this register. For PCI Bus Emulation Mode compatibility, if the DC.PCIBEM register is set, this register returns a value of 00h when read, else this register returns the value from the table above.
7:0	0h RW	**Interrupt Line (ILINE):** Software written value to indicate which interrupt line (vector) the interrupt is connected to. No hardware action is taken on this register.

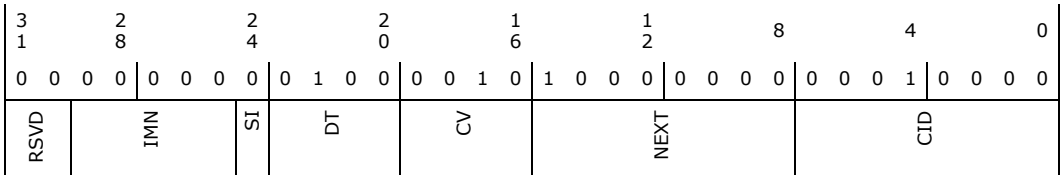
22.7.13 Capabilities List; PCI Express Capabilities (CLIST_XCAP)—Offset 40h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 428010h



Bit Range	Default and Access	Field Name (ID): Description
31:30	0h RO	Reserved.
29:25	0h RO	**Interrupt Message Number (IMN):** The root port does not have multiple MSI interrupt numbers.
24	0h RW/O	**Slot Implemented (SI):** Indicates whether the root port is connected to a slot. Slot support is platform specific. BIOS programs this field, and it is maintained until a platform reset.
23:20	4h RO	**Device / Port Type (DT):** Indicates this is a PCI-Express root port
19:16	2h RO	**Capability Version (CV):** Version 2.0 indicates devices compliant to the PCI Express 2.0 specification which incorporates the Register Expansion ECN.
15:8	80h RW/O	**Next Capability (NEXT):** Indicates the location of the next capability. The default value of this register is 80h which points to the MSI Capability structure. BIOS can determine which capabilities will be exposed by including or removing them from the capability linked list. As this register is RWO, BIOS must write a value to this register, even if it is to re-write the default value.
7:0	10h RO	**Capability ID (CID):** Indicates this is a PCI Express capability

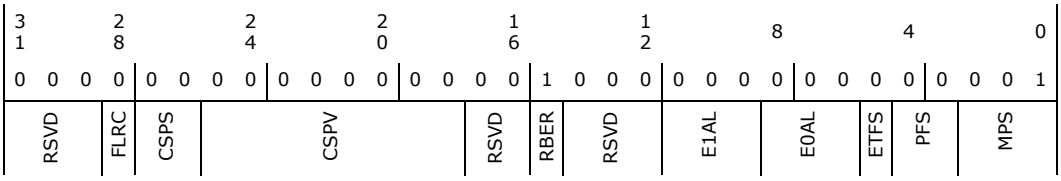
22.7.14 Device Capabilities (DCAP)—Offset 44h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 8001h



Bit Range	Default and Access	Field Name (ID): Description
31:29	0h RO	Reserved.
28	0h RO	**Function Level Reset Capable (FLRC):** Not supported in Root Ports
27:26	0h RO	**Captured Slot Power Limit Scale (CSPS):** Not supported
25:18	0h RO	**Captured Slot Power Limit Value (CSPV):** Not supported
17:16	0h RO	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
15	1h RO	**Role Based Error Reporting (RBER):** When Set, this bit indicates that the Function implements the functionality originally defined in the Error Reporting ECN for *PCI Express Base Specification*, Revision 1.0a, and later incorporated into PCI Express Base Specification, Revision 1.1. This bit must be Set by all Functions conforming to the ECN, *PCI Express Base Specification*, Revision 1.1, or subsequent PCI Express Base Specification revisions.
14:12	0h RO	Reserved.
11:9	0h RO	**Endpoint L1 Acceptable Latency (E1AL):** Reserved for root ports.
8:6	0h RO	**Endpoint L0 Acceptable Latency (E0AL):** Reserved for Root port.
5	0h RO	**Extended Tag Field Supported (ETFS):** The root port never needs to initiate a transaction as a Requester with the Extended Tag bits being set. This bit does not affect the root port's ability to forward requests as a bridge as the root port always supports forwarding requests with extended tags.
4:3	0h RO	**Phantom Functions Supported (PFS):** No phantom functions supported
2:0	1h RW/O	**Max Payload Size Supported (MPS):** BIOS should write to this field during system initialization. Only Max Payload Size of up to 256B is supported. Programming this field to any values other than 128B max payload size will result in aliasing to 128B max payload size. 000b: 128 bytes max payload size. 001b: 256 bytes max payload size. 010b: 512 bytes max payload size. 011b: 1024 bytes max payload size. 100b: 2048 bytes max payload size. 101b: 4096 bytes max payload size. 110b: Reserved. 111b: Reserved. This field applies only to the PCIe link interface.

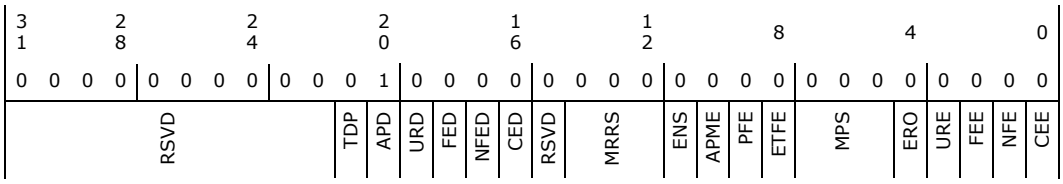
22.7.15 Device Control; Device Status (DCTL_DSTS)—Offset 48h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 100000h



Bit Range	Default and Access	Field Name (ID): Description
31:22	0h RO	Reserved.
21	0h RO	**Transactions Pending (TDP):** This bit has no meaning for the root port since it never initiates a non-posted request with its own Requester ID.
20	1h RO	**AUX Power Detected (APD):** The root port contains AUX power for wakeup

Bit Range	Default and Access	Field Name (ID): Description
19	0h RW/1C/V	**Unsupported Request Detected (URD):** Indicates an unsupported request was detected.
18	0h RW/1C/V	**Fatal Error Detected (FED):** Indicates a fatal error was detected. Set when a fatal error occurred on from a data link protocol error, buffer overflow, or malformed tlp
17	0h RW/1C/V	**Non-Fatal Error Detected (NFED):** Indicates a non-fatal error was detected. Set when an received a non-fatal error occurred from a poisoned tlp, unexpected completions, unsupported requests, completer abort, or completer timeout
16	0h RW/1C/V	**Correctable Error Detected (CED):** Indicates a correctable error was detected. Set when received an internal correctable error from receiver errors / framing errors, tlp crc error, dllp crc error, replay num rollover, replay timeout.
15	0h RO	Reserved.
14:12	0h RO	**Max Read Request Size (MRRS):** Hardwired to 0. This field applies only to the PCIe link interface.
11	0h RO	**Enable No Snoop (ENS):** Not supported. The root port will never issue non-snoop requests.
10	0h RW/P	**Aux Power PM Enable (APME):** The OS will set this bit to '1' if the device connected has detected aux power.
9	0h RO	**Phantom Functions Enable (PFE):** Not supported
8	0h RO	**Extended Tag Field Enable (ETFE):** Not supported
7:5	0h RW	**Max Payload Size (MPS):** The root port only supports up to 256B max payload. Programming this field to any values other than 128B or 256B max payload size will result in aliasing to 128B max payload size. If the DCAP.MPS indicates 128B max payload size support, programming this field to any values other than 128B will result in aliasing to 128B max payload size. Programming this field to any values greater than DCAP.MPS will result in aliasing to 128B max payload size. 000b: 128 bytes max payload size. 001b: 256 bytes max payload size. 010b: 512 bytes max payload size. 011b: 1024 bytes max payload size. 100b: 2048 bytes max payload size. 101b: 4096 bytes max payload size. 110b: Reserved. 111b: Reserved. This field applies only to the PCIe link interface. ***Note:*** Software should ensure that the system is quiescent and no TLP is in progress prior to changing this field. BIOS should program this field prior to enabling BME.
4	0h RO	**Enable Relaxed Ordering (ERO):** Not supported
3	0h RW	**Unsupported Request Reporting Enable (URE):** When set, allows signaling ERR_NONFATAL, ERR_FATAL, or ERR_COR to the Root Control register when detecting an unmasked Unsupported Request (UR). An ERR_COR is signaled when a unmasked Advisory Non-Fatal UR is received. An ERR_FATAL, ERR_or NONFATAL, is sent to the Root Control Register when an uncorrectable non-Advisory UR is received with the severity set by the Uncorrectable Error Severity register.
2	0h RW	**Fatal Error Reporting Enable (FEE):** enables signaling of ERR_FATAL to the Root Control register due to internally detected errors or error messages received across the link. Other bits also control the full scope of related error reporting.
1	0h RW	**Non-Fatal Error Reporting Enable (NFE):** When set, enables signaling of ERR_NONFATAL to the Root Control register due to internally detected errors or error messages received across the link. Other bits also control the full scope of related error reporting.
0	0h RW	**Correctable Error Reporting Enable (CEE):** When set, enables signaling of ERR_CORR to the Root Control register due to internally detected errors or error messages received across the link. Other bits also control the full scope of related error reporting.

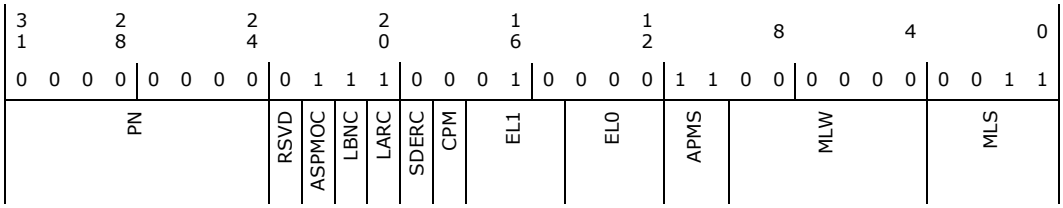
22.7.16 Link Capabilities (LCAP)—Offset 4Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 710C00h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO/V	**Port Number (PN):** Indicates the port number for the root port. This value is different for each implemented port: Port # Value of PN field 1 01h 2 02h 3 03h 4 04h
23	0h RO	Reserved.
22	1h RW/O	**ASPM Optionality Compliance (ASPMOC):** ASPM Optionality Compliance(ASPMOC): This bit must be set to 1b for PCIe 3.0 compliant port. Components implemented against certain earlier versions of this specification will have this bit set to 0b. Software is permitted to use the value of this bit to help determine whether to enable ASPM or whether to run ASPM compliance tests.
21	1h RO	**Link Bandwidth Notification Capability (LBNC):** This port supports Link Bandwidth Notification status and interrupt mechanisms.
20	1h RO	**Link Active Reporting Capable (LARC):** This port supports the optional capability of reporting the DL_Active state of the Data Link Control and Management State Machine.
19	0h RO	**Surprise Down Error Reporting Capable (SDERC):** Set to '0' to indicate the root port does not support Surprise Down Error Reporting
18	0h RO	**Clock Power Management (CPM):** '0' Indicates that root ports do not support the CLKREQ# mechanism.
17:15	2h RW/O	**L1 Exit Latency (EL1):** Indicates an exit latency of 2us to 4us. 000b Less than 1 us 001b 1 us to less than 2 us 010b 2 us to less than 4 us 011b 4 us to less than 8 us 100b 8 us to less than 16 us 101b 16 us to less than 32 us 110b 32 us to 64 us 111b More than 64 us ***Note:*** If PXP PLL shutdown is enabled, BIOS should program this latency to comprehend PLL lock latency.
14:12	0h RO/V	**L0s Exit Latency (EL0):** Indicates an exit latency based upon common-clock configuration: LCTL.CCC Value 0 MPC.UCEL 1 MPC.CCEL

Bit Range	Default and Access	Field Name (ID): Description
11:10	3h RW/O	**Active State Link PM Support (APMS):** Indicates the level of active state power management on this link Bits Definition 00 No ASPM Supported 01 L0s Supported 10 L1 Supported 11 L0s and L1 supported ***Note:*** If STRPFUSECFG.ASPMDIS is 1, the default of this field is '01'. Otherwise, the default of this field is '11'. If STRPFUSECFG.ASPMDIS is 1, BIOS writing '11' to this field will have the same effect as writing '01'. '01' will be reflected on this register when read and the register will turn to Read-Only once written once.
9:4	0h RO/V	**Maximum Link Width (MLW):** For the root ports, several values can be taken, based upon the value of the chipset configuration register field RPC.PC1 for ports 1-4: Port # Value of PN field RPC.PC1 00 01 10 11 1 01h 02h 02h 04h 2 01h 01h 01h 01h 3 01h 01h 02h 01h 4 01h 01h 01h 01h
3:0	3h RO/V	**Max Link Speeds (MLS):** Indicates the supported link speeds of the Uplink Port. This field indicates the supported link speed(s) of the associated port. 0001b: 2.5 Gb/s link speed is supported 0010b: 5.0 Gb/s and 2.5 Gb/s link speed supported 0011b: 8.0 Gb/s and 5.0 and 2.5 Gb/s link speed supported Others: Reserved. ***Note:*** The encoding is the binary value of the bit location in the Supported Link Speeds Vector (in the Link Capabilities 2 register) that corresponds to the maximum Link speed.

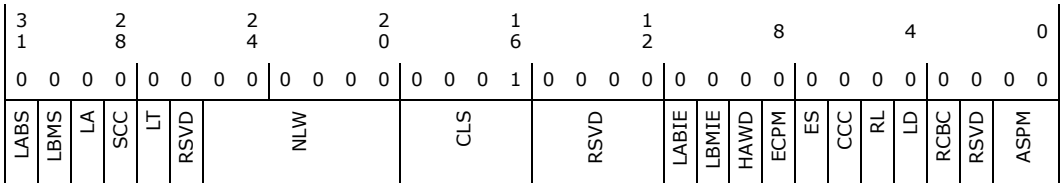
22.7.17 Link Control; Link Status (LCTL_LSTS)—Offset 50h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 10000h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RW/1C/V	**Link Autonomous Bandwidth Status (LABS):** This bit is Set by hardware to indicate that hardware has autonomously changed Link speed or width, without the Port transitioning through DL_Down status, for reasons other than to attempt to correct unreliable Link operation. This bit must be set if the Physical Layer reports a speed or width change was initiated by the Downstream component that was indicated as an autonomous change. The default value of this bit is 0b.
30	0h RW/1C/V	**Link Bandwidth Management Status (LBMS):** This bit is Set by hardware to indicate that either of the following has occurred without the Port transitioning through DL_Down status: - A Link retraining has completed following a write of 1b to the Retrain Link bit. ***Notes:*** This bit is Set following any write of 1b to the Retrain Link bit, including when the Link is in the process of retraining for some other reason. - Hardware has changed Link speed or width to attempt to correct unreliable Link operation, either through an LTSSM timeout or a higher level process This bit must be set if the Physical Layer reports a speed or width change was initiated by the Downstream component that was not indicated as an autonomous change. The default value of this bit is 0b.
29	0h RO/V	**Link Active (LA):** Set to 1b when the Data Link Control and Management State Machine is in the DL_Active state, 0b otherwise.
28	0h RO/V	**Slot Clock Configuration (SCC):** In normal mode, root port uses the same reference clock as on the platform and does not generate its own clock. ***Note:*** 1The default of this register bit is dependent on the 'PCIe Non-Common Clock With SSC Mode Enable Strap'. If the strap enables non-common clock with SSC support, this bit shall default to '0'. Otherwise, this bit shall default to '1'.
27	0h RO/V	**Link Training (LT):** The root port sets this bit whenever link training is occurring, or that 1b was written to the Retrain Link bit but Link training has not yet begun. It clears the bit upon completion of link training.
26	0h RO	Reserved.
25:20	0h RO/V	**Negotiated Link Width (NLW):** For the root ports, this register could take on several values: Port # Value of PN field RPC.PC1 00 01 10 11 1 01h 02h 02h 04h 2 01h 01h 01h 01h 3 01h 01h 02h 01h 4 01h 01h 01h 01h The value of this register is undefined if the link has not successfully trained.

Bit Range	Default and Access	Field Name (ID): Description
19:16	1h RO/V	**Current Link Speed (CLS):** 0001b Link is 2.5Gb/s Link 0010b 5.0 GT/s Link This field indicates the negotiated Link speed of the given link. The encoded value specifies a bit location in the Supported Link Speeds Vector (in the Link Capabilities 2 register) that corresponds to the current Link speed. Defined encodings are: 0001b: Supported Link Speeds Vector field bit 0. 0010b: Supported Link Speeds Vector field bit 1. 0011b: Supported Link Speeds Vector field bit 2. 0100b: Supported Link Speeds Vector field bit 3. 0101b: Supported Link Speeds Vector field bit 4. 0110b: Supported Link Speeds Vector field bit 5. 0111b: Supported Link Speeds Vector field bit 6. All other encodings are reserved. The value of this field is undefined if the link is not up.
15:12	0h RO	Reserved.
11	0h RW	**Link Autonomous Bandwidth Interrupt Enable (LABIE):** Link Autonomous Bandwidth Interrupt Enable - When Set, this bit enables the generation of an interrupt to indicate that the Link Autonomous Bandwidth Status bit has been Set.
10	0h RW	**Link Bandwidth Management Interrupt Enable (LBMIE):** When Set, this bit enables the generation of an interrupt to indicate that the Link Bandwidth Management Status bit has been Set. This bit is not applicable and is reserved for Endpoints, PCI Express-to-PCI/PCI-X bridges, and Upstream Ports of Switches. Functions that do not implement the Link Bandwidth Notification Capability must hardwire this bit to 0b. Default value of this bit is 0b.
9	0h RW	**Hardware Autonomous Width Disable (HAWD):** When Set, this bit disables hardware from changing the Link width for reasons other than attempting to correct unreliable Link operation by reducing Link width. Default value of this bit is 0b. ***Note:*** When operating as PCI Express, this bit defines the value of the Link Upconfigure Capability in TS2 Ordered Sets.
8	0h RO	**Enable Clock Power Management (ECPM):** Reserved. Not supported on Root Ports.
7	0h RW	**Extended Synch (ES):** When set, forces extended transmission of FTS ordered sets in FTS and extra TS2 at exit from L1 prior to entering L0. ***Note:*** This functionality is not applicable for Mobile Express.
6	0h RW	**Common Clock Configuration (CCC):** When set, indicates that the root port and device are operating with a distributed common reference clock.
5	0h WO	**Retrain Link (RL):** When set, the root port will train its downstream link. This bit always returns '0' when read. Software uses LSTS.LT to check the status of training. It is permitted to write 1b to this bit while simultaneously writing modified values to other fields in this register. If the LTSSM is not already in Recovery or Configuration, the resulting Link training must use the modified values. If the LTSSM is already in Recovery or Configuration, the modified values are not required to affect the Link training that's already in progress.
4	0h RW	**Link Disable (LD):** When set, the root port will disable the link by directing the LTSSM to the Disabled state.
3	0h RW/O	**Read Completion Boundary Control (RCBC):** Indicates the read completion boundary is 64 bytes.
2	0h RO	Reserved.
1:0	0h RW	**Active State Link PM Control (ASPM):** Indicates whether the root port should enter L0s or L1 or both. Bits Definition 00 Disabled 01 L0s Entry Enabled 10 L1 Entry Enabled 11 L0s and L1 Entry Enabled The value of this register is used unless the Root Port ASPM Control Override Enable register is set, in which case the Root Port ASPM Control Override value is used. ***Note:*** If STRPFUSECFG.ASPMDIS is '1', hardware will always see '00' as an output from this register. BIOS reading this register should always return the correct value.

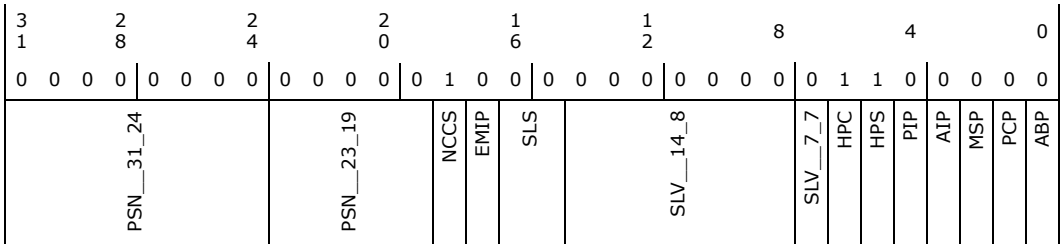
22.7.18 Slot Capabilities (SLCAP)—Offset 54h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 40060h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RW/O	**Physical Slot Number (PSN__31_24):** This is a value that is unique to the slot number. BIOS sets this field and it remains set until a platform reset.
23:19	0h RW/O	**Physical Slot Number (PSN__23_19):** This is a value that is unique to the slot number. BIOS sets this field and it remains set until a platform reset.
18	1h RO	**No Command Completed Support (NCCS):** Set to '1' as this port does not implement a Hot Plug controller and can handle back-2-back writes to all fields of the slot control register without delay between successive writes.
17	0h RO	**Electromechanical Interlock Present (EMIP):** Set to 0 to indicate that no electro-mechanical interlock is implemented.
16:15	0h RW/O	**Slot Power Limit Scale (SLS):** specifies the scale used for the slot power limit value. BIOS sets this field and it remains set until a platform reset.
14:8	0h RW/O	**Slot Power Limit Value (SLV__14_8):** Specifies the upper limit (in conjunction with SLS value), on the upper limit on power supplied by the slot. The two values together indicate the amount of power in watts allowed for the slot. BIOS sets this field and it remains set until a platform reset.
7	0h RW/O	**Slot Power Limit Value (SLV__7_7):** Specifies the upper limit (in conjunction with SLS value), on the upper limit on power supplied by the slot. The two values together indicate the amount of power in watts allowed for the slot. BIOS sets this field and it remains set until a platform reset.
6	1h RW/O	**Hot Plug Capable (HPC):** When set, Indicates that hot plug is supported.
5	1h RW/O	**Hot Plug Surprise (HPS):** When set, indicates the device may be removed from the slot without prior notification.
4	0h RO	**Power Indicator Present (PIP):** Indicates that a power indicator LED is not present for this slot.
3	0h RO	**Attention Indicator Present (AIP):** Indicates that an attention indicator LED is not present for this slot.
2	0h RO	**MRL Sensor Present (MSP):** Indicates that an MRL sensor is not present
1	0h RO	**Power Controller Present (PCP):** Indicates that a power controller is not implemented for this slot
0	0h RO	**Attention Button Present (ABP):** Indicates that an attention button is not implemented for this slot.

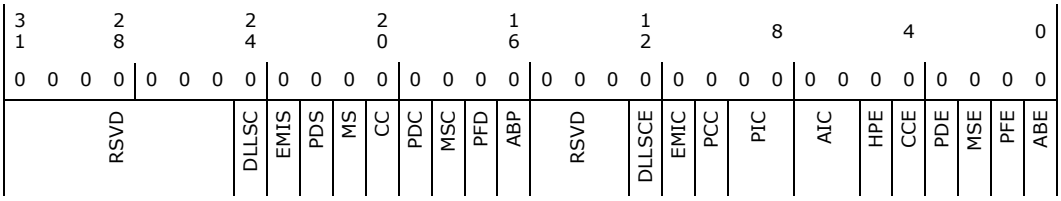
22.7.19 Slot Control; Slot Status (SLCTL_SLSTS)—Offset 58h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:25	0h RO	Reserved.
24	0h RW/1C/V	**Data Link Layer State Changed (DLLSC):** This bit is set when the value reported in Data Link Layer Link Active field of the Link Status register is changed. In response to a Data Link Layer State Changed event, software must read Data Link Layer Link Active field of the Link Status register to determine if the link is active before initiating configuration cycles to the hot plugged device.
23	0h RO	**Electromechanical Interlock Status (EMIS):** Reserved as this port does not support and electromechanical interlock.
22	0h RO/V	**Presence Detect State (PDS):** If XCAP.SI is set (indicating that this root port spawns a slot), then this bit indicates whether a device is connected ('1') or empty ('0'). If XCAP.SI is cleared, this bit is a '1'.
21	0h RO	**MRL Sensor State (MS):** Reserved as the MRL sensor is not implemented.
20	0h RO	**Command Completed (CC):** This register is RO as this port does not implement a Hot Plug Controller.
19	0h RW/1C/V	**Presence Detect Changed (PDC):** This bit is set by the root port when the SLSTS.PDS bit changes state.
18	0h RO	**MRL Sensor Changed (MSC):** Reserved as the MRL sensor is not implemented.
17	0h RO	**Power Fault Detected (PFD):** Reserved as a power controller is not implemented.
16	0h RO	**Attention Button Pressed (ABP):** This register is RO as this port does not implement an attention button
15:13	0h RO	Reserved.
12	0h RW	**Data Link Layer State Changed Enable (DLLSCE):** When set, this field enables generation of a hot plug interrupt when the Data Link Layer Link Active field is changed
11	0h RO	**Electromechanical Interlock Control (EMIC):** Reserved as this port does not support an Electromechanical Interlock.
10	0h RO	**Power Controller Control (PCC):** This bit has no meaning for module based hot plug.
9:8	0h RO	**Power Indicator Control (PIC):** This register is RO as this port does not implement a Hot Plug Controller.

Bit Range	Default and Access	Field Name (ID): Description
7:6	0h RO	**Attention Indicator Control (AIC):** This register is RO as this port does not implement a Hot Plug Controller.
5	0h RW	**Hot Plug Interrupt Enable (HPE):** When set, enables generation of a hot plug interrupt on enabled hot plug events.
4	0h RO	**Command Completed Interrupt Enable (CCE):** This register is RO as this port does not implement a Hot Plug Controller.
3	0h RW	**Presence Detect Changed Enable (PDE):** When set, enables the generation of a hot plug interrupt or wake message when the presence detect logic changes state.
2	0h RO	**MRL Sensor Changed Enable (MSE):** This register is RO as this port does not implement a Hot Plug Controller.
1	0h RO	**Power Fault Detected Enable (PFE):** This register is RO as this port does not implement a Hot Plug Controller...
0	0h RO	**Attention Button Pressed Enable (ABE):** This register is RO as this port does not implement a Hot Plug Controller.

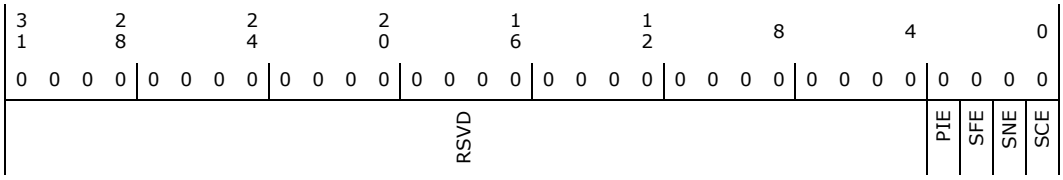
22.7.20 Root Control (RCTL)—Offset 5Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:4	0h RO	Reserved.
3	0h RW	**PME Interrupt Enable (PIE):** When set, enables interrupt generation when RSTS.PS is in a set state (either due to a '0' to '1' transition, or due to this bit being set with RSTS.PS already set).
2	0h RW	**System Error on Fatal Error Enable (SFE):** When set, an SERR# will be generated if a fatal error is reported by any of the devices in the hierarchy of this root port, including fatal errors in this root port. This register is not dependent on CMD.SEE being set.
1	0h RW	**System Error on Non-Fatal Error Enable (SNE):** When set, an SERR# will be generated if a non-fatal error is reported by any of the devices in the hierarchy of this root port, including non-fatal errors in this root port. This register is not dependent on CMD.SEE being set.
0	0h RW	**System Error on Correctable Error Enable (SCE):** When set, an SERR# will be generated if a correctable error is reported by any of the devices in the hierarchy of this root port, including correctable errors in this root port. This register is not dependent on CMD.SEE being set.

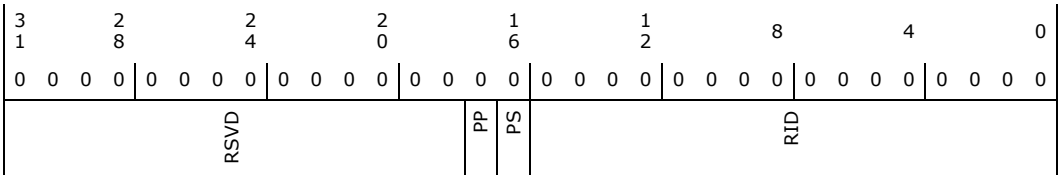
22.7.21 Root Status (RSTS)—Offset 60h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:18	0h RO	Reserved.
17	0h RO/V	**PME Pending (PP):** Indicates another PME is pending when the PME status bit is set. When the original PME is cleared by software, it will be set again, the requestor ID will be updated, and this bit will be cleared. Root ports have a one deep PME pending queue.
16	0h RW/1C/V	**PME Status (PS):** Indicates that PME was asserted by the requestor ID in RID. Subsequent PMEs are kept pending until this bit is cleared.
15:0	0h RO/V	**PME Requestor ID (RID):** Indicates the PCI requestor ID of the last PME requestor. Valid only when PS is set. Root ports are capable of storing the requester ID for two PM_PME messages, with one active (this register) and a one deep pending queue. Subsequent PM_PME messages will be dropped.

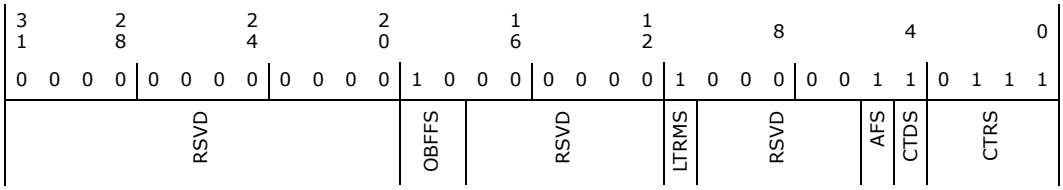
22.7.22 Device Capabilities 2 (DCAP2)—Offset 64h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 80837h



Bit Range	Default and Access	Field Name (ID): Description
31:20	0h RO	Reserved.
19:18	2h RW/O	**Optimized Buffer Flush/Fill Supported (OBFFS):** 00b - OBFF is not supported. 01b - OBFF is supported using Message signaling only. 10b - OBFF is supported using WAKE# signaling only. 11b - OBFF is supported using WAKE# and Message signaling. BIOS should program this field to 00b or 10b during system initialization to advertise the level of hardware OBFF support to software. BIOS should never program this field to 01b or 11b since OBFF messaging is not supported.
17:12	0h RO	Reserved.
11	1h RW/O	**LTR Mechanism Supported (LTRMS):** A value of 1b indicates support for the optional Latency Tolerance Reporting (LTR) mechanism capability. BIOS must write to this register with either a '1' or a '0' to enable/disable the root port from declaring support for the LTR capability.
10:6	0h RO	Reserved.
5	1h RO	**ARI Forwarding Supported (AFS):** ARI Forwarding Supported (AFS): Applicable only to Switch Downstream Ports and Root Ports; must be 0b for other Function types. This bit must be set to 1b if a Switch Downstream Port or Root Port supports this optional capability. ***Note:*** This bit is not made RWO to simplify implementation, since there is a requirement that the ARI Forwarding Enable bit must be hardwired to 0b if ARI Forwarding Supported bit is 0b. It is low risk to keep this risk 1b.
4	1h RO	**Completion Timeout Disable Supported (CTDS):** A value of 1b indicates support for the Completion Timeout Disable mechanism.

Bit Range	Default and Access	Field Name (ID): Description
3:0	7h RO	**Completion Timeout Ranges Supported (CTRS):** This field indicates device support for the optional Completion Timeout programmability mechanism. This mechanism allows system software to modify the Completion Timeout value. This field is applicable only to Root Ports, Endpoints that issue requests on their own behalf, and PCI Express to PCI/PCI-X Bridges that take ownership of requests issued on PCI Express. For all other devices this field is reserved and must be hardwired to 0000b. Four time value ranges are defined: Range A: 50us to 10ms Range B: 10ms to 250ms Range C: 250ms to 4s Range D: 4s to 64s Bits are set according to the table below to show timeout value ranges supported. 0000b Completion Timeout programming not supported. 0001b Range A 0010b Range B 0011b Ranges A [amp] B 0110b Ranges B [amp] C 0111b Ranges A, B [amp] C [lt]-- This is what PCH supports 1110b Ranges B, C [amp] D 1111b Ranges A, B, C [amp] D All other values are reserved.

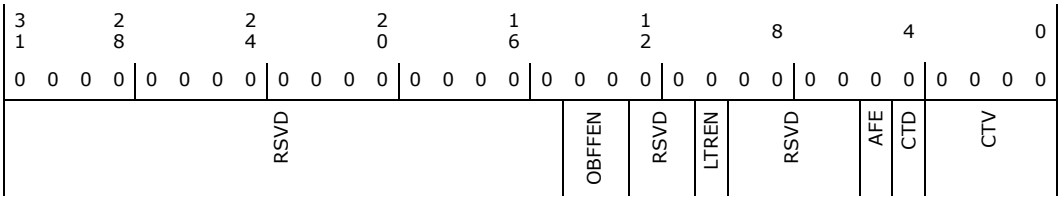
22.7.23 Device Control 2; Device Status 2 (DCTL2_DSTS2)—Offset 68h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:15	0h RO	Reserved.
14:13	0h RW	**Optimized Buffer Flush/Fill Enable (OBFFEN):** 00b Disable OBFF mechanism. 01b Enable OBFF mechanism using Message signaling (Variation A). 10b Enable OBFF mechanism using Message signaling (Variation B). 11b Enable OBFF using WAKE# signaling. ***Note:*** Only encoding 00b and 11b are supported. The encoding of 01b or 10b would be aliased to 00b. If DCAP2.OBFFS is clear, programming this field to any non-zero values will have no effect.
12:11	0h RO	Reserved.
10	0h RW	**LTR Mechanism Enable (LTREN):** When Set to 1b, this bit enables the Latency Tolerance Reporting (LTR) mechanism. For Downstream Ports, this bit must be reset to the default value if the Port goes to DL_Down status. If DCAP2.LTRMS is clear, programming this field to any non-zero values will have no effect.

Bit Range	Default and Access	Field Name (ID): Description
9:6	0h RO	Reserved.
5	0h RW	**ARI Forwarding Enable (AFE):** ARI Forwarding Enable (AFE): When set, the Downstream Port disables its traditional Device Number field being 0b enforcement when turning a Type 1 Configuration Request into a Type 0 Configuration Request, permitting access to Extended Functions in an ARI Device immediately below the Port.
4	0h RW	**Completion Timeout Disable (CTD):** When set to 1b, this bit disables the Completion Timeout mechanism. This field is required for all devices that support the Completion Timeout Disable Capability. Software is permitted to set or clear this bit at any time. When set, the Completion Timeout detection mechanism is disabled. If there are outstanding requests when the bit is cleared, it is permitted but not required for hardware to apply the completion timeout mechanism to the outstanding requests. If this is done, it is permitted to base the start time for each request on either the time this bit was cleared or the time each request was issued.
3:0	0h RW	**Completion Timeout Value (CTV):** In Devices that support Completion Timeout programmability, this field allows system software to modify the Completion Timeout value. This field is applicable to Root Ports, Endpoints that issue requests on their own behalf, and PCI Express to PCI/PCI-X Bridges that take ownership of requests issued on PCI Express. For all other devices this field is reserved and must be hardwired to 0000b. A Device that does not support this optional capability must hardwire this field to 0000b and is required to implement a timeout value in the range 50us to 50ms. Devices that support Completion Timeout programmability must support the values given below corresponding to the programmability ranges indicated in the Completion Timeout Values Supported field. The root port targeted configurable ranges are listed below, along with the range allowed by the PCI Express 2.0 specification. Defined encodings: 0000b Default range: 40-50ms (specification range 50us to 50ms) Values available if Range A (50 us to 10 ms) programmability range is supported: 0001b 90-100us (specification range is 50 us to 100 us) 0010b 9-10ms (specification range is 1 ms to 10 ms) Values available if Range B (10ms to 250ms) programmability range is supported: 0101b 40-50ms (specification range is 16 ms to 55 ms) 0110b 160-170ms (specification range is 65 ms to 210 ms) Values available if Range C (250ms to 4s) programmability range is supported: 1001b 400-500ms (specification range is 260 ms to 900 ms) 1010b 1.6-1.7s (specification range is 1s to 3.5s) Values not defined above are Reserved. Software is permitted to change the value in this field at any time. For requests already pending when the Completion Timeout Value is changed, hardware is permitted to use either the new or the old value for the outstanding requests, and is permitted to base the start time for each request either when this value was changed or when each request was issued.

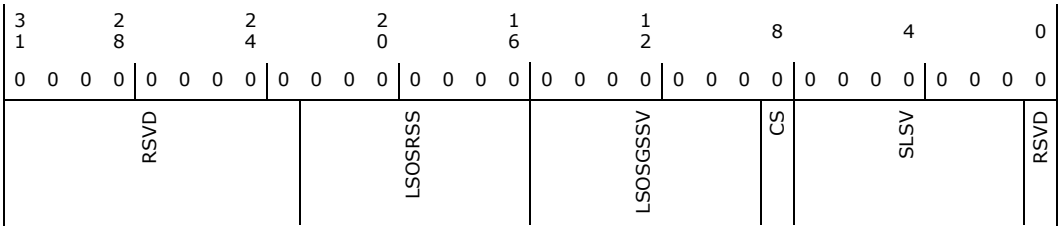
22.7.24 Link Capabilities 2 (LCAP2)—Offset 6Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:23	0h RO	Reserved.
22:16	0h RO	**Lower SKP OS Reception Supported Speeds Vector (LSOSRSS):** Lower SKP OS Reception Supported Speeds Vector(LSOSRSS): If this field is non-zero, it indicates that the Port, when operating at the indicated speed(s) supports SRIS and also supports receiving SKP OS at the rate defined for SRNS while running in SRIS. Bit definitions within this field are: Bit 0 2.5 GT/s Bit 1 5.0 GT/s Bit 2 8.0 GT/s Bits 6:3 RsvdP Behavior is undefined if a bit is set in this field and the corresponding bit is not set in the Supported Link Speeds Vector.
15:9	0h RO	**Lower SKP OS Generation Supported Speeds Vector (LSOSGSSV):** Lower SKP OS Generation Supported Speeds Vector(LSOSGSSV): If this field is non-zero, it indicates that the Port, when operating at the indicated speed(s) supports SRIS and also supports software control of the SKP Ordered Set transmission scheduling rate. Bit definitions within this field are: Bit 0 2.5 GT/s Bit 1 5.0 GT/s Bit 2 8.0 GT/s Bits 6:3 RsvdP Behavior is undefined if a bit is set in this field and the corresponding bit is not set in the Supported Link Speeds Vector.
8	0h RO	**Crosslink Supported (CS):** Crosslink Supported (CS): No support for Crosslink.
7:1	0h RO/V	**Supported Link Speeds Vector (SLSV):** Supported Link Speeds Vector (SLSV): This field indicates the supported Link speed of the associated Port. For each bit, a value of 1b indicates that the corresponding Link speed is supported; otherwise, the Link speed is not supported. Bit definitions within this field are: Bit 0: 2.5 GT/s. Bit 1: 5.0 GT/s. Bit 2: 8.0 GT/s. Bits 6:3: Reserved. This register reports a value of 0001b if GEN1 data rate is supported but both GEN2 and GEN3 data rate support are disabled through Fuse or MPC.PCIESD register. This register reports a value of 0011b if both GEN1 and GEN2 data rate are supported but GEN3 data rate support is disabled through Fuse or MPC.PCIESD register.
0	0h RO	Reserved.

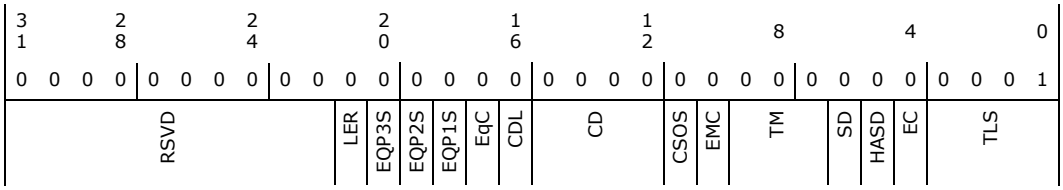
22.7.25 Link Control 2; Link Status 2 (LCTL2_LSTS2)—Offset 70h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 1h



Bit Range	Default and Access	Field Name (ID): Description
31:22	0h RO	Reserved.
21	0h RW/1C/V/P	**Link Equalization Request (LER):** Link Equalization Request (LER): This bit is set by hardware to request the Link equalization process to be performed on the Link. Register Attribute: Dynamic.
20	0h RO/V/P	**Equalization Phase 3 Successful (EQP3S):** Equalization Phase 3 Successful (EQP3S): When set to 1, this bit indicates that Phase 3 of the Transmitter Equalization procedure has successfully completed.
19	0h RO/V/P	**Equalization Phase 2 Successful (EQP2S):** Equalization Phase 2 Successful (EQP2S): When set to 1, this bit indicates that Phase 2 of the Transmitter Equalization procedure has successfully completed.
18	0h RO/V/P	**Equalization Phase 1 Successful (EQP1S):** Equalization Phase 1 Successful (EQP1S): When set to 1, this bit indicates that Phase 1 of the Transmitter Equalization procedure has successfully completed.
17	0h RO/V/P	**Equalization Complete (EqC):** Equalization Complete (EC): When set to 1, this bit indicates that the Transmitter Equalization procedure has completed
16	0h RO/V	**Current De-emphasis Level (CDL):** When the Link is operating at 5.0 GT/s speed, this bit reflects the level of de-emphasis. Encodings: 1b -3.5 dB 0b -6 dB The value in this bit is undefined when the Link is not operating at 5.0 GT/s speed.
15:12	0h RW/P	**Compliance Preset/De-emphasis (CD):** For 8.0 GT/s Data Rate: This field sets the Transmitter Preset in Polling.Compliance state if the entry occurred due to the Enter Compliance bit being 1b. Results are undefined if a reserved preset encoding is used when entering Polling.Compliance in this way. For 5.0 GT/s Data Rate: This bit sets the de-emphasis level in Polling.Compliance state if the entry occurred due to the Enter Compliance bit being 1b. Encodings: 0001b -3.5 dB 0000b -6 dB When the Link is not operating at 5.0 GT/s speed, the setting of this bit has no effect. The default value of this field is 0000b. This bit is intended for debug, compliance testing purposes. System firmware and software is allowed to modify this bit only during debug or compliance testing. In all other cases, the system must ensure that this field is set to the default value.
11	0h RW/P	**Compliance SOS (CSOS):** When set to 1b, the LTSSM is required to send SKP Ordered Sets periodically in between the (modified) compliance patterns. The default value of this bit is 0b. This bit is applicable when the Link is operating at 2.5 GT/s or 5.0 GT/s data rates only.

Bit Range	Default and Access	Field Name (ID): Description
10	0h RW/P	**Enter Modified Compliance (EMC):** When this bit is set to 1b, the device transmits Modified Compliance Pattern if the LTSSM enters Polling.Compliance substate. Default value of this bit is 0b. This register is intended for debug, compliance testing purposes only. System firmware and software is allowed to modify this register only during debug or compliance testing. In all other cases, the system must ensure that this register is set to the default value.
9:7	0h RW/P	**Transmit Margin (TM):** This field controls the value of the nondeemphasized voltage level at the Transmitter pins. This field is reset to 000b on entry to the LTSSM Polling.Configuration substate (see PCI Express Chapter 4 for details of how the Transmitter voltage level is determined in various states). Encodings: 000b Normal operating range 001b 800-1200 mV for full swing and 400-700 mV for half-swing 010b - (n-1) Values must be monotonic with a non-zero slope. The value of n must be greater than 3 and less than 7. At least two of these must be below the normal operating range of n : 200-400 mV for full-swing and 100-200 mV for half-swing n - 111b reserved For a Multi-Function device associated with an Upstream Port, the field in Function 0 is of type RWS, and only Function 0 controls the component's Link behavior. In all other Functions of that device, this field is of type RsvdP. Default value of this field is 000b. Components that support only the 2.5 GT/s speed are permitted to hardwire this bit to 000b. This register is intended for debug, compliance testing purposes only. System firmware and software is allowed to modify this register only during debug or compliance testing. In all other cases, the system must ensure that this register is set to the default value.
6	0h RW/P	**Selectable De-emphasis (SD):** When the Link is operating at 5.0 GT/s speed, this bit selects the level of de-emphasis for an Upstream component. Encodings: 1b -3.5 dB 0b -6 dB When the Link is not operating at 5.0 GT/s speed, the setting of this bit has no effect. When the Link is operating at 2.5 GT/s speed, the setting of this bit has no effect.
5	0h RO	**Hardware Autonomous Speed Disable (HASD):** Reserved. This port cannot autonomously change speeds.
4	0h RW/P	**Enter Compliance (EC):** Software is permitted to force a Link to enter Compliance mode at the speed indicated in the Target Link Speed field by setting this bit to 1b in both components on a Link and then initiating a hot reset on the Link. Default value of this bit following Fundamental Reset is 0b. This bit is intended for debug, compliance testing purposes only. System firmware and software is allowed to modify this bit only during debug or compliance testing. In all other cases, the system must ensure that this bit is set to the default value. This bit is intended for debug, compliance testing purposes only. System firmware and software is allowed to modify this bit only during debug or compliance testing. In all other cases, the system must ensure that this bit is set to the default value.
3:0	1h RW/V/P	**Target Link Speed (TLS):** Target Link Speed (TLS): This field sets an upper limit on Link operational speed by restricting the values advertised by the upstream component in its training sequences. The encoded value specifies a bit location in the Supported Link Speeds Vector (in the Link Capabilities 2 register) that corresponds to the current Link speed. Defined encodings are: 0001b: Supported Link Speeds Vector field bit 0. 0010b: Supported Link Speeds Vector field bit 1. 0011b: Supported Link Speeds Vector field bit 2. 0100b: Supported Link Speeds Vector field bit 3. 0101b: Supported Link Speeds Vector field bit 4. 0110b: Supported Link Speeds Vector field bit 5. 0111b: Supported Link Speeds Vector field bit 6. All other encodings are reserved. If a value is written to this field that does not correspond to a supported speed, as indicated by the Supported Link Speeds Vector, the result is undefined. The default value of this field is GEN1. ***Note:*** This register field could be used by REUT software to limit the link speed to 2.5 GT/s or 5 GT/s data rate.

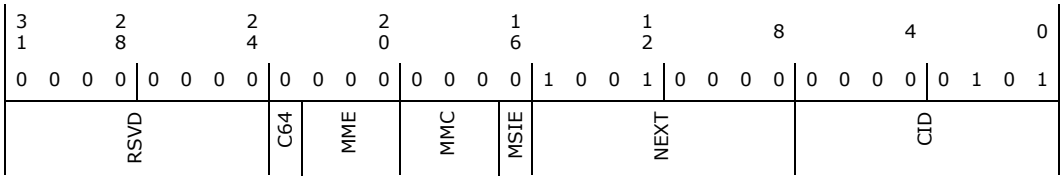
22.7.26 Message Signaled Interrupt Identifiers; Message Signaled Interrupt Message Control (MID_MC)—Offset 80h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 9005h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23	0h RO	**64-Bit Address Capable (C64):** Capable of generating a 32-bit message only.
22:20	0h RW	**Multiple Message Enable (MME):** These bits are RW for software compatibility, but only one message is ever sent by the root port.
19:17	0h RO	**Multiple Message Capable (MMC):** Only one message is required.
16	0h RW	**MSI Enable (MSIE):** If set, MSI is enabled and traditional interrupt pins are not used to generate interrupts. CMD.BME must be set for an MSI to be generated. If CMD.BME is cleared, and this bit is set, no interrupts (not even pin based) are generated.
15:8	90h RW/O	**Next Pointer (NEXT):** Indicates the location of the next capability in the list. The default value of this register is 90h which points to the Subsystem Vendor capability structure. BIOS can determine which capabilities will be exposed by including or removing them from the capability linked list. As this register is RWO, BIOS must write a value to this register, even if it is to re-write the default value.
7:0	5h RO	**Capability ID (CID):** Capabilities ID indicates MSI.

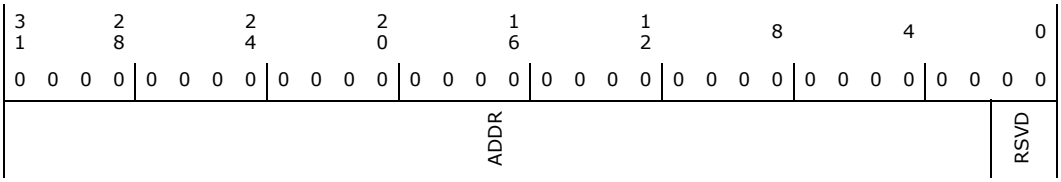
22.7.27 Message Signaled Interrupt Message Address (MA)—Offset 84h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:2	0h RW	**Address (ADDR):** Lower 32 bits of the system specified message address, always DW aligned.
1:0	0h RO	Reserved.

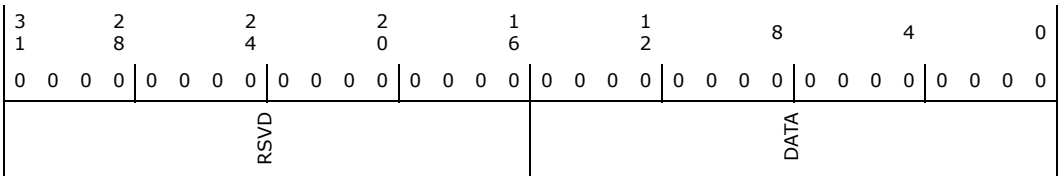
22.7.28 Message Signaled Interrupt Message Data (MD)—Offset 88h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:0	0h RW	**Data (DATA):** This 16-bit field is programmed by system software if MSI is enabled. Its content is driven onto the lower word (PCI AD[lb]15:0[rb]) during the data phase of the MSI memory write transaction.

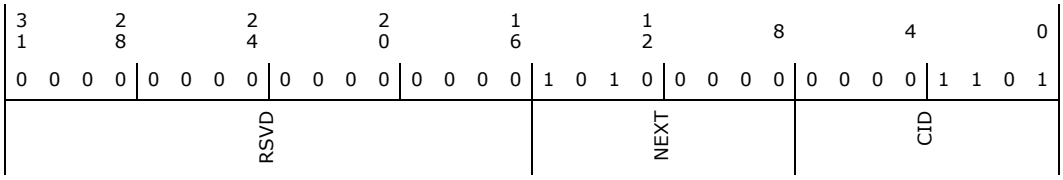
22.7.29 Subsystem Vendor Capability (SVCAP)—Offset 90h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: A00Dh



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:8	A0h RW/O	**Next Capability (NEXT):** Indicates the location of the next capability in the list. The default value of this register is A0h which points to the PCI Power Management capability structure. BIOS can determine which capabilities will be exposed by including or removing them from the capability linked list. As this register is RWO, BIOS must write a value to this register, even if it is to re-write the default value.
7:0	Dh RO	**Capability Identifier (CID):** Value of 0Dh indicates this is a PCI bridge subsystem vendor capability.

22.7.30 Subsystem Vendor IDs (SVID)—Offset 94h

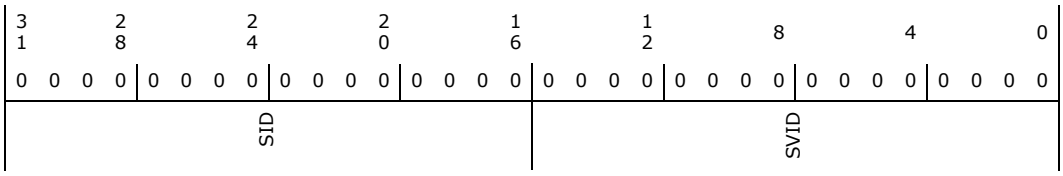
Size:32 bits

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RW/O	**Subsystem Identifier (SID):** Indicates the subsystem as identified by the vendor. This field is write once and is locked down until a bridge reset occurs (not the PCI bus reset).
15:0	0h RW/O	**Subsystem Vendor Identifier (SVID):** Indicates the manufacturer of the subsystem. This field is write once and is locked down until a bridge reset occurs (not the PCI bus reset).

22.7.31 Power Management Capability; PCI Power Management Capabilities (PMCAP_PMC)—Offset A0h

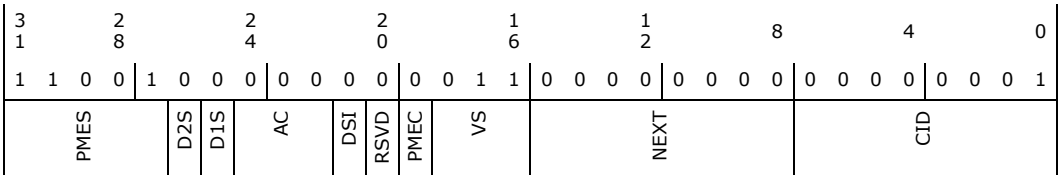
Size:32 bits

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: C8030001h



Bit Range	Default and Access	Field Name (ID): Description
31:27	19h RO	**PME Support (PMES):** Indicates PME# is supported for states D0, $D3_{HOT}$ and $D3_{COLD}$. The root port does not generate PME#, but reporting that it does is necessary for legacy Microsoft operating systems to enable PME# in devices connected behind this root port.
26	0h RO	**D2_Support (D2S):** The D2 state is not supported.
25	0h RO	**D1_Support (D1S):** The D1 state is not supported.
24:22	0h RO	**Aux_Current (AC):** Reports 0mA (self-powered), as use of this controller does not add to suspect well power consumption.
21	0h RO	**Device Specific Initialization (DSI):** Indicates that no device-specific initialization is required.
20	0h RO	Reserved.
19	0h RO	**PME Clock (PMEC):** Indicates that PCI clock is not required to generate PME#.
18:16	3h RO	**Version (VS):** Indicates support for Revision 1.2 of the PCI Power Management Specification.
15:8	0h RO	**Next Capability (NEXT):** Indicates this is the last item in the list.
7:0	1h RO	**Capability Identifier (CID):** Value of 01h indicates this is a PCI power management capability.

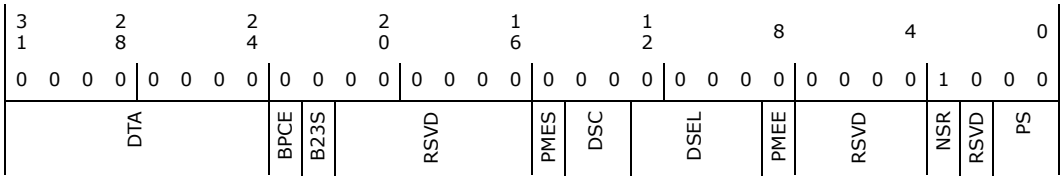
22.7.32 PCI Power Management Control and Status (PMCS)—Offset A4h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 8h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	**Data (DTA):** Reserved
23	0h RO	**Bus Power / Clock Control Enable (BPCE):** Reserved per PCI Express specification
22	0h RO	**B2/B3 Support (B23S):** Reserved per PCI Express specification.
21:16	0h RO	Reserved.
15	0h RO	**PME Status (PMES):** Indicates a PME was received on the downstream link.
14:13	0h RO	**Data Scale (DSC):** Reserved
12:9	0h RO	**Data Select (DSEL):** Reserved
8	0h RW/P	**PME Enable (PMEE):** Indicates PME is enabled. The root port takes no action on this bit, but it must be RW for legacy Microsoft operating systems to enable PME# on devices connected to this root port.
7:4	0h RO	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
3	1h RW/O	**No Soft Reset (NSR):** When set to 1 this bit indicates that devices transitioning from $D3_{HOT}$ to D0 because of Power State commands do not perform an internal reset. Configuration context is preserved. Upon transition from $D3_{HOT}$ to D0 Initialized state, no additional operating system intervention is required to preserve Configuration Context beyond writing the Power State bits. When clear, devices do perform an internal reset upon transitioning from $D3_{HOT}$ to D0 via software control of the Power State bits. Configuration Context is lost when performing the soft reset. Upon transition from $D3_{HOT}$ to D0 state, full reinitialization sequence is needed to return the device to D0 Initialized. Regardless of this bit, devices that transition from $D3_{HOT}$ to D0 by a system or bus segment reset will return to the device state D0 Uninitialized with only PME context preserved if PME is supported and enabled.
2	0h RO	Reserved.
1:0	0h RW	**Power State (PS):** This field is used both to determine the current power state of the root port and to set a new power state. The values are: 00 D0 state 11 $D3_{HOT}$ state When in the $D3_{HOT}$ state, the controller's configuration space is available, but the I/O and memory spaces are not. Type 1 configuration cycles are also not accepted. Interrupts are not required to be blocked as software will disable interrupts prior to placing the port into $D3_{HOT}$. If software attempts to write a '10' or '01' to these bits, the write will be ignored.

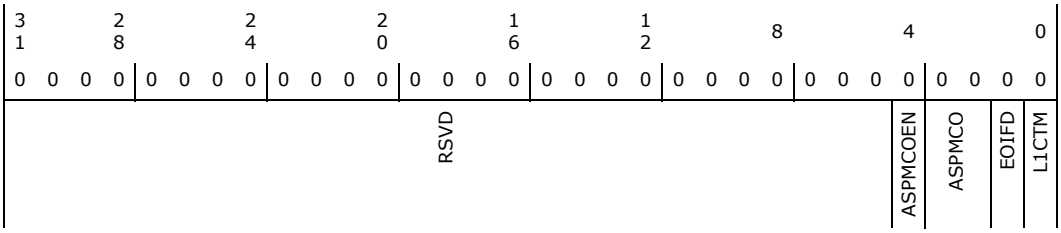
22.7.33 Miscellaneous Port Configuration 2 (MPC2)—Offset D4h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:5	0h RO	Reserved.
4	0h RW	**ASPM Control Override Enable (ASPMCOEN):** When set to '1', the PCIe Root Port will use the values in the ASPM Control Override registers instead of ASPM Registers in the Link Control register. This register allows BIOS to control the DMI ASPM settings instead of the OS.
3:2	0h RW	**ASPM Control Override (ASPMCO):** Provides BIOS control of whether root port should enter L0s or L1 or both. 00 = Disabled 01 = L0s Entry Enabled 10 = L1 Entry Enabled 11 = L0s and L1 Entry Enabled.
1	0h RW	**EOI Forwarding Disable (EOIFD):** 0 = Broadcast EOI messages that are sent on the backbone are claimed by this port and forwarded across the PCIe* link.{br] 1 = Broadcast EOI messages are not claimed on the backbone by this port and will not be forwarded across the PCIe* Link.

Bit Range	Default and Access	Field Name (ID): Description
0	0h RW	**L1 Completion Timeout Mode (L1CTM):** 0 = PCI Express* Specification Compliant. Completion timeout is disabled during software initiated L1, and enabled during ASPM initiate L1. 1 = Completion timeout is enabled during L1, regardless of how L1 entry was initiated.

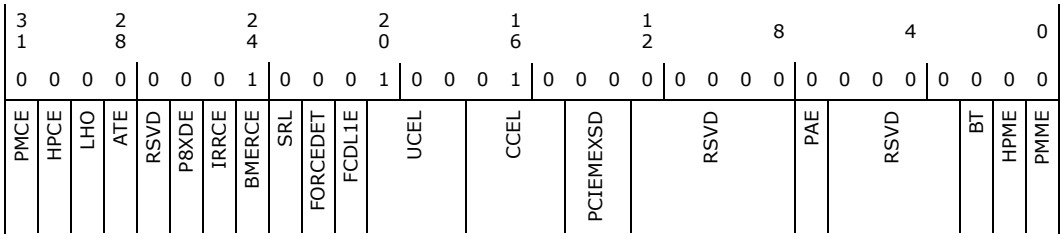
22.7.34 Miscellaneous Port Configuration (MPC)—Offset D8h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 1110000h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RW	**Power Management SCI Enable (PMCE):** 0 = SCI generation based on a power management event is disabled. 1 = Enables the root port to generate SCI whenever a power management event is detected.
30	0h RW	**Hot Plug SCI Enable (HPCE):** 0 = SCI generation based on a hot-plug event is disabled. 1 = Enables the root port to generate SCI whenever a hot-plug event is detected.
29	0h RW/L	**Link Hold Off (LHO):** When set, the port will not take any TLP. This is used during loopback mode to fill up the downstream queue.
28	0h RW/L	**Address Translater Enable (ATE):** Used to enable address translation via the AT bits in this register during loopback mode. 0: Disable 1: Enable
27	0h RO	Reserved.
26	0h RW/L	**Port8xh Decode Enable (P8XDE):** When set, allows PCIe Root Port to claim I/O cycles within the range from 80h - 8Fh inclusive and forwarding the cycle to the link. The claiming of these cycles are independent of I/O Base/Limit and I/O Space Enable bits. BIOS must ensure that at any one time, no more than one PCIe Root Port are enabled to claim Port 8xh cycles.
25	0h RW/L	**Invalid Receive Range Check Enable (IRRCE):** When set, the receive transaction layer will treat the TLP as an Unsupported Request error if the address range of a Memory request does not fall outside the range between prefetchable and non-prefetchable base and limit. Messages, I/O, Config, and Completions are never checked for valid address ranges This register bit is Read-Only when the MPC.SRL bit is set.
24	1h RW/L	**BME Receive Check Enable (BMERCE):** When set, the receive transaction layer will treat the TLP as an Unsupported Request error if a memory or I/O read or write request is received and the Bus Master Enable bit is not set. Messages, Config, and Completions are never checked for BME.

Bit Range	Default and Access	Field Name (ID): Description
23	0h RW/O	**Secured Register Lock (SRL):** When this bit is set, all the secured registers will be locked and will be Read-Only.
22	0h RW/L	**Detect Override (FORCEDET):** 0: Normal operation. Detect output from AFE is sampled for presence detection. 1: Override mode. Ignores AFE detect output and Link Training proceeds as if a device were detected.
21	0h RW	**Flow Control During L1 Entry (FCDL1E):** br] 0: No flow control update DLLPs sent during L1 Ack transmission 1: Flow control update DLLPs sent during L1 Ack transmission as required to meet the 30us periodic flow control update.
20:18	4h RW	**Unique Clock Exit Latency (UCEL):** This value represents the L0s Exit Latency for unique-clock configurations (LCTL.CCC = '0'). It defaults to 512ns to less than 1us, but may be overridden by BIOS.
17:15	2h RW	**Common Clock Exit Latency (CCEL):** This value represents the L0s Exit Latency for common-clock configurations (LCTL.CCC = '1'). It defaults to 128ns to less than 256ns, but may be overridden by BIOS.
14:13	0h RW	**PCIe MEx Speed Disable (PCIEMEXSD):** When operating as PCI Express: 00: PCIe supported data rate is as defined by the Supported Link Speed and Target Link Speed register. 01: PCIe supported data rate is limited to just 2.5 GT/s. Supported Link Speed field will reflect 0000001b. Max Link Speed field will reflect 0001b. 10: PCIe supported data rate is limited to 2.5 GT/s and 5.0 GT/s. Supported Link Speed register will reflect 0000011b. Max Link Speed field will reflect 0010b. 11: Reserved. When operating as Mobile Express: 00: MEx supported data rate is as defined by the Supported Link Speed and Target Link Speed register. 01: MEx supported data rate is limited to just HS-G1. Supported Link Speed field will reflect 0000001b. Max Link Speed field will reflect 0000b. 10: PCIe supported data rate is limited to HS-G1 and HS-G2. Supported Link Speed register will reflect 0000001b. Max Link Speed field will reflect 0000b. 11: Reserved. When this bit is changed, link retrain needs to be performed for the change to be effective.
12:8	0h RO	Reserved
7	0h RW	**Port I/OxApic Enable (PAE):** When set, a range is opened through the bridge for the following memory addresses: Port# Address 1 FEC1_0000h - FEC1_7FFFh 2 FEC1_8000h - FEC1_FFFFh 3 FEC2_0000h - FEC2_7FFFh 4 FEC2_8000h - FEC2_FFFFh 5 FEC3_0000h - FEC3_7FFFh 6 FEC3_8000h - FEC3_FFFFh 7 FEC4_0000h - FEC4_7FFFh 8 FEC4_8000h - FEC4_FFFFh When cleared, the hole is disabled.
6:3	0h RO	Reserved.
2	0h RW/O	**Bridge Type (BT):** This register can be used to modify the Base Class and Header Type fields from the default PCI-to-PCI bridge to a Host Bridge. Having the root port appear as a Host Bridge is useful in some server configurations. 0 = The root port bridge type is a PCI-to-PCI Bridge, Header Sub-Class = 04h, and Header Type = Type 1. 1 = The root port bridge type is a PCI-to-PCI Bridge, Header Sub-Class = 00h, and Header Type = Type 0.
1	0h RW	**Hot Plug SMI Enable (HPME):** 0 = SMI generation based on a hot-plug event is disabled. 1 = Enables the root port to generate SMI whenever a hot-plug event is detected.
0	0h RW	**Power Management SMI Enable (PMME):** 0 = SMI generation based on a power management event is disabled. 1 = Enables the root port to generate SMI whenever a power management event is detected.

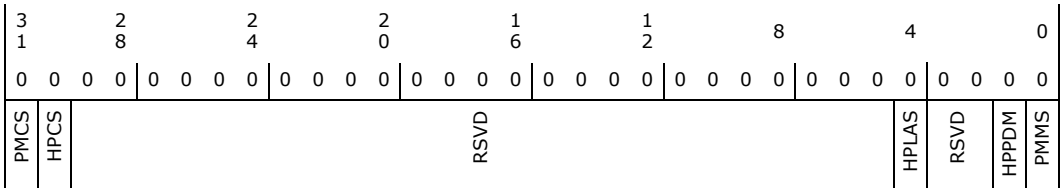
22.7.35 SMI / SCI Status (SMSCS)—Offset DCh

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RW/1C/V	**Power Management SCI Status (PMCS):** This bit is set if the root port PME control logic needs to generate an interrupt, and this interrupt has been routed to generate an SCI.
30	0h RW/1C/V	**Hot Plug SCI Status (HPCS):** This bit is set if the hot plug controller needs to generate an interrupt, and has this interrupt been routed to generate an SCI.
29:5	0h RO	Reserved.
4	0h RW/1C/V	**Hot Plug Link Active State Changed SMI Status (HPLAS):** This bit is set when SLSTS.DLLSC transitions from '0' to '1', and MPC.HPME is set. When this bit is set, an SMI# will be generated.
3:2	0h RO	Reserved.
1	0h RW/1C/V	**Hot Plug Presence Detect SMI Status (HPPDM):** This bit is set when SLSTS.PDC transitions from '0' to'1', and MPC.HPME is set. When this bit is set, an SMI# will be generated.
0	0h RW/1C/V	**Power Management SMI Status (PMMS):** This bit is set when RSTS.PS transitions from '0' to'1', and MPC.PMME is set.

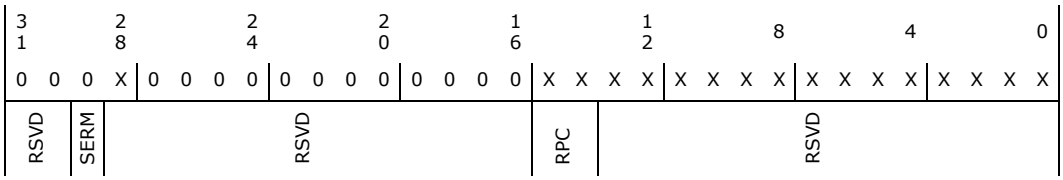
22.7.36 Strap and Fuse Configuration (STRPFUSECFG)—Offset FCh

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:30	00b RO	**Reserved**
29	0b RW	**Server Error Reporting Mode (SERM):** When set, if the PCIe port detects a fatal, non-fatal, or correctable error on the link, it sends one of ERR_FATAL, ERR_NONFATAL, or ERR_CORR message to the IOSF primary fabric. If the PCIe port receives an ERR_* message from the downstream device, it sends the message to the IOSF primary fabric. ***Note:*** For each x4 instance, only the value from Port 0 is used.
28:16	x00000000 0000b RO/W/S	**Reserved**
15:14	XX R0/s	**Root Port Configuration Strap (RPC):** The reflects that state of the soft straps. 11: 1x4 Port 0 (x4), Ports 1-3 (disabled 10: 2x2 Port 0 (x2), Port 2 (x2), Ports 1,3 (disabled) 01: 1x2, 2x1, Port 0 (x2), Port 1 (disabled), Ports 2,3 (x1) 00: 4x1 Ports 0-3 (x1)
13:0	xxh RO	**Reserved**

22.7.37 Advanced Error Extended Reporting Capability Header (AECH)—Offset 100h

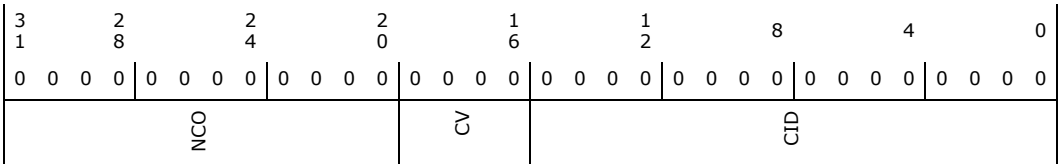
Size:32 bits
The AER capability can optionally be included or excluded from the capabilities list. The full AER is supported.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:20	0h RW/O	**Next Capability Offset (NCO):** Set to 000h as this is the last capability in the list.
19:16	0h RW/O	**Capability Version (CV):** For systems that support AER, BIOS should write a 1h to this register else it should write 0.
15:0	0h RW/O	**Capability ID (CID):** For systems that support AER, BIOS should write a 0001h to this register else it should write 0.

22.7.38 Uncorrectable Error Status (UES)—Offset 104h

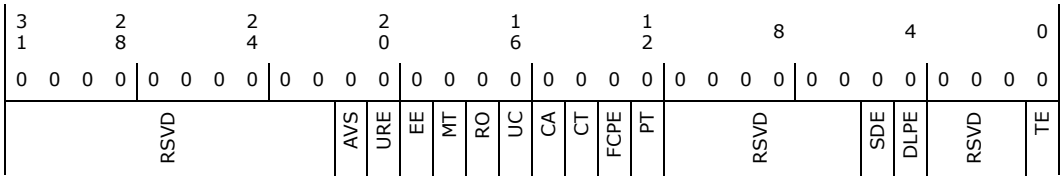
This register must maintain its state through a platform reset. It loses its state upon loss of core power.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:22	0h RO	Reserved.
21	0h RW/1C/V/P	**ACS Violation Status (AVS):** Reserved. Access Control Services are not supported
20	0h RW/1C/V/P	**Unsupported Request Error Status (URE):** Indicates an unsupported request was received.
19	0h RO	**ECRC Error Status (EE):** ECRC is not supported.
18	0h RW/1C/V/P	**Malformed TLP Status (MT):** Indicates a malformed TLP was received.
17	0h RW/1C/V/P	**Receiver Overflow Status (RO):** Indicates a receiver overflow occurred.
16	0h RW/1C/V/P	**Unexpected Completion Status (UC):** Indicates an unexpected completion was received.
15	0h RW/1C/V/P	**Completor Abort Status (CA):** Indicates a completer abort was received
14	0h RW/1C/V/P	**Completion Timeout Status (CT):** Indicates a completion timed out. This is signaled if Completion Timeout is enabled and a completion fails to return within the amount of time specified by the Completion Timeout Value
13	0h RO	**Flow Control Protocol Error Status (FCPE):** Not supported.
12	0h RW/1C/V/P	**Poisoned TLP Status (PT):** Indicates a poisoned TLP was received.
11:6	0h RO	Reserved.
5	0h RO	**Surprise Down Error Status (SDE):** Surprise Down is not supported.

Bit Range	Default and Access	Field Name (ID): Description
4	0h RW/1C/V/P	**Data Link Protocol Error Status (DLPE):** Indicates a data link protocol error occurred.
3:1	0h RO	Reserved.
0	0h RO	**Training Error Status (TE):** Not supported.

22.7.39 Uncorrectable Error Mask (UEM)—Offset 108h

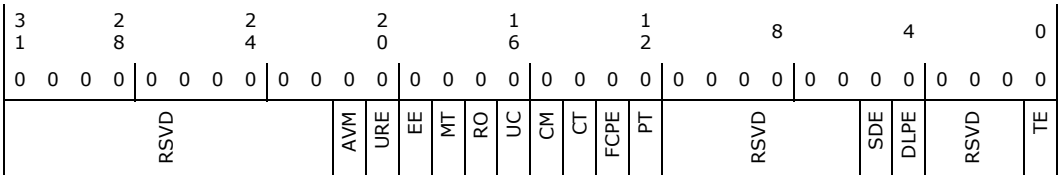
When set, the corresponding error in the UES register is masked, and the logged error will cause no action. When cleared, the corresponding error is enabled. This register is only reset by a loss of core power.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:22	0h RO	Reserved.
21	0h RW/P	**ACS Violation Mask (AVM):** Reserved. Access Control Services are not supported
20	0h RW/P	**Unsupported Request Error Mask (URE):** Mask for uncorrectable errors.
19	0h RO	**ECRC Error Mask (EE):** ECRC is not supported.
18	0h RW/P	**Malformed TLP Mask (MT):** Mask for malformed TLPs
17	0h RW/P	**Receiver Overflow Mask (RO):** Mask for receiver overflows.
16	0h RW/P	**Unexpected Completion Mask (UC):** Mask for unexpected completions.
15	0h RW/P	**Completor Abort Mask (CM):** Mask for completer abort.
14	0h RW/P	**Completion Timeout Mask (CT):** Mask for completion timeouts.

Bit Range	Default and Access	Field Name (ID): Description
13	0h RO	**Flow Control Protocol Error Mask (FCPE):** Not supported.
12	0h RW/P	**Poisoned TLP Mask (PT):** Mask for poisoned TLPs.
11:6	0h RO	Reserved.
5	0h RO	**Surprise Down Error Mask (SDE):** Surprise Down is not supported.
4	0h RW/P	**Data Link Protocol Error Mask (DLPE):** Mask for data link protocol errors.
3:1	0h RO	Reserved.
0	0h RO	**Training Error Mask (TE):** Not supported.

22.7.40 Uncorrectable Error Severity (UEV)—Offset 10Ch

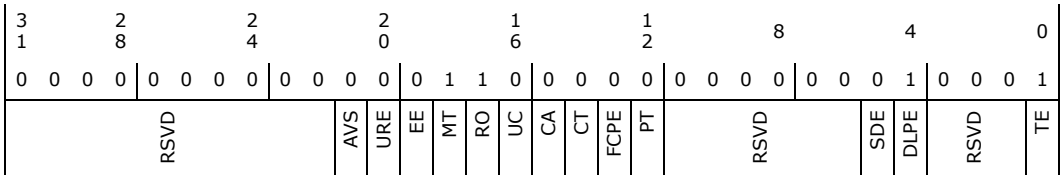
This register gives the option to make an uncorrectable error fatal or non-fatal. An error is fatal if the bit is set. An error is non-fatal if the bit is cleared. This register is only reset by a loss of core power.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 60011h



Bit Range	Default and Access	Field Name (ID): Description
31:22	0h RO	Reserved.
21	0h RW/P	**ACS Violation Severity (AVS):** Severity for ACS violation.
20	0h RW/P	**Unsupported Request Error Severity (URE):** Severity for unsupported request reception.
19	0h RO	**ECRC Error Severity (EE):** ECRC is not supported.
18	1h RW/P	**Malformed TLP Severity (MT):** Severity for malformed TLP reception.
17	1h RW/P	**Receiver Overflow Severity (RO):** Severity for receiver overflow occurrences.

Bit Range	Default and Access	Field Name (ID): Description
16	0h RW/P	**Unexpected Completion Severity (UC):** Severity for unexpected completion reception.
15	0h RW/P	**Completor Abort Severity (CA):** Severity for completer abort.
14	0h RW/P	**Completion Timeout Severity (CT):** Severity for completion timeout.
13	0h RO	**Flow Control Protocol Error Severity (FCPE):** Not supported.
12	0h RW/P	**Poisoned TLP Severity (PT):** Severity for poisoned TLP reception.
11:6	0h RO	Reserved.
5	0h RO	**Surprise Down Error Severity (SDE):** Surprise Down is not supported.
4	1h RW/P	**Data Link Protocol Error Severity (DLPE):** Severity for data link protocol errors.
3:1	0h RO	Reserved.
0	1h RO	**Training Error Severity (TE):** TE not supported. This bit is left as RO='1' for ease of implementation.

22.7.41 Correctable Error Status (CES)—Offset 110h

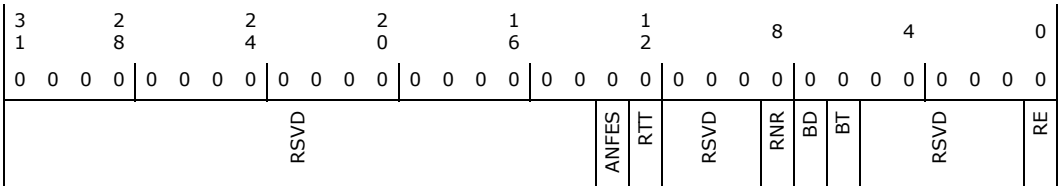
This register is only reset by a loss of core power.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13	0h RW/1C/V/P	**Advisory Non-Fatal Error Status (ANFES):** When set, indicates that an Advisory Non-Fatal Error occurred.
12	0h RW/1C/V/P	**Replay Timer Timeout Status (RTT):** Indicates the replay timer timed out.

Bit Range	Default and Access	Field Name (ID): Description
11:9	0h RO	Reserved.
8	0h RW/1C/V/P	**Replay Number Rollover Status (RNR):** Indicates the replay number rolled over.
7	0h RW/1C/V/P	**Bad DLLP Status (BD):** Indicates a bad DLLP was received.
6	0h RW/1C/V/P	**Bad TLP Status (BT):** Indicates a bad TLP was received.
5:1	0h RO	Reserved.
0	0h RW/1C/V/P	**Receiver Error Status (RE):** Indicates a receiver error occurred.

22.7.42 Correctable Error Mask (CEM)—Offset 114h

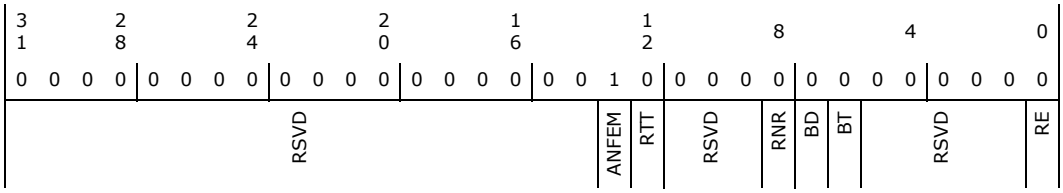
When set, the corresponding error in the CES register is masked, and the logged error will cause no action. When cleared, the corresponding error is enabled. This register is only reset by a loss of core power.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 2000h



Bit Range	Default and Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13	1h RW/P	**Advisory Non-Fatal Error Mask (ANFEM):** When set, masks Advisory Non-Fatal errors from (a) signaling ERR_COR to the device control register and (b) updating the Uncorrectable Error Status register. This register is set by default to enable compatibility with software that does not comprehend Role-Based Error Reporting.
12	0h RW/P	**Replay Timer Timeout Mask (RTT):** Mask for replay timer timeout.
11:9	0h RO	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
8	0h RW/P	**Replay Number Rollover Mask (RNR):** Mask for replay number rollover.
7	0h RW/P	**Bad DLLP Mask (BD):** Mask for bad DLLP reception.
6	0h RW/P	**Bad TLP Mask (BT):** Mask for bad TLP reception.
5:1	0h RO	Reserved.
0	0h RW/P	**Receiver Error Mask (RE):** Mask for receiver errors.

22.7.43 Advanced Error Capabilities and Control (AECC)—Offset 118h

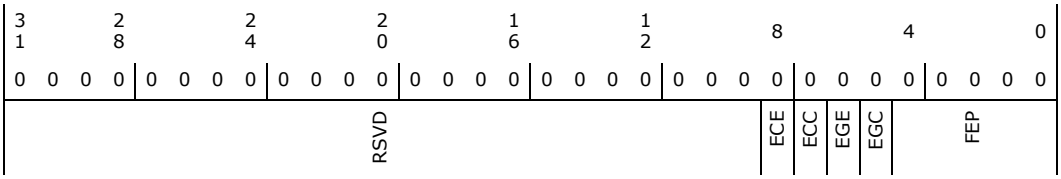
This register is only reset by a loss of core power.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	Reserved.
8	0h RO	**ECRC Check Enable (ECE):** ECRC is not supported.
7	0h RO	**ECRC Check Capable (ECC):** ECRC is not supported.
6	0h RO	**ECRC Generation Enable (EGE):** ECRC is not supported.
5	0h RO	**ECRC Generation Capable (EGC):** ECRC is not supported.
4:0	0h RO/V/P	**First Error Pointer (FEP):** Identifies the bit position of the first error reported in the Uncorrectable Error Status Register.

22.7.44 Root Error Command (REC)—Offset 12Ch

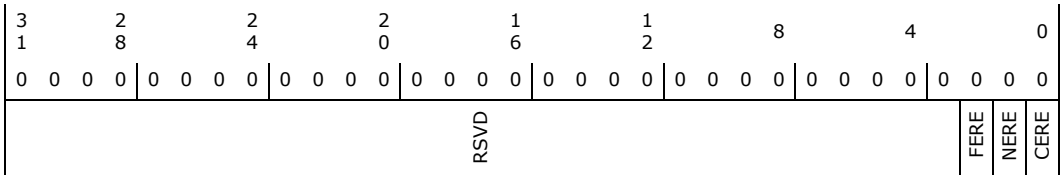
This register allows errors to generate interrupts.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:3	0h RO	Reserved.
2	0h RW	**Fatal Error Reporting Enable (FERE):** When set, the root port will generate an interrupt when a fatal error is reported by the attached device.
1	0h RW	**Non-fatal Error Reporting Enable (NERE):** When set, the root port will generate an interrupt when a non-fatal error is reported by the attached device.
0	0h RW	**Correctable Error Reporting Enable (CERE):** When set, the root port will generate an interrupt when a correctable error is reported by the attached device.

22.7.45 Root Error Status (RES)—Offset 130h

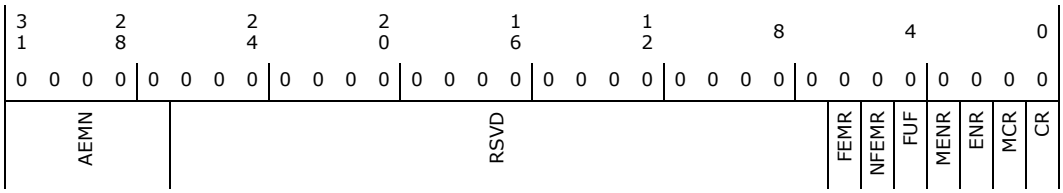
This register can track more than one error and set the 'multiple' bits if a second or subsequent error occurs and the first has not been serviced. This register is only reset by a loss of core power.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:27	0h RO	**Advanced Error Interrupt Message Number (AEMN):** Reserved. There is only one error interrupt allocated.
26:7	0h RO	Reserved.
6	0h RW/1C/V/P	**Fatal Error Message Received (FEMR):** Set when one or more Fatal Uncorrectable Error Messages have been received.
5	0h RW/1C/V/P	**Non-Fatal Error Messages Received (NFEMR):** Set when one or more Non-Fatal Uncorrectable error messages have been received
4	0h RW/1C/V/P	**First Uncorrectable Fatal (FUF):** Set when the first Uncorrectable Error message received is for a fatal error.
3	0h RW/1C/V/P	**Multiple ERR_FATAL/NONFATAL Received (MENR):** Set when either a fatal or a non-fatal error is received and the ENR bit is already set.
2	0h RW/1C/V/P	**ERR_FATAL/NONFATAL Received (ENR):** Set when either a fatal or a non-fatal error message is received.
1	0h RW/1C/V/P	**Multiple ERR_COR Received (MCR):** Set when a correctable error message is received and the CR bit is already set.
0	0h RW/1C/V/P	**ERR_COR Received (CR):** Set when a correctable error message is received.

22.7.46 Error Source Identification (ESID)—Offset 134h

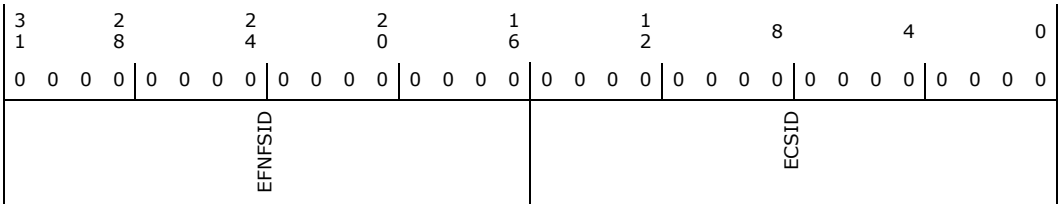
Size:32 bits
Identifies the source (Requester ID) of the first correctable and uncorrectable (Non-Fatal / Fatal) errors reported in the Root Error Status register. This register is only reset by a loss of core power.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO/V/P	**ERR_FATAL/NONFATAL Source Identification (EFNFSID):** Loaded with the Requester ID indicated in the received ERR_FATAL or ERR_NONFATAL Message with the ERR_FATAL/NONFATAL Received register is not already set.
15:0	0h RO/V/P	**ERR_COR Source Identification (ECSID):** Loaded with the Requester ID indicated in the received ERR_COR Message with the ERR_COR Received register is not already set.

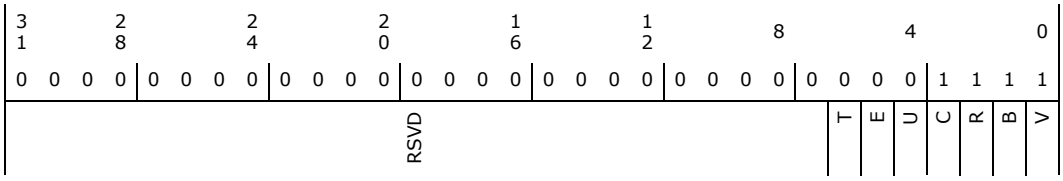
22.7.47 ACS Capability Register (ACSCAPR)—Offset 144h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: Fh



Bit Range	Default and Access	Field Name (ID): Description
31:7	0h RO	Reserved.
6	0h RO	**ACS Direct Translated P2P (T):** ACS Direct Translated P2P (T): ACS Direct Translated P2P is not supported.
5	0h RO	**ACS P2P Egress Control (E):** ACS P2P Egress Control (E): ACS P2P Egress Control is not supported.
4	0h RO	**ACS Upstream Forwarding (U):** ACS Upstream Forwarding (U): ACS Upstream Forwarding is not supported.
3	1h RW/O	**ACS P2P Completion Redirect (C):** ACS P2P Completion Redirect (C): Required for all Functions that support ACS P2P Request Redirect; must be hardwired to 0b otherwise. If 1b, indicates that the component implements ACS P2P Completion Redirect.
2	1h RW/O	**ACS P2P Request Redirect (R):** ACS P2P Request Redirect (R): Required for Root Ports that support peer-to-peer traffic with other Root Ports; required for Switch Downstream Ports; required for multi-function device Functions that support peer-to-peer traffic with other Functions; must be hardwired to 0b otherwise. If 1b, indicates that the component implements ACS P2P Request Redirect.
1	1h RW/O	**ACS Translation Blocking (B):** ACS Translation Blocking (B): Required for Root Ports and Switch Downstream Ports; must be hardwired to 0b otherwise. If 1b, indicates that the component implements ACS Translation Blocking.
0	1h RW/O	**ACS Source Validation (V):** ACS Source Validation (V): Required for Root Ports and Switch Downstream Ports; must be hardwired to 0b otherwise. If 1b, indicates that the component implements ACS Source Validation.

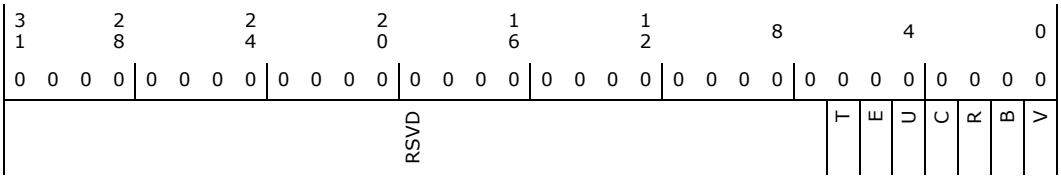
22.7.48 ACS Control Register (ACSCTLR)—Offset 148h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:7	0h RO	Reserved.
6	0h RO	**ACS Direct Translated P2P Enable (T):** ACS Direct Translated P2P Enable (T): ACS Direct Translated P2P is not supported.
5	0h RO	**ACS P2P Egress Control Enable (E):** ACS P2P Egress Control Enable (E): ACS P2P Egress Control is not supported.
4	0h RO	**ACS Upstream Forwarding Enable (U):** ACS Upstream Forwarding Enable (U): ACS Upstream Forwarding is not supported.
3	0h RW	**ACS P2P Completion Redirect (C):** ACS P2P Completion Redirect (C): Determines when the component redirects peer-to-peer Completions upstream; applicable only to Read Completions whose Relaxed Ordering Attribute is clear.
2	0h RW	**ACS P2P Request Redirect (R):** ACS P2P Request Redirect (R): Determines when the component redirects peer-to-peer memory Requests targeting another peer port upstream. I/O, Configuration, VDM Messages and Completions are never affected by ACS P2P Request Redirect.
1	0h RW	**ACS Translation Blocking (B):** ACS Translation Blocking (B): When set, the component blocks all upstream Memory Requests whose Address Translation (AT) field is not set to the default value. I/O, Configuration, Completions and Messages are never affected by ACS Translation Blocking.
0	0h RW	**ACS Source Validation (V):** ACS Source Validation (V): When set, the component validates the Bus Number from the Requester ID of upstream Requests against the secondary / subordinate Bus Numbers. I/O, Configuration and Completions are never affected by ACS Source Validation.

22.7.49 L1 Sub-States Extended Capability Header (L1SECH)—Offset 200h

Size:32 bits.

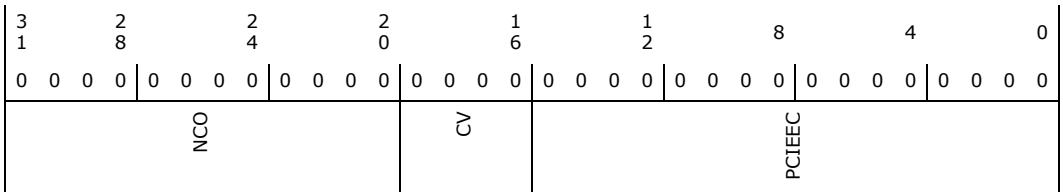
Note: When operating in Mobile Express mode, this capability should not be enabled.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:20	0h RW/O	**Next Capability Offset (NCO):** This field contains the offset to the next PCI Express Capability structure or 000h if no other items exist in the linked list of Capabilities. For Extended Capabilities implemented in Configuration Space, this offset is relative to the beginning of PCI compatible Configuration Space and thus must always be either 000h (for terminating list of Capabilities) or greater than 0FFh. The bottom 2 bits of this offset are Reserved and must be implemented as 00b and software must mask them to allow for future uses of these bits.
19:16	0h RW/O	**Capability Version (CV):** This field is a PCI-SIG defined version number that indicates the version of the Capability structure present. Must be 1h for this version of the specification. For systems that support L1 Sub-State Extended Capability, BIOS should set this field to 1h.
15:0	0h RW/O	**PCI Express Extended Capability ID (PCIEEC):** This field is a PCI-SIG defined ID number that indicates the nature and format of the Extended Capability. For systems that support L1 Sub-State Extended Capability, BIOS should set this field to 001Eh.

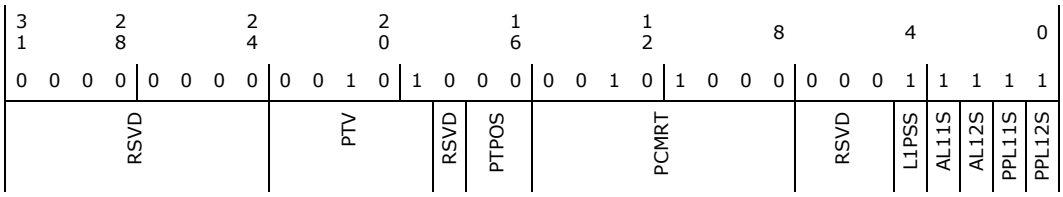
22.7.50 L1 Sub-States Capabilities (L1SCAP)—Offset 204h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 28281Fh



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23:19	5h RW/O	**Port Tpower_on Value (PTV):** Along with the Port T_POWER_ON Scale Field in the L1 Substates Capabilities register sets the Time (in us) that this Port requires the port on the opposite side of Link to wait in L1.OFF_EXIT after sampling CLKREQ# asserted before actively driving the interface. Port Tpower_on is calculated by multiplying the value in this field by the value in the Port Tpower_on scale field in the L1 Sub-States Capabilities 2 register. Required for all Ports that support L1.OFF.
18	0h RO	Reserved.
17:16	0h RW/O	**Port Tpower_on Scale (PTPOS):** Specifies the scale used for Tpower_on value field in the L1 Substates Capabilities register. '00b': 2 us '01b': 10 us '10b': 100 us '11b': Reserved Required for all Ports that support L1.OFF.
15:8	28h RW/O	**Port Common Mode Restore Time (PCMRT):** This is the time (in us) required for this Port to re-establish common mode. Required for all ports that support L1.OFF.
7:5	0h RO	Reserved.
4	1h RW/O	**L1 PM Substates Supported (L1PSS):** When Set this bit indicates that this Port supports L1 PM Substates. For compatibility with possible future extensions, software must not enable L1 PM Substates unless this bit is set. This RWO field must be programmed prior to enabling ASPM.
3	1h RW/O	**ASPM L1.1 Substates Supported (AL11S):** When set, this bit indicates that this port supports L1 substates for ASPM L1.SNOOZ. Required for both Upstream and Downstream ports. This RWO field must be programmed prior to enabling ASPM. Register Attribute: Static
2	1h RW/O	**ASPM L1.2 Supported (AL12S):** When set, this bit indicates that ASPM_L1.OFF is supported. Required for both Upstream and Downstream ports. This RWO field must be programmed prior to enabling ASPM. Register Attribute: Static
1	1h RW/O	**PCI-PM L1.1 Supported (PPL11S):** When set, this bit indicates that L1.SNOOZ sub-state is supported and this bit must be set by all ports implementing L1 Sub-States. A port that supports L1.OFF must support L1.SNOOZ. Required for both upstream and downstream ports. This RWO field must be programmed prior to enabling ASPM. Register Attribute: Static
0	1h RW/O	**PCI-PM L1.2 Supported (PPL12S):** When set, this bit indicates that L1.OFF power management feature is supported. Required for both upstream and downstream ports. This RWO field must be programmed prior to enabling ASPM. Register Attribute: Static

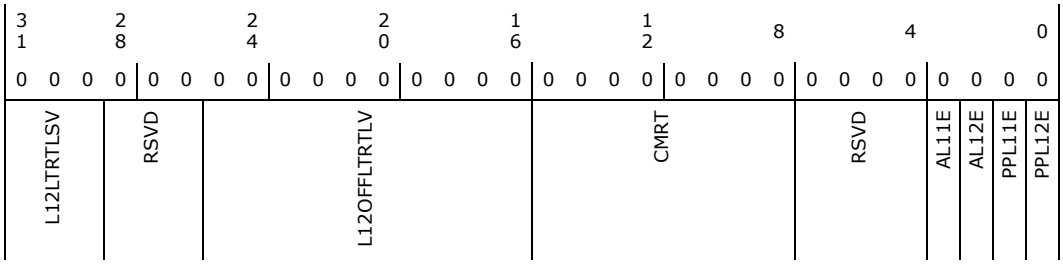
22.7.51 L1 Sub-States Control 1 (L1SCTL1)—Offset 208h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:29	0h RW	**L1.2 LTR Threshold Latency ScaleValue (L12LTRTLSV):** This field contains the L1.OFF LTR Threshold Latency Scale Value for this particular PCIe root port. The value in this field, together with L12LTRTLV is compared against both the snoop and non-snoop LTR values of the device. 000: L12LTRSTLV times 1 ns 001: L12LTRSTLV times 32 ns 010: L12LTRSTLV times 1024 ns 011: L12LTRSTLV times 32768 ns 100: L12LTRSTLV times 1048576 ns 101: L12LTRSTLV times 33554432 ns Others: Not Permitted. This field must be programmed prior to enabling L1.OFF. Register Attribute: Static
28:26	0h RO	Reserved.
25:16	0h RW	**L1.2 LTR Threshold Latency Value (L12OFFLTRTLV):** This field contains the L1.2 LTR Threshold Latency Value for this particular PCIe root port. The value in this field, together with L12LTRTLSV is compared against both the snoop and non-snoop LTR values of the device. This field must be programmed prior to enabling L1.OFF. Register Attribute: Static
15:8	0h RW	**Common Mode Restore Time (CMRT):** This is the Tcommon_mode time the PCIe root port needs to continue sending TS1 and refrain from sending TS2 in Recovery state to allow the TX common mode to be established prior to sending TS2. The timer starts from the time when the first TS1 has been sent and the receiver has detected un-squelch. The value in this field defines the time in micro-seconds. This field must be programmed prior to enabling L1.OFF. Register Attribute: Static
7:4	0h RO	Reserved.
3	0h RW	**ASPM L1.1 Enabled (AL11E):** When set, this bit indicates that ASPM L1.SNOOZ substates are enabled for ASPM. Required for both upstream and downstream ports. Register Attribute: Dynamic

Bit Range	Default and Access	Field Name (ID): Description
2	0h RW	**ASPM L1.2 Enable (AL12E):** When set, this bit indicates that ASPM L1.OFF substates are enabled for PCI-PM. Required for both upstream and downstream ports. Register Attribute: Dynamic
1	0h RW	**PCI-PM L1.SNOOZ Enable (PPL11E):** When set, this bit indicates that PCI-PM L1.SNOOZ power management feature is enabled. If L1.OFF is enabled, L1.SNOOZ must also be enabled. This field must be programmed prior to enabling ASPM L1. Register Attribute: Dynamic ***Note:*** If STRPFUSECFG.mPHYIOPMDIS is '1', hardware will always see '0' as an output from this register. BIOS reading this register should always return the correct value.
0	0h RW	**PCI-PM L1.2 Enabled (PPL12E):** When set, this bit indicates that PCI-PM L1.OFF power management feature is enabled. L1.OFF can only be enabled if the platform supports bi-directional CLKREQPLUS#. This field must be programmed prior to enabling ASPM L1. Ports that support L1.OFF shall support Latency Tolerance Reporting. Register Attribute: Dynamic ***Note:*** If STRPFUSECFG.mPHYIOPMDIS is '1', hardware will always see '0' as an output from this register. BIOS reading this register should always return the correct value.

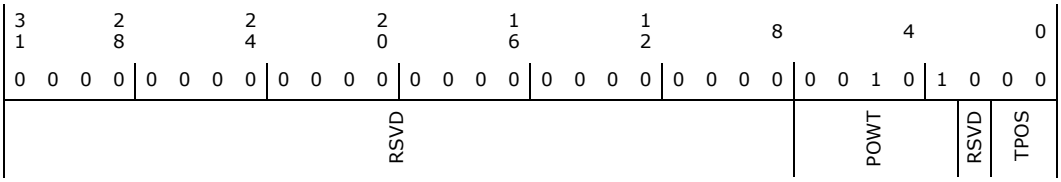
22.7.52 L1 Sub-States Control 2 (L1SCTL2)—Offset 20Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 28h



Bit Range	Default and Access	Field Name (ID): Description
31:8	0h RO	Reserved.
7:3	5h RW	**Power On Wait Time (POWT):** Along with the Tpower_on Scale sets the minimum amount of time (in us) that the Port must wait in L1.OFF EXIT after sampling CLKREQPLUS# asserted before actively driving the interface. The timer starts counting when CLKREQPLUS# is sampled asserts in L1.OFF state. Tpower_on value is calculated by multiplying the value in this field by the value in the TPOS field. This field must be programmed prior to enabling L1.OFF. Register Attribute: Static
2	0h RO	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
1:0	0h RW	**Tpower_on Scale (TPOS):** Specifies the scale used for Tpower_on value. '00b': 2 us '01b': 10 us '10b': 100us '11b': Reserved. Required for all Ports that support L1.OFF. Register Attribute: Static

22.7.53 Secondary PCI Express Extended Capability Header (SPEECH)—Offset 220h

Size:32 bits.

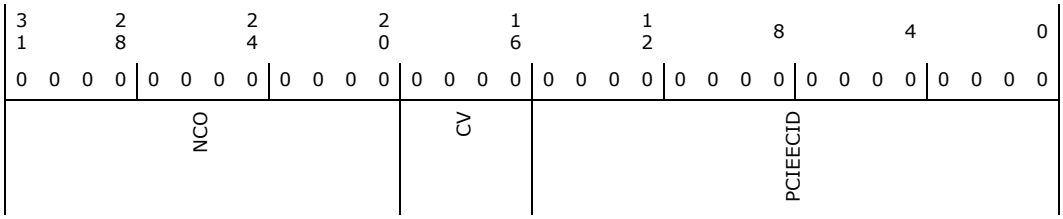
Note: When operating in Mobile Express mode, this capability should not be enabled.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:20	0h RW/O	**Next Capability Offset (NCO):** Next Capability Offset (NCO): This field contains the offset to the next PCI Express Capability structure or 000h if no other items exist in the linked list of Capabilities. For Extended Capabilities implemented in Configuration Space, this offset is relative to the beginning of PCI compatible Configuration Space and thus must always be either 000h (for terminating list of Capabilities) or greater than 0FFh. The bottom 2 bits of this offset are Reserved and must be implemented as 00b and software must mask them to allow for future uses of these bits.
19:16	0h RW/O	**Capability Version (CV):** Capability Version (CV): This field is a PCI-SIG defined version number that indicates the version of the Capability structure present. For systems that support Secondary PCI Express Extended Capability, BIOS should write a 1h to this register else it should write 0
15:0	0h RW/O	**PCI Express Extended Capability ID (PCIEECID):** PCI Express Extended Capability ID (PCIEECID): This field is a PCI-SIG defined ID number that indicates the nature and format of the Extended Capability. PCI Express Extended Capability ID for the Secondary PCI Express Extended Capability is 0019h. For systems that support Secondary PCI Express Extended Capability, BIOS should write a 0019h to this register else it should write 0.

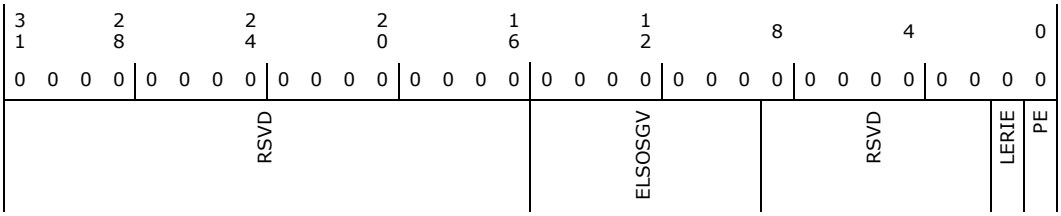
22.7.54 Link Control 3 (LCTL3)—Offset 224h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:9	0h RO	**Enable Lower SKP OS Generation Vector (ELSOSGV):** Enable Lower SKP OS Generation Vector(ELSOSGV): When the Link is in L0 and the bit in this field corresponding to the current Link speed is Set, SKP Ordered Sets are scheduled at the rate defined for SRNS, overriding the rate required based on the clock tolerance architecture. Bit definitions within this field are: Bit 0 2.5 GT/s Bit 1 5.0 GT/s Bit 2 8.0 GT/s Bits 6:3 RsvdP Behavior is undefined if a bit is Set in this field and the corresponding bit in the Lower SKP OS Generation Supported Speeds Vector is not set.
8:2	0h RO	Reserved.
1	0h RW	**Link Equalization Request Interrupt Enable (LERIE):** Link Equalization Request Interrupt Enable (LERIE): When set, this bit enables the generation of an interrupt to indicate that the Link Equalization Request bit has been set.
0	0h RW	**Perform Equalization (PE):** Perform Equalization (PE): When this bit is 1b and Link Retrain bit is set with the Target Link Speed field set to 8 GT/s, the Downstream Port must perform Link Equalization. This bit is cleared by Root Port upon entry to Link Equalization

22.7.55 Lane 0 and Lane 1 Equalization Control (L01EC)—Offset 22Ch

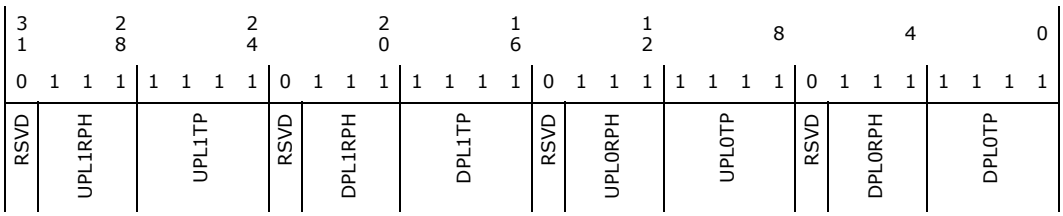
Each entry contains the values for the Lane with the corresponding default Lane number which is invariant to Link width and Lane reversal negotiation that occurs during Link training.
Based on the Port Configuration setting, if the maximum link width of the port is less than x4, the upper lane status bits should be ignored.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 7F7F7F7Fh



Bit Range	Default and Access	Field Name (ID): Description
31	0h RO	Reserved.
30:28	7h RW	**Upstream Port Lane 1 Receiver Preset Hint (UPL1RPH):** Upstream Port Lane 1 Receiver Preset Hint (UPL1RPH): Field contains the Receiver Preset Hint value sent or received during Link Equalization.
27:24	Fh RW	**Upstream Port Lane 1 Transmitter Preset (UPL1TP):** Upstream Port Lane 1 Transmitter Preset (UPL1TP): Field contains the Transmit Preset value sent or received during Link Equalization.
23	0h RO	Reserved.
22:20	7h RW	**Downstream Port Lane 1 Receiver Preset Hint (DPL1RPH):** Downstream Port Lane 1 Receiver Preset Hint (DPL1RPH): Receiver Preset Hint may be used as a hint for receiver equalization by this Port when the Port is operating as a Downstream Port.
19:16	Fh RW	**Downstream Port Lane 1 Transmitter Preset (DPL1TP):** Downstream Port Lane 1 Transmitter Preset (DPL1TP): Transmitter Preset used for equalization by this Port when the Port is operating as a Downstream Port. This field is ignored when the Port is operating as an Upstream Port.
15	0h RO	Reserved.
14:12	7h RW	**Upstream Port Lane 0 Receiver Preset Hint (UPL0RPH):** Upstream Port Lane 0 Receiver Preset Hint (UPL0RPH): Field contains the Receiver Preset Hint value sent or received during Link Equalization.
11:8	Fh RW	**Upstream Port Lane 0 Transmitter Preset (UPL0TP):** Upstream Port Lane 0 Transmitter Preset (UPL0TP): Field contains the Transmit Preset value sent or received during Link Equalization.
7	0h RO	Reserved.
6:4	7h RW	**Downstream Port Lane 0 Receiver Preset Hint (DPL0RPH):** Downstream Port Lane 0 Receiver Preset Hint (DPL0RPH): Receiver Preset Hint may be used as a hint for receiver equalization by this Port when the Port is operating as a Downstream Port.
3:0	Fh RW	**Downstream Port Lane 0 Transmitter Preset (DPL0TP):** Downstream Port Lane 0 Transmitter Preset (DPL0TP): Transmitter Preset used for equalization by this Port when the Port is operating as a Downstream Port. This field is ignored when the Port is operating as an Upstream Port.

22.7.56 Lane 2 and Lane 3 Equalization Control (L23EC)—Offset 230h

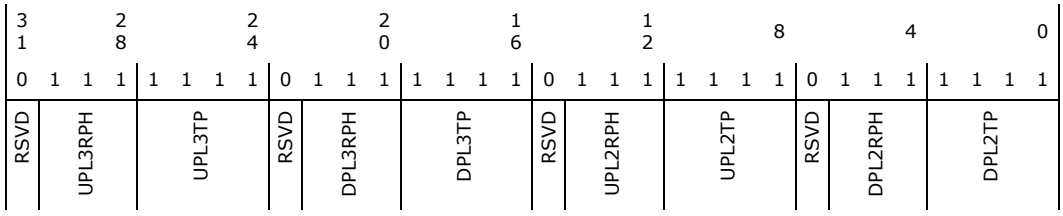
Each entry contains the values for the Lane with the corresponding default Lane number which is invariant to Link width and Lane reversal negotiation that occurs during Link training.
Based on the Port Configuration setting, if the maximum link width of the port is less than x4, the upper lane status bits should be ignored.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 7F7F7F7Fh



Bit Range	Default and Access	Field Name (ID): Description
31	0h RO	Reserved.
30:28	7h RW	**Upstream Port Lane 3 Receiver Preset Hint (UPL3RPH):** Upstream Port Lane 3 Receiver Preset Hint (UPL3RPH): Field contains the Receiver Preset Hint value sent or received during Link Equalization.
27:24	Fh RW	**Upstream Port Lane 3 Transmitter Preset (UPL3TP):** Upstream Port Lane 3 Transmitter Preset (UPL3TP): Field contains the Transmit Preset value sent or received during Link Equalization.
23	0h RO	Reserved.
22:20	7h RW	**Downstream Port Lane 3 Receiver Preset Hint (DPL3RPH):** Downstream Port Lane 3 Receiver Preset Hint (DPL3RPH): Receiver Preset Hint may be used as a hint for receiver equalization by this Port when the Port is operating as a Downstream Port.
19:16	Fh RW	**Downstream Port Lane 3 Transmitter Preset (DPL3TP):** Downstream Port Lane 3 Transmitter Preset (DPL3TP): Transmitter Preset used for equalization by this Port when the Port is operating as a Downstream Port. This field is ignored when the Port is operating as an Upstream Port.
15	0h RO	Reserved.
14:12	7h RW	**Upstream Port Lane 2 Receiver Preset Hint (UPL2RPH):** Upstream Port Lane 2 Receiver Preset Hint (UPL2RPH): Field contains the Receiver Preset Hint value sent or received during Link Equalization.
11:8	Fh RW	**Upstream Port Lane 2 Transmitter Preset (UPL2TP):** Upstream Port Lane 2 Transmitter Preset (UPL2TP): Field contains the Transmit Preset value sent or received during Link Equalization.
7	0h RO	Reserved.
6:4	7h RW	**Downstream Port Lane 2 Receiver Preset Hint (DPL2RPH):** Downstream Port Lane 2 Receiver Preset Hint (DPL2RPH): Receiver Preset Hint may be used as a hint for receiver equalization by this Port when the Port is operating as a Downstream Port.
3:0	Fh RW	**Downstream Port Lane 2 Transmitter Preset (DPL2TP):** Downstream Port Lane 2 Transmitter Preset (DPL2TP): Transmitter Preset used for equalization by this Port when the Port is operating as a Downstream Port. This field is ignored when the Port is operating as an Upstream Port.

22.8 PCI Express NP16 Uplink Port Configuration Registers Register Summary

The NP16 Uplink is a single endpoint PCIe controller. This controller is configurable into being x16, x8, x4 or x1. However, it is not capable of being configured into two endpoints (2x8 for example)

Table 22-9. Summary of PCI Express Port Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	Identifiers (ID)—Offset 0h	37408086h
4h	7h	Device Command; Primary Status (CMD_PSTS)—Offset 4h	00100000h
8h	Bh	Revision ID;Class Code (RID_CC)—Offset 8h	06040000h
Ch	Fh	Cache Line Size; Primary Latency Timer; Header Type (CLS_PLT_HTYPE)—Offset Ch	0001000h
10h	17h	Express Port memory Base Address Register (EXPPTMBAR_UX16) - Offset 10h	000000000 0000004H
18h	1Bh	Bus Numbers; Secondary Latency Timer (BNUM_SLT)—Offset 18h	00000000h
1Ch	1Fh	I/O Base and Limit; Secondary Status (IOBL_SSTS)—Offset 1Ch	00000000h
20h	23h	Memory Base and Limit (MBL)—Offset 20h	00000000h
24h	27h	Prefetchable Memory Base and Limit (PMBL)—Offset 24h	00010001h
28h	2Bh	Prefetchable Memory Base Upper 32 Bits (PMBU32)—Offset 28h	00000000h
2Ch	2Fh	Prefetchable Memory Limit Upper 32 Bits (PMLU32)—Offset 2Ch	00000000h
34h	37h	Capabilities List Pointer (CAPP)—Offset 34h	00000040h
38h	3Bh	Expansion ROM Base Address Register (EXPROM) - Offset 38h	00000000
3Ch	3Fh	Interrupt Information; Bridge Control (INTR_BCTRL)—Offset 3Ch	00000100h
40h	43h	Capabilities List; PCI Express Capabilities (CLIST_XCAP)—Offset 40h	00528010h
44h	47h	Device Capabilities (DCAP)—Offset 44h	00008002h
48h	4Bh	Device Control; Device Status (DCTL_DSTS)—Offset 48h	00002000h
4Ch	4Fh	Link Capabilities (LCAP)—Offset 4Ch	0041C903h
50h	53h	Link Control; Link Status (LCTL_LSTS)—Offset 50h	00010000h
64h	67h	Device Capabilities 2 (DCAP2)—Offset 64h	00000000h
68h	6Bh	Device Control 2; Device Status 2 (DCTL2_DSTS2)—Offset 68h	00000000h
6Ch	6Fh	Link Capabilities 2 (LCAP2)—Offset 6Ch	0000000Eh
70h	73h	Link Control 2; Link Status 2 (LCTL2_LSTS2)—Offset 70h	00000003h
80h	83h	Power Management Capability, PCI Power Management Capabilities (PMCAP_PMC) - Offset 80h	C8038801h
84h	87h	PCI Power Management Control and Status (PMCS) - Offset 84h	00000008h
88h	8Bh	Subsystem Capability List Register (SVCAP) - Offset 88h	0000000D
8C	8F	Subsystem Vendor ID (SVID) - Offset 8C	00008086
EA	EF	Personality Lock Key Control Register (PLKCTL) - Offset EAh	00000000
100h	103h	Advanced Error Extended Reporting Capability Header (AECH)—Offset 100h	20010001h
104h	107h	Uncorrectable Error Status (UES)—Offset 104h	00000000h
108h	10Bh	Uncorrectable Error Mask (UEM)—Offset 108h	00200000h
10Ch	10Fh	Uncorrectable Error Severity (UEV)—Offset 10Ch	00262010h
110h	113h	Correctable Error Status (CES)—Offset 110h	00000000h

Table 22-9. Summary of PCI Express Port Configuration Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
114h	117h	Correctable Error Mask (CEM)—Offset 114h	0000E000h
118h	11Bh	Advanced Error Capabilities and Control (AECC)—Offset 118h	00000000h
11Ch	128h	Header Log Register (AERHDRLOG[1-4]) - Offset 11c - 128h	00000000h
140h	143h	Uncorrectable Error Detect Mask Register (ERRUNCDETMSK) - Offset 140h	04000000
144h	147h	Correctable Error Detect Mask Register (ERRCORDETMSK) - Offset 144h	00000000
200h	203h	Secondary PCI Express Extended Capability Header (SPEECH)—Offset 200h	00010019h
204h	207h	Link Control 3 (LCTL3)—Offset 204h	00000000h
208h	20Bh	Lane Error Status Register (LANEERRSTS) - Offset 208	00000000
20Ch	22Ah	Lane Equalization Control Register (LANEEQCTL[0-15]) - Offset 20C - 22A by 2 (for up to 16 upstream lanes	7F00h

22.8.1 Identifiers (ID)—Offset 0h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 37C08086h

31			28				24				20				16				12				8				4				0
0	0	1	1	0	1	1	1	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0
DID																VID															

Bit Range	Default and Access	Field Name (ID): Description
31:16	37C0 RO/V	**Device Identification (DID):** Identifies this particular function.
15:0	8086h RO	**Vendor Identification (VID):** Indicates Intel.

22.8.2 Device Command; Primary Status (CMD_PSTS)—Offset 4h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00100000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DPE	SSE	RMA	RTA	STA	PDTS		DPD	PFBC	RSVD	PC66	CLIST	IS	RSVD								ID	FBE	SEE	WCC	PERE	VGA_PSE	MWIE	SCE	BME	MSE	IOSE

Bit Range	Default and Access	Field Name (ID): Description
31	0b RW1C/V	**DPE - Detected Parity Error (DPE):** Set when the port receives a poisoned tlp from the PCIe interface of the internal backboner. This is set even if CMD.PERE is not set.
30	0b RW1C/V	**Signaled System Error (SSE):** This bit is set when ERR_FATAL or ERR_NONFATAL messages are sent to the root complex and the SERR enable bit in the PCICMD Register is set
29	0b RW1C/V	**Received Master Abort (RMA):** Set when the requester receives a completion with unsupported request status.
28	0b RW1C/V	**Received Target Abort (RTA):** Set when the root port receives a completion with completer abort.
27	0b RW1C/V	**Signaled Target Abort (STA):** This bit is Set when the port completes a Posted or Non-Posted Request as a Completer Abort error. This applies to a Function with a Type 1 Configuration header when the Completer Abort was generated by its Primary Side.
26:25	0b RO	**Primary DEVSEL# Timing Status (PDTS):** Reserved per *PCI Express Specification*.
24	0b RW1C/V	**Master Data Parity Error Detected (DPD):** This bit is set by a requester (primary side for type1 configuration headerfunctions) if the parity error response enable bit (PERE) in the Command Register is set and either of the following two conditions occur: • Requester receives a completion marked poisoned. • Requester sends a poisoned request (includes writes and messages) If the parity error bit is 0b, this bit is never set.
23	0b RO	**Primary Fast Back to Back Capable (PFBC):** Reserved per *PCI Express Specification*.
22	0b RV	**Reserved**.
21	0b RO	**Primary 66 MHz Capable (PC66):** Reserved per *PCI Express Specification*.
20	1b RO	**Capabilities List (CLIST):** Indicates the presence of a capabilities list.
19	0b RO/V	**Interrupt Status (IS):** When set indicates that an INTx emulation interrupt is pending internally for this function.
18:11	0h RV	**Reserved**.

Bit Range	Default and Access	Field Name (ID): Description
10	0b RW	**Interrupt Disable (ID):** This disables pin-based INTx# interrupts on enabled hot plug and power management events. This bit has no effect on MSI operation. When set, internal INTx# messages will not be generated. When cleared, internal INTx# messages are generated if there is an interrupt for hot plug or power management and MSI is not enabled. This bit does not affect interrupt forwarding from devices connected to the root port. Assert_INTx and Deassert_INTx messages will still be forwarded to the internal interrupt controllers if this bit is set. For PCI Bus Emulation Mode compatibility, if the PCIBEM register is set, this register is RO and returns a value of 0 when read, else it is RW with the functionality described above.
9	0b RO	**Fast Back to Back Enable (FBE):** Reserved per *PCI Express Specification*.
8	0b RW	**SERR# Enable (SEE):** When set, enables the root port to generate an SERR# message when PSTS.SSE is set.
7	0b RO	**Wait Cycle Control (WCC):** Reserved per *PCI Express Specification*.
6	0b RW	**Parity Error Response Enable (PERE):** Indicates that the device is capable of reporting parity errors as a master on the backbone.
5	0b RO	**VGA Palette Snoop (VGA_PSE):** Reserved per *PCI Express Specification*.
4	0b RO	**Memory Write and Invalidate Enable (MWIE):** Reserved per *PCI Express Specification*.
3	0b RO	**Special Cycle Enable (SCE):** Reserved per *PCI Express Specification* and *PCI Bridge Specification*.
2	0b RW	**Bus Master Enable (BME):.**This bit controls the ability of the Function to issue Memory and I/O read or write requests, and the ability of Root or Switch port to forward memory and I/O read or write requests in the upstream direction. When this bit is 0b, memory and I/O requests received at the root port or downstream side of a switch port (secondary side) must be handled as an Unsupported Request (UR). For Non-posted requests, a completion with UR completion status must be returned. For an endpoint, when this bit is Set, the PCI Express Function is allowed to issue Memory or I/O Requests. When this bit is Clear, the PCI Express Function is not allowed to issue any Memory or I/O Requests.
1	0b RW	**Memory Space Enable (MSE):** When set, memory cycles within the range specified by the memory base and limit registers can be forwarded to the PCI-Express device. When cleared, these memory cycles are responded to on the primary interface with an unsupported request completion.
0	0b RW	**I/O Space Enable (IOSE):** When set, I/O cycles within the range specified by the I/O base and limit registers can be forwarded to the PCI-Express device. When cleared, these cycles are responded to on the primary interface with an unsupported request completion.

22.8.3 Revision ID;Class Code (RID_CC)—Offset 8h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 06040000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
BCC								SCC								PI								RID								

Bit Range	Default and Access	Field Name (ID): Description
31:24	06h RW/L	**Base Class Code (BCC):** Indicates the device is a bridge device.
23:16	04h RW/L	**Sub-Class Code (SCC):** The default indicates the device is a PCI-to-PCI bridge. If the MPC.BT register is set to a '1' for a Host Bridge, this register reads 00h.
15:8	0h RW/L	**Programming Interface (PI):** This bit indicates that this device is standard (non-subtractive) PCI-to-PCI Bridge.
7:0	00h RO/V	**Revision ID (RID):** Indicates the revision of the function.

22.8.4 Cache Line Size; Primary Latency Timer; Header Type (CLS_PLT_HTYPE)—Offset Ch

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00010000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BIST_TST								MFD	HTYPE							CT								LS								

Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	**BIST tests (BIST_TST):** Not supported, hard wired to 00h
23	0b RW/L	**Multi-function Device (MFD):** This bit is '0' to indicate a single function device.

Bit Range	Default and Access	Field Name (ID): Description
22:16	01h RW/L	**Header Type (HTYPE):** The default mode identifies the header layout of the configuration space, which is a PCI-to-PCI bridge. If the MPC.BT register is set to a '1' for a Host Bridge, this register reads 00h.
15:8	00h RO	**Latency Count (CT):** Reserved per *PCI Express Specification*.
7:0	00h RW	**Line Size (LS):** This is read/write but contains no functionality, per *PCI Express Specification*.

22.8.5 Express Port Memory Base Address Register (EXPPTMBAR_UX16) — Offset 10h

Access Method

Type: CFG Register
(Size: 64 bits)

Device: 0
Function: 0

Default: 0000000000000004h

63			60				56				52				48				44				40				36				32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BADDR																															

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
BADDR															RSVD														PREF	TYPE		MIS

Bit Range	Default and Access	Field Name (ID): Description
63:17	00000000 0000h RW	**Base Address (BADDR):**
16:4	000h Rv	**Reserved**
3	0b RO	**Prefetchable Memory (Pref)**: Not prefetchable memory space.
2:1	10b RO	**Type (type):** The memory mapped space can be placed anywhere in the 64-bit addressable region of the device
0	0b RO	**Memory Space Indicator (MSI):** 0b = Memory space, 1b - I/O space. Indicates the bus number of the backbone.

22.8.6 Bus Numbers (BusNUM)—Offset 18h

Type: CFG Register
(Size: 24bits)

Device: 0
Function: 0

Default: 0h

			20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SBBN								SCBN								PBN							

Bit Range	Default and Access	Field Name (ID): Description
23:16	0h RW	**Subordinate Bus Number (SBBN):** Indicates the highest PCI bus number below the bridge.
15:8	0h RW	**Secondary Bus Number (SCBN):** Indicates the bus number secondary interface.
7:0	0h RW	**Primary Bus Number (PBN):** Indicates the PCI Express bus number.

22.8.7 I/O Base and Limit; Secondary Status (IOBL_SSTS)—Offset 1Ch

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DPE	RSE	RMA	RTA	STA	SDTS		MDPD	FBC	RSVD	SC66	RSVD				IOLA				IOLC				IOBA				IOBC				

Bit Range	Default and Access	Field Name (ID): Description
31	0b RW1C/V	**Detected Parity Error (DPE):** Set when the port receives a poisoned TLP.
30	0b RW1C/V	**Received System Error (RSE):** Set when the port receives an ERR_FATAL or ERR_NONFATAL message from the device.
29	0b RW1C/V	**Received Master Abort (RMA):** Set when the port receives a completion with 'Unsupported Request' status from the device.
28	0b RW1C/V	**Received Target Abort (RTA):** Set when the port receives a completion with 'Completion Abort' status from the device.
27	0b RW1C/V	**Signaled Target Abort (STA):** Set when the port generates a completion with 'Completion Abort' status to the device.

Bit Range	Default and Access	Field Name (ID): Description
26:25	0h RO	**DEVSEL# Timing Status (DVT):** Reserved per *PCI Express Specification*.
24	0b RW1C/V	**Master Data Parity Error Detected (MDPD):** Set when the BCTRL.PERE, and either of the following two conditions occurs: Port receives completion marked poisoned. Port poisons a write request to the secondary side.
23	0h RO	**Fast Back to Back Capable (FBC):** Reserved per *PCI Express Specification*.
22	0b RV	**Reserved**.
21	0b RO	**Secondary 66 MHz Capable (SC66):** Reserved per *PCI Express Specification*.
20:16	0h RV	Reserved.
15:12	0h RW	**I/O Address Limit (IOLA):** I/O Base bits corresponding to address lines 15:12 for 4KB alignment. Bits 11:0 are assumed to be padded to FFFh.
11:8	0h RO	**I/O Limit Address Capability (IOLC):** Indicates that the bridge does not support 32-bit I/O addressing.
7:4	0h RW	**I/O Base Address (IOBA):** I/O Base bits corresponding to address lines 15:12 for 4KB alignment. Bits 11:0 are assumed to be padded to 000h.
3:0	0h RO	**I/O Base Address Capability (IOBC):** Indicates that the bridge does not support 32-bit I/O addressing.

22.8.8 Memory Base and Limit (MBL)—Offset 20h

Accesses that are within the ranges specified in this register will be sent to the attached device if CMD.MSE is set. Accesses from the attached device that are outside the ranges specified will be forwarded to the backbone if CMD.BME is set. The comparison performed is MB [gt]= AD[lb]31:20[rb] [lt]= ML.

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ML												RSVD				MB												RSVD				

Bit Range	Default and Access	Field Name (ID): Description
31:20	0h RW	**Memory Limit (ML):** These bits are compared with bits [31:20] of the incoming address to determine the upper 1MB aligned value of the range.

Bit Range	Default and Access	Field Name (ID): Description
19:16	0h RV	**Reserved**.
15:4	0h RW	**Memory Base (MB):** These bits are compared with bits [31:20] of the incoming address to determine the lower 1MB aligned value of the range.
3:0	0h RV	**Reserved**.

22.8.9 Prefetchable Memory Base and Limit (PMBL)—Offset 24h

Accesses that are within the ranges specified in this register will be sent to the device if CMD.MSE is set. Accesses from the device that are outside the ranges specified will be forwarded to the backbone if CMD.BME is set. The comparison performed is PMBU32:PMB [gt]= AD[lb]63:32[rb]:AD[lb]31:20[rb] [lt]= PMLU32:PML.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00010001h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
PML													I64L				PMB												I64B			

Bit Range	Default and Access	Field Name (ID): Description
31:20	000h RW	**Prefetchable Memory Limit (PML):** These bits are compared with bits [31:20] of the incoming address to determine the upper 1MB aligned value of the range.
19:16	1h RO	**64-bit Indicator (I64L):** Indicates support for 64-bit addressing.
15:4	000h RW	**Prefetchable Memory Base (PMB):** These bits are compared with bits [31:20] of the incoming address to determine the lower 1MB aligned value of the range.
3:0	1h RO	**64-bit Indicator (I64B):** Indicates support for 64-bit addressing.

22.8.10 Prefetchable Memory Base Upper 32 Bits (PMBU32)—Offset 28h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PMBU																															

Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Prefetchable Memory Base Upper Portion (PMBU):** Upper 32-bits of the prefetchable address base.

22.8.11 Prefetchable Memory Limit Upper 32 Bits (PMLU32)—Offset 2Ch

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PMLU																															

Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Prefetchable Memory Limit Upper Portion (PMLU):** Upper 32-bits of the prefetchable address limit.

22.8.12 Capabilities List Pointer (CAPP)—Offset 34h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000040h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
RSVD																								PTR							

Bit Range	Default and Access	Field Name (ID): Description
31:8	0h RO	Reserved.
7:0	40h RW/O	**Capabilities Pointer (PTR):** Indicates that the pointer for the first entry in the capabilities list. BIOS can determine which capabilities will be exposed by including or removing them from the capability linked list.

22.8.13 Expansion ROM Base Address Register (EXPROM)—Offset 38h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BADDR												RSVD																			EXPROMEN

Bit Range	Default and Access	Field Name (ID): Description
31:20	0h RW	**Base Address (BADDR)**: This marks the base address for the 1MB expansion ROM for the endpoint device. Firmware should initialize this region prior to PCIe enumeration.
19:1	0h RV	Reserved
0	0b RW/v	**Expansion ROM Enable (EXPROMEN):** This field enables the expansion ROM base address to be used by the device. 0: Disables access to Expansion ROM 1: Enables Access to Expansion ROM.

22.8.14 Interrupt Information; Bridge Control (INTR_BCTRL)—Offset 3Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000100h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
RSVD				DTSE	DTS	SDT	PDT	FBE	SBR	MAM	V16	VE	IE	SE	PERE	IPIN								ILINE								

Bit Range	Default and Access	Field Name (ID): Description
31:28	0h RV	**Reserved**.
27	0b RO	**Discard Timer SERR# Enable (DTSE):** Reserved per *PCI Express Specification*.
26	0b RO	**Discard Timer Status (DTS):** Reserved per *PCI Express Specification*.
25	0b RO	**Secondary Discard Timer (SDT):** Reserved per *PCI Express Specification*.
24	0b RO	**Primary Discard Timer (PDT):** Reserved per *PCI Express Specification*.
23	0b RO	**Fast Back to Back Enable (FBE):** Reserved per *PCI Express Specification*.
22	0b RW	**Secondary Bus Reset (SBR):** Triggers a hot reset on the PCI-Express port.
21	0b RO	**Master Abort Mode (MAM):** Reserved per *PCI Express Specification*.
20	0b RW	**VGA 16-Bit Decode (V16):** When set, indicates that the I/O aliases of the VGA range (see BCTRL:VE definition below), are not enabled. 0: Execute 10-bit address decode on VGA I/O accesses. 1: Execute 16-bit address decode on VGA I/O accesses.
19	0b RW	**VGA Enable (VE):** When set to 1b, the following ranges will be forwarded from the primary to the secondary side of the bridge regardless of the value of the I/O base and limit registers. Memory ranges A0000h-BFFFFh I/O ranges 3B0h 3BBh and 3C0h 3DFh, and all aliases of bits [15:10] in any combination of 1's
18	0b RW	**ISA Enable (IE):** This bit only applies to I/O addresses that are enabled by the I/O Base and I/O Limit registers and are in the first 64 KB of PCI I/O space. If this bit is set, the PCIe bridge will block any forwarding from the primary to the secondary sided I/O transactions addressing the last 768 bytes in each 1KB block (offsets 100h to 3FFh). In the opposite direction (secondary to primary) I/O transactions will be forwarded if they address the last 768B in each 1 KB block. 1: Forward upstream ISA I/O addresses in the address range defined by the I/O Base and I/O Limit registers that are in the firsts 64 KB of PCI I/O address space (Top 768B of each 1K block). 0: Forward downstream all I/O addresses in the address range defined by the I/O Base and I/O Limit registers.

Bit Range	Default and Access	Field Name (ID): Description
17	0b RW	SERR# Enable (SE): This bit controls the forwarding of ERR_COR, ERR_NONFATAL AND ERR_FATAL messages to the primary side of the PCIe port. 1: Enables forwarding of ERR_COR, ERR_NONFATA, ERR_FATAL messages. 0: Disables forwarding of ERR_COR, ERR_NONFATA, ERR_FATAL messages.
16	0b RW	Parity Error Response Enable (PERE): This bit controls the response to poisoned TLPs in the PCI Express port. 1: Enables reporting of poisoned TLP errors. 0: Disables reporting of poisoned TLP errors.
15:8	01h RW/L	**Interrupt Pin (IPIN):** This register tells which interrupt pin the function uses. 01h: Generate INTA 02h: Generate INTB 03h: Generate INTC 04h: Generate INTD Others: Reserved BIOS has the ability to write this register during boot to setup the correct interrupt for the function. Once the bits are locked they can not be updated until another PLTRST#
7:0	0h RW	**Interrupt Line (ILINE):** Software written value to indicate which interrupt line (vector) the interrupt is connected to. No hardware action is taken on this register.

22.8.15 Capabilities List; PCI Express Capabilities (CLIST_XCAP)—Offset 40h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00528010h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
RSVD			IMN				SI	DT				CV				NEXT								CID								

Bit Range	Default and Access	Field Name (ID): Description
31:30	0h RV	**Reserved.**
29:25	0h RO	**Interrupt Message Number (IMN):** This field indicates the interrupt message number that is generated from the PCI Express port. When there is more than one MSI interrupt number, this register is required to contain the offset between the base Message Data and the MSI Message that is generated when the status bits in the slot status register or root port status registers are set. The chipset is required to update this field if the number of MSI messages change
24	0b RO	**Slot Implemented (SI):** Hardwired to 0b.
23:20	5h RO	**Device / Port Type (DT):** Indicates this the upstream port of a PCIe switch.

Bit Range	Default and Access	Field Name (ID): Description
19:16	2h RO	**Capability Version (CV):** Version 2.0 indicates devices compliant to the *PCI Express 2.0 Specification* which incorporates the Register Expansion ECN.
15:8	80h RO	**Next Capability (NEXT):** Indicates the location of the next capability.
7:0	10h RO	**Capability ID (CID):** Indicates this is a PCI Express capability.

22.8.16 Device Capabilities (DCAP)—Offset 44h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00008002h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
RSVD			FLRC	CSPS			CSPV								RSVD	RBER	RSVD				E1AL			E0AL			ETFS	PFS			MPS	

Bit Range	Default and Access	Field Name (ID): Description
31:29	0h RV	**Reserved**.
28	0b RV	**Function Level Reset Capable (FLRC):** Reserved, not applicable.
27:26	0h RO	**Captured Slot Power Limit Scale (CSPS):** This value is set by the set_slot_power_limit message.
25:18	0h RO	**Captured Slot Power Limit Value (CSPV):** This value is set by the set_slot_power_limit message.
17:16	0h RV	Reserved.
15	1b RO	**Role Based Error Reporting (RBER):** The PCIe cluster supports Role-based Error reporting.
14:12	0h RV	**Reserved**.
11:9	0h RO	**Endpoint L1 Acceptable Latency (E1AL):** Reserved
8:6	0h RO	**Endpoint L0 Acceptable Latency (E0AL):** Reserved

Bit Range	Default and Access	Field Name (ID): Description
5	0h RO	Extended Tag Field Supported (ETFS): 0: The PCIe supports 5-bit extended tag. 1: The PCIe supports 8-bit extended tag.
4:3	0h RO	**Phantom Functions Supported (PFS):** No phantom functions supported.
2:0	2h RW/L	**Max Payload Size Supported (MPS):** BIOS should write to this field during system initialization. The Max Payload Size supported is 512 bytes. 000b: 128 bytes max payload size. 001b: 256 bytes max payload size. 010b: 512 bytes max payload size. 011b: 1024 bytes max payload size. 100b: 2048 bytes max payload size. 101b: 4096 bytes max payload size. 110b: Reserved. 111b: Reserved.

22.8.17 Device Control; Device Status (DCTL_DSTS)—Offset 48h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00002000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD										TDP	APD	URD	FED	NFED	CED	RSVD	MRRS			ENS	APME	PFE	ETFE	MPS			ERO	URE	FEE	NFE	CEE	

Bit Range	Default and Access	Field Name (ID): Description
31:22	0h RV	Reserved. Software must always write a “0” to these bits.
21	0b RO-V	**Transactions Pending (TDP):** Not applicable
20	0b RO	**AUX Power Detected (APD):** Auxiliary power is not supported.
19	0b RW/1C/V	**Unsupported Request Detected (URD):** Indicates an unsupported request was detected from the PCI Express link.
18	0b RW/1C/V	**Fatal Error Detected (FED):** Indicates a fatal error was detected. Errors are logged in this register regardless of whether error reporting is enabled or not in the Device Control Register.
17	0b RW/1C/V	**Non-Fatal Error Detected (NFED):** Indicates a non-fatal error was detected. Errors are logged in this register regardless of whether error reporting is enabled or not in the Device Control Register.
16	0b RW/1C/V	**Correctable Error Detected (CED):** Indicates a correctable error was detected. Errors are logged in this register regardless of whether error reporting is enabled or not in the Device Control Register.
15	0h RV	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
14:12	2h RW	**Max Read Request Size (MRRS):** This field sets the maximum Read Requests size of the function as a requester. The Function must not generate read requests with size exceeding the set value. 000b: 128 bytes maximum Read Request size 001b: 256 bytes maximum Read Request size 010b: 512 bytes maximum Read Request size 011b: 1024 bytes maximum Read Request size 100b: 2048 bytes maximum Read Request size 101b: 4096 bytes maximum Read Request size Others: Reserved
11	0h RO	**Enable No Snoop (ENS):** Not supported.
10	0h RO	**Aux Power PM Enable (APME):** Not supported.
9	0h RO	**Phantom Functions Enable (PFE):** Not supported.
8	0h RO	**Extended Tag Field Enable (ETFE):** Not supported.
7:5	0h RW	**Max Payload Size (MPS):** This field sets maximum TLP payload size for the function. As a receiver, the function must handle TLPs as large as the set value. As a Transmitter, the function must not generate TLPs exceeding the set value. 000b: 128 bytes maximum payload size (default) 001b: 256 bytes maximum payload size 010b: 512 bytes maximum payload size 011b: 1024 bytes maximum payload size 100b: 2048 bytes maximum payload size 101b: 4096 bytes maximum payload size Others: Reserved
4	0h RO	**Enable Relaxed Ordering (ERO):** Not supported.
3	0h RW	**Unsupported Request Reporting Enable (URE):** This bit controls the enabling of ERR_CORR, ERR_NONFATAL or ERR_FATAL messages on PCI Express for reporting “Unsupported Request” errors.
2	0h RW	**Fatal Error Reporting Enable (FEE):** When this bit is set, generation of the ERR_FATAL message is enabled.
1	0h RW	**Non-Fatal Error Reporting Enable (NFE):** When this bit is set, generation of the ERR_NONFATAL message is enabled.
0	0h RW	**Correctable Error Reporting Enable (CEE):** When this bit is set, generation of the ERR_CORR message is enabled.

22.8.18 Link Capabilities (LCAP)—Offset 4Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 0041C903h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0	1	1	0	0	1	0	0	1	0	0	0	0	0	0	1	1
PN								RSVD	ASPMOC	LBNC	LARC	SDERC	CPM	EL1			EL0		APMS			MLW				MLS					

Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RW-L	**Port Number (PN):** Indicates the port number assigned to the uplink.
23	0b RO	**Reserved**.
22	1b RW-L	**ASPM Optionality Compliance (ASPMOC):** ASPM Optionality Compliance(ASPMOC): This bit must be set to 1b for PCIe 3.0 compliant port. Components implemented against certain earlier versions of this specification will have this bit set to 0b. Software is permitted to use the value of this bit to help determine whether to enable ASPM or whether to run ASPM compliance tests.
21	0b RO	**Link Bandwidth Notification Capability (LBNC):** Reserved, not supported
20	0b RO	**Link Active Reporting Capable (LARC):** Reserved, not supported
19	0b RO	**Surprise Down Error Reporting Capable (SDERC):** Set to '0' to indicate the uplink does not support Surprise Down Error Reporting
18	0b RO	**Clock Power Management (CPM):** '0' Indicates that the upstream port does not support the CLKREQ# mechanism.
17:15	5h RW-L	**L1 Exit Latency (EL1):** Indicates an exit latency of 2 us to 4 us. 000b Less than 1 us 001b 1 us to less than 2 us 010b 2 us to less than 4 us 011b 4 us to less than 8 us 100b 8 us to less than 16 us 101b 16 us to less than 32 us 110b 32 us to 64 us 111b More than 64 us
14:12	4h RW-L	**L0s Exit Latency (EL0):** This field indicates the L0s exit latency for the given PCI-Express Link. It indicates the length of time this port requires to complete transition from L0s to L0. 000b: Less than 64 ns 001b: 64ns to less than 128 ns 010b: 128ns to less than 256 ns 011b: 256ns to less than 512 ns 100b 512 ns to less than 1 µs 101b: 1us to less than 2 us 110b: 2us to less than 4 us 111b: More than 4 us

Bit Range	Default and Access	Field Name (ID): Description
11:10	2h RW-L	**Active State Link PM Support (APMS):** Indicates the level of active state power management on this link Bits Definition 00 No ASPM Supported 01 L0s Supported 10 L1 Supported 11 L0s and L1 supported ***Note:*** L0s should not be advertised when operating with SRIS.
9:4	10h RO/V	**Maximum Link Width (MLW):** This field indicates the maximum link width implemented by the given PCI Express Link. 00h: Reserved 01h: x1 02h: x2 04h: x4 08h: x8 10h: x16 20h: x32 (Unsupported) Others Reserved ***Note:*** Default value is determined by HW after port training is initiated for ports that bifurcate. Hence effective MLW can be smaller than indicated by default
3:0	3h RO/V	**Max Link Speeds (MLS):** Indicates the supported link speeds of the Uplink Port. This field indicates the supported link speed(s) of the associated port. 0001b: 2.5 Gb/s link speed is supported 0010b: 5.0 Gb/s and 2.5 Gb/s link speed supported 0011b: 8.0 Gb/s and 5.0 and 2.5 Gb/s link speed supported Others: Reserved. ***Note:*** The encoding is the binary value of the bit location in the Supported Link Speeds Vector (in the Link Capabilities 2 register) that corresponds to the maximum Link speed.

22.8.19 Link Control; Link Status (LCTL_LSTS)—Offset 50h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00010000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
LABS	LBMS	LA	SCC	LT	RSVD	NLW								CLS				RSVD					LABIE	LBMIE	HAWD	ECPM	ES	CCC	RL	LD	RCBC	RSVD	ASPM		

Bit Range	Default and Access	Field Name (ID): Description
31	0b RO	**Link Autonomous Bandwidth Status (LABS):** This bit is not applicable and hardwired to 0b.
30	0b RO	**Link Bandwidth Management Status (LBMS):** This bit is not applicable and hardwired to 0b.
29	0h RO/V	**Link Active (LA):** Set to 1b when the Data Link Control and Management State Machine is in the DL_Active state, 0b otherwise.

Bit Range	Default and Access	Field Name (ID): Description
28	1b RW-L	**Slot Clock Configuration (SCC):** When to 1b, this bit indicates that the component uses the same physical reference clock that he platform provides on the connector. If the device uses an independent clock irrespective of the presence of a reference on the connector, this bit must be clear. 1: Indicates same reference clock. 0: Indicates independent reference clock
27	0b RO	**Link Training (LT):** This bit is not applicable for an upstream port and is hardwired to 0.
26	0b RV	**Reserved.**
25:20	0h RO/V	**Negotiated Link Width (NLW):** This field indicates the negotiated width of the PCI Express link. 00 0001b: x1 00 0010b: X2 00 0100b: x4 00 1000b: X8 00 1100b: X12—not supported 01 0000b: X16 10 0000b: X32—not supported All other values are reserved. ***Note:*** The value in this field is undefined when the link is not up.
19:16	1h RO/V	**Current Link Speed (CLS):** This field indicates the negotiated link speed of the given PCI Express link. 0001b: 2.5 Gb/s PCI Express Link 0010b: 5.0 Gb/s PCI Express Link 0011b: 8.0 Gb/s PCI Express Link Others: Reserved ***Notes:*** The encoding is the binary value of the bit location in the Supported Link Speeds Vector (in the Link Capabilities 2 register) that corresponds to the current Link speed. The value in this field is undefined when the link is not up.
15:12	0h RO	**Reserved**.
11	0b RO	**Link Autonomous Bandwidth Interrupt Enable (LABIE):** Link Autonomous Bandwidth Interrupt Enable - this is not applicable so hardwired to 0b.
10	0b RO	**Link Bandwidth Management Interrupt Enable (LBMIE):** The uplink does not implement the Link Bandwidth Notification Capability so this is hardwired to 0.
9	0b RW	**Hardware Autonomous Width Disable (HAWD):** When Set, this bit disables hardware from changing the Link width for reasons other than attempting to correct unreliable Link operation by reducing Link width.
8	0b RO	**Enable Clock Power Management (ECPM):** Reserved. Not supported by the upstream port.
7	0h RW	**Extended Synch (ES):** When set, forces extended transmission of FTS ordered sets in FTS and extra TS2 at exit from L1 prior to entering L0.
6	0b RW	**Common Clock Configuration (CCC):** When set, indicates that the root port and device are operating with a distributed common reference clock. A value of 0b indicates that this component and the component at the opposite end of this link are operating with asynchronous reference clock. After changing the value in this bit in both components on a link, software must trigger the link to retrain by writing a 1b to the Retrain Link bit of the downstream port.
5	0b RO	**Retrain Link (RL):** Read only of 0b for an upstream port.
4	0b RO	**Link Disable (LD):** Read only of 0b for an upstream port.

Bit Range	Default and Access	Field Name (ID): Description
3	0b RO	**Read Completion Boundary Control (RCBC):** Indicates the read completion boundary is 64 bytes.
2	0b RV	**Reserved**.
1:0	00b RW	**Active State Link PM Control (ASPM):** Indicates whether the upstream port should enter L0s or L1 or both. Bits Definition 00 Disabled 01 L0s Entry Enabled 10 L1 Entry Enabled 11 L0s and L1 Entry Enabled

22.8.20 Device Capabilities 2 (DCAP2)—Offset 64h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD												OBFFS		RSVD				TPHCS		LTRMS	RSVD				AFS	CTDS	CTRS				

Bit Range	Default and Access	Field Name (ID): Description
31:20	0h Rv	**Reserved**.
19:18	00b RO	**Optimized Buffer Flush/Fill Supported (OBFFS):** 00b - OBFF is not supported.
17:14	0h RO	**Reserved**.
13:12	00b RO	**TPH Completer Supported (TPHCS):** 00: TPH and Extended TPH Completer not supported
11	0b RW-L	**LTR Mechanism Supported (LTRMS):** A value of 1b indicates support for the optional Latency Tolerance Reporting (LTR) mechanism capability.
10	0b RO	**No RO-enabled PR-PR Passing (NROEPRPASS):** Set to 0b as this feature is not supported.
9	0b RO	**AD128 CAS Completer Supported (AD128ACS):** Not supported
8	0b RO	**AD64-bit AtomicOP Completer Supported (AD64ACS):** Not supported
7	0b RO	**AD32 bit AtomicOP Completere Supported (AD32ACS):** not supported
6	0b RW-L	**AtomicOP Routing Supported (ARS)** This function is not supported. Any writes should be “0”.

Bit Range	Default and Access	Field Name (ID): Description
5	0b RO	**ARI Forwarding Supported (AFS):** Not Supported
4	0b RO	**Completion Timeout Disable Supported (CTDS):** A value of 1b indicates support for the Completion Timeout Disable mechanism.
3:0	0h RO	**Completion Timeout Ranges Supported (CTRS):** Not applicable to an upstream port, set to 0h

22.8.21 Device Control 2; Device Status 2 (DCTL2_DSTS2)—Offset 68h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																		OBFFEN		RSVD		LTREN	IDOCE	IDORE	AEB	AFE	AFE	CTD	CTV			

Bit Range	Default and Access	Field Name (ID): Description
31:15	0h RV	Reserved.
14:13	00b RO	**Optimized Buffer Flush/Fill Enable (OBFFEN):** Not supported so RO.
12:11	0h RO	**Reserved**.
10	0h RW	**LTR Mechanism Enable (LTREN):** Not supported
9	0b RO	**IDO Completion Enable (IDOCE):**. Not supported
8	0b RO	**IDO Request Enable (IDORE):** Not supported
7	0b RO	**AtomicOp Egress Blocking (AEB):** Not supported
6	0b RO	**AtomicOp Requester Enable (ARE):** Not supported
5	0b RO	**ARI Forwarding Enable (AFE):** Not applicable
4	0b RO	**Completion Timeout Disable (CTD):** Not supported
3:0	0h RO	**Completion Timeout Value (CTV):** Not supported

22.8.22 Link Capabilities 2 (LCAP2)—Offset 6Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 0000000Eh

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0
RSVD											LSOSRSS								LSOSGSSV				CS				SLSV				RSVD	

Bit Range	Default and Access	Field Name (ID): Description
31:23	0h RV	**Reserved**.
22:16	0h RW-L	**Lower SKP OS Reception Supported Speeds Vector (LSOSRSS):** If this field is non-zero, it indicates that the Port, when operating at the indicated speed(s) supports SRIS and also supports receiving SKP OS at the rate defined for SRNS while running in SRIS. Bit definitions within this field are: Bit 0 2.5 GT/s Bit 1 5.0 GT/s Bit 2 8.0 GT/s Bits 6:3 RsvdP
15:9	0h RW-L	**Lower SKP OS Generation Supported Speeds Vector (LSOSGSSV):** If this field is non-zero, it indicates that the Port, when operating at the indicated speed(s) supports SRIS and also supports software control of the SKP Ordered Set transmission scheduling rate. Bit definitions within this field are: Bit 0 2.5 GT/s Bit 1 5.0 GT/s Bit 2 8.0 GT/s Bits 6:3 RsvdP
8	0b RW-L	Crosslink Supported (CS): Not supported
7:1	07h RO/V	**Supported Link Speeds Vector (SLSV):** This field indicates the supported Link speed(s) of the associated Port. For each bit, a value of 1b indicates that the corresponding Link speed is supported; otherwise, the Link speed is not supported. Bit definitions are: bit 1 2.5 GT/s bit 2 5.0 GT/s bit 3 8.0 GT/s Others: Reserved
0	0b RV	**Reserved**.

22.8.23 Link Control 2; Link Status 2 (LCTL2_LSTS2)—Offset 70h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000003h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
RSVD										LER	EQP3S	EQP2S	EQP1S	EqC	CDL	CD			CSOS	EMC	TM			SD	HASD	EC	TLS				

Bit Range	Default and Access	Field Name (ID): Description
31:22	0h RO	Reserved.
21	0b RW1CS	**Link Equalization Request (LER):** This bit is set by hardware to request the Link equalization process to be performed on the Link.
20	0b ROS-V	**Equalization Phase 3 Successful (EQP3S):** When set to 1, this bit indicates that Phase 3 of the Transmitter Equalization procedure has successfully completed.
19	0b ROS-V	**Equalization Phase 2 Successful (EQP2S):** When set to 1, this bit indicates that Phase 2 of the Transmitter Equalization procedure has successfully completed.
18	0b ROS-V	**Equalization Phase 1 Successful (EQP1S):** When set to 1, this bit indicates that Phase 1 of the Transmitter Equalization procedure has successfully completed.
17	0b ROS-V	**Equalization Complete (EqC):** When set to 1, this bit indicates that the Transmitter Equalization procedure has completed
16	0b RO V	**Current De-emphasis Level (CDL):** When the Link is operating at 5.0 GT/s speed, this bit reflects the level of de-emphasis. Encodings: 1b -3.5 dB 0b -6 dB The value in this bit is undefined when the Link is not operating at 5.0 GT/s speed.

Bit Range	Default and Access	Field Name (ID): Description
15:12	0h RWS	**Compliance Preset/De-emphasis (CD):** For 8 GT/s and 5 GT/s: This bit sets the transmitter preset level forPolling.Compliance state if the entry occurred due to the Enter Compliance bit being 1b. The Encoding are defined as follows: 8 GT/s Rate: 0000b: -6 dB for de-emphasis, 0 dB for preshoot 0001b: -3.5 dB for de-emphasis, 0 dB for preshoot 0010b: -4.5 dB for de-emphasis, 0 dB for preshoot 0011b: -2.5 dB for de-emphasis, 0 dB for preshoot 0100b: 0 dB for de-emphasis, 0 dB for preshoot 0101b: 0 dB for de-emphasis, 2 dB for preshoot 0110b: 0 dB for de-emphasis, 2.5 dB for preshoot 0111b: -6 dB for de-emphasis, 3.5 dB for preshoot 1000b: -3.5 dB for de-emphasis, 3.5 dB for preshoot 1001b: 0 dB for de-emphasis, 3.5 dB for preshoot 1010b: The Transmitter Preset encoding of 1010b corresponds to the maximum Vtx-boost that the Transmitter advertises with FS and LF in both full swing and reduced swing modes with C_{-1} = 0 and C_{+1} = FS – (2*LF). All other encodings are reserved. 5 GT/s Rate: 001b: -3.5dB 000b: -6 dB All other are reserved. When the link is operating at 2.5 Gb/s, the setting of this bit has no effect.
11	0h RWS	**Compliance SOS (CSOS):** When set to 1b, the LTSSM is required to send SKP Ordered Sets periodically in between the (modified) compliance patterns. The default value of this bit is 0b.
10	0b RWS	**Enter Modified Compliance (EMC):** When this bit is set to 1b, the device transmits Modified Compliance Pattern if the LTSSM enters Polling.Compliance substate. Default value of this bit is 0b. This register is intended for debug, compliance testing purposes only. System firmware and software is allowed to modify this register only during debug or compliance testing. In all other cases, the system must ensure that this register is set to the default value.
9:7	0h RWS-V	**Transmit Margin (TM):** This field controls the value of the non-de-emphasized voltage level at the transmitter pins. This field is reset to 000b on entry to the LTSSM Polling.Configuration substate. 000b: Normal operating range 001b: 800-1200 mV for full swing and 400-700 mV for half-swing Others: Reserved This register is intended for debug, compliance testing purposes only. System firmware and software is allowed to modify this register only during debug or compliance testing. In all other cases, the system must ensure that this register is set to the default value.
6	0b RO	**Selectable De-emphasis (SD):** not supported.
5	0b RWS	**Hardware Autonomous Speed Disable (HASD):** When set, this bit disables hardware from changing the link speed for device specific reasons other than attempting to correct unreliable link operation by reducing link speed for device-specific reasons other than attempting to correct unreliable link operations by reducing link speed. Initial transition to the highest supported common link speed is not blocked by this bit.

Bit Range	Default and Access	Field Name (ID): Description
4	0b RWS-V	**Enter Compliance (EC):** Software is permitted to force a Link to enter Compliance mode at the speed indicated in the Target Link Speed field by setting this bit to 1b in both components on a Link and then initiating a hot reset on the Link. Default value of this bit following Fundamental Reset is 0b. This bit is intended for debug, compliance testing purposes only. System firmware and software is allowed to modify this bit only during debug or compliance testing. In all other cases, the system must ensure that this bit is set to the default value. This bit is intended for debug, compliance testing purposes only. System firmware and software is allowed to modify this bit only during debug or compliance testing. In all other cases, the system must ensure that this bit is set to the default value.
3:0	3h RWS-V	**Target Link Speed (TLS):** For downstream ports and root ports, this field sets an upper limit on link operational speed by restricting the values advertised by the Downstream Port in its training sequences. 0001b: 2.5 Gb/s Target Link Speed 0010b: 5.0 Gb/s Target Link Speed 0011b: 8.0 Gb/s Target Link Speed Others: Reserved ***Notes:*** The encoding is the binary value of the bit in the Supported Link Speeds Vector (in the Link Capabilities 2 register) that corresponds to the desired target Link speed. All other encodings are reserved. If a value is written to this field that does not correspond to a speed included in the Supported Link Speeds field, the result is undefined. The default value of this field is the highest Link speed supported by the component (as reported in the Max Link Speed field of the Link Capabilities register) unless the corresponding platform/form factor requires a different default value. For both Upstream and Downstream Ports, this field is used to set the target compliance mode speed when software is using the Enter Compliance bit to force a Link into compliance mode.

22.8.24 Power Management Capability; PCI Power Management Capabilities (PMCAP_PMC)—Offset 80h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: C8038801h

31			28				24				20				16				12				8				4				0
1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	1
PMES						D2S	D1S	AC			DSI	RSVD	PMEC	VS			NEXT							CID								

Bit Range	Default and Access	Field Name (ID): Description
31:27	19h RO	**PME Support (PMES):** Indicates PME# is supported for states D0, $D3_{HOT}$ and $D3_{COLD}$. There is no PME support for $D3_{COLD}$ as the part does not have the capability, but for PCIe compliance purposes, bit 15is set.
26	0b RO	**D2_Support (D2S):** The D2 state is not supported.
25	0b RO	**D1_Support (D1S):** The D1 state is not supported.
24:22	0h RO	**Aux_Current (AC):** Reports 0mA (self-powered),
21	0b RO	**Device Specific Initialization (DSI):** Indicates that no device-specific initialization is required.
20	0h RO	**Reserved**.
19	0h RO	**PME Clock (PMEC):** Indicates that PCI clock is not required to generate PME#.
18:16	3h RO	**Version (VS):** Indicates support for Revision 1.2 of the PCI Power Management Specification.
15:8	88h RO	**Next Capability (NEXT):** Contains the offset of the next item in the Capabilities List.
7:0	1h RO	**Capability Identifier (CID):** Value of 01h indicates this is a PCI power management capability.

22.8.25 PCI Power Management Control And Status (PMCS)—Offset 84h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000008h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
RSVD								BPCE	B23S	RSVD							PMES	DSC	DSEL				PMEE	RSVD				NSR	RSVD	PS		

Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RV	**Reserved**
23	0b RO	**Bus Power / Clock Control Enable (BPCE):** Reserved per *PCI Express Specification*.
22	0b RO	**B2/B3 Support (B23S):** Reserved per *PCI Express Specification*.
21:16	0h RV	**Reserved.**
15	0b RW1CS	**PME Status (PMES):** Indicates a PME was received.
14:13	0h RO	**Data Scale (DSC):** Not supported
12:9	0h RO	**Data Select (DSEL):** Not supported
8	0b RWS	**PME Enable (PMEE):** Gates assertion of the PME message.
7:4	0h RV	**Reserved**.
3	1b RWL	**No Soft Reset (NSR):** When set to 1 this bit indicates that devices transitioning from $D3_{HOT}$ to D0 because of Power State commands do not perform an internal reset. Configuration context is preserved. Upon transition from $D3_{HOT}$ to D0 Initialized state, no additional operating system intervention is required to preserve Configuration Context beyond writing the Power State bits. When clear, devices do perform an internal reset upon transitioning from $D3_{HOT}$ to D0 via software control of the Power State bits. Configuration Context is lost when performing the soft reset. Upon transition from $D3_{HOT}$ to D0 state, full reinitialization sequence is needed to return the device to D0 Initialized.
2	0b RV	**Reserved**.
1:0	0h RW	**Power State (PS):** This field is used both to determine the current power state of the upstream port and to set a new power state. The values are: 00 D0 state 11 $D3_{HOT}$ state If software attempts to write a '10' or '01' to these bits, the write will be ignored.

22.8.26 Subsystem Capability List Register (SVCAP)—Offset 88h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 000000Dh

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	1	1	0	1
RSVD																NEXT								CID							

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h Rv	Reserved.
15:8	0h RO	**Next Capability (NEXT):** For upstream ports there is no MSI structure so the capability is terminated
7:0	Dh RO	**Capability Identifier (CID):** Value of 0Dh indicates the function as subsystem identification capable.

22.8.27 Subsystem Vendor IDs (SVID)—Offset 8Ch

Size:32 bits.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00008086h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0
SID																SVID															

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RW-L	**Subsystem Identifier (SID):** Indicates the subsystem as identified by the vendor. This field is write once and is locked down until a bridge reset occurs (not the PCI bus reset).
15:0	8086h RW-L	**Subsystem Vendor Identifier (SVID):** Indicates the manufacturer of the subsystem. This field is write once and is locked down until a bridge reset occurs (not the PCI bus reset).

22.8.28 Advanced Error Extended Reporting Capability Header (AECH)—Offset 100h

Size:32 bits.

The AER capability can optionally be included or excluded from the capabilities list. The full AER is supported.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 20010001h

31			28				24				20				16				12				8				4				0
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
NCO													CV				CID															

Bit Range	Default and Access	Field Name (ID): Description
31:20	200h RW-L	**Next Capability Offset (NCO):** Contains the offset of the next structure in the extended capabilities list.
19:16	1h RO	**Capability Version (CV):** Indicates the version of the capability structure present.
15:0	0001h RO	**Capability ID (CID):** Identifies the function as Advanced Error Reporting capable.

22.8.29 Uncorrectable Error Status (UES)—Offset 104h

This register must maintain its state through a platform reset. It loses its state upon loss of core power.

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD						PLTLPEBS	TPEBS	AEBE	MCE	UIE	AVS	URE	EE	MT	RO	UC	CA	CT	FCPE	PT	RSVD							SDE	DLPE	RSVD			

Bit Range	Default and Access	Field Name (ID): Description
31:27	000h RV	Reserved.
26	0b RO	**Poisoned TLP Egress Blocked Status (PLTLPEBS)**
25	0b RO	**TLP Prefix Blocked Error Status (TPBES)**
24	0b RW1CS	**Atomic Egress Blocked Status (AEBE):** This bit will never be set
23	0b RW1CS	**MC Blocked TLP Status (MCE):** This bit is set whenever a Multicast TLP is blocked
22	0b RW1CS	**Uncorrectable Internal Error Status (UIE):** This bit is set whenever an uncorrectable internal error is detected.
21	0b RO	**ACS Violation Status (AVS):** Reserved, not supported
20	0b RW1CS	**Unsupported Request Error Status (URE):** Indicates an unsupported request was received.
19	0b RW1CS	**ECRC Error Status (EE):** Set to 1b when an ECRC is detected
18	0b RW1CS	**Malformed TLP Status (MT):** Indicates a malformed TLP was received.
17	0b RW1CS	**Receiver Overflow Status (RO):** Indicates a receiver overflow occurred.
16	0b RW1CS	**Unexpected Completion Status (UC):** Indicates an unexpected completion was received.
15	0b RW1CS	**Completor Abort Status (CA):** Indicates a completer abort was received
14	0b RO	**Completion Timeout Status (CT):** not applicable as the upstream port will not issue non-posted requests on it's one behalf.
13	0b RW1CS	**Flow Control Protocol Error Status (FCPE):** This bit is set when a flow control error protocol is detected.
12	0b RW1CS	**Poisoned TLP Status (PT):** Indicates a poisoned TLP was received.
11:6	0h RV	**Reserved.** Software must write '0' to these bits
5	0b RO	**Surprise Down Error Status (SDE):** Surprise Down is not supported.
4	0b RW1CS	**Data Link Protocol Error Status (DLPE):** Indicates a data link protocol error occurred.
3:0	0h RV	**Reserved.**

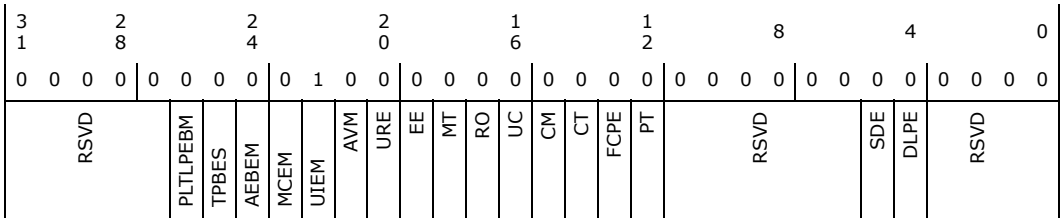
22.8.30 Uncorrectable Error Mask (UEM)—Offset 108h

When set, the corresponding error in the UES register is masked, and the logged error will cause no action. When cleared, the corresponding error is enabled. This register is only reset by a loss of core power.

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00200000h



Bit Range	Default and Access	Field Name (ID): Description
31:27	0h RV	**Reserved.**
26	0b RO	**Poisoned TLP Egress Blocked Mask (PLTLPEBM)**: Not supported
25	0b RO	**TLP Prefix Blocked Error Mask (TPBES)**: not supported
24	0b RWS	**AtomicOP Egress Blocked Mask (AEBEM)**
23	0b RWS	**MC Blocked TLP Mask (MCEM):**
22	1b RWS	**Uncorrectable Internal Error Mask (UIEM)**
21	0b RO	**ACS Violation Mask (AVM):** Reserved. Access Control Services are not supported
20	0b RWS	**Unsupported Request Error Mask (URE):**
19	0b RWS	**ECRC Error Mask (EE):**
18	0b RWS	**Malformed TLP Mask (MT):**
17	0b RWS	**Receiver Overflow Mask (RO):**
16	0b RWS	**Unexpected Completion Mask (UC):**
15	0b RWS	**Completor Abort Mask (CM):**
14	0b RWS	**Completion Timeout Mask (CT):** Not supported.
13	0b RWS	**Flow Control Protocol Error Mask (FCPE):**

Bit Range	Default and Access	Field Name (ID): Description
12	0b RWS	**Poisoned TLP Mask (PT):**
11:6	0h RV	**Reserved.**
5	0h RO	**Surprise Down Error Mask (SDE):** not supported.
4	0b RWS	**Data Link Protocol Error Mask (DLPE):**
3:0	0h RV	**Reserved.**

22.8.31 Uncorrectable Error Severity (UEV)—Offset 10Ch

This register gives the option to make an uncorrectable error fatal or non-fatal. An error is fatal if the bit is set. An error is non-fatal if the bit is cleared. This register is only reset by a loss of core power.

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00262010h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0
RSVD					PTLBEBS	TPBES	AEBES	MCES	UIES	AVS	URE	EE	MT	RO	UC	CA	CT	FCPE	PT	RSVD					SDE	DLPE	RSVD				

Bit Range	Default and Access	Field Name (ID): Description
31:27	000h RV	**Reserved.**
26	0b RO	**Poisoned TLP Egress Blocked Severity (PTLBEBS)**
25	0b RO	**TLP Prefix Blocked Error Severity (TPBES)**
24	0b RWS	**AtomicOP Egress Blocked Severity (AEBES)**
23	0b RWS	**MC Blocked TLP Severity (MCES)**
22	1b RWS	**Uncorrectable Internal Error Severity (UIES)**
21	0b RO	**ACS Violation Severity (AVS):** Not supported
20	0b RWS	**Unsupported Request Error Severity (URE):**

Bit Range	Default and Access	Field Name (ID): Description
19	0b RWS	**ECRC Error Severity (EE):**
18	1h RWS	**Malformed TLP Severity (MT):**
17	1b RWS	**Receiver Overflow Severity (RO):**
16	0b RWS	**Unexpected Completion Severity (UC):**
15	0b RWS	**Completor Abort Severity (CA):**
14	0b RO	**Completion Timeout Severity (CT):** Not Supported
13	1b RWS	**Flow Control Protocol Error Severity (FCPE):**
12	0b RWS	**Poisoned TLP Severity (PT):**
11:6	0h RV	Reserved.
5	0b RO	**Surprise Down Error Severity (SDE):** Not supported.
4	1b RWS	**Data Link Protocol Error Severity (DLPE):**
3:0	0h RV	Reserved.

22.8.32 Correctable Error Status (CES)—Offset 110h

This register is only reset by a loss of core power.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																HLOE	CIE	ANFES	RTT	RSVD			RNR	BD	BT	RSVD					RE	

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RV	**Reserved**. Software must write a "0" to these bits.
15	0b RW1CS	**Header Log Overflow Error (HLOE)**
14	0b RW1CS	**Correctable Internal Error (CIE)**
13	0b RW1CS	**Advisory Non-Fatal Error Status (ANFES):**
12	0b RW1CS	**Replay Timer Timeout Status (RTT):** Indicates the replay timer timed out.
11:9	0h RV	**Reserved.** Software must write '0' to these bits.
8	0b RW1CS	**Replay Number Rollover Status (RNR):** Indicates the replay number roll over from 11 to 00.
7	0b RW1CS	**Bad DLLP Status (BD):** Indicates a bad DLLP was received.
6	0b RW1CS	**Bad TLP Status (BT):** Indicates a bad TLP was received.
5:1	0h RV	**Reserved**. Software must write '0' to these bits.
0	0b RW1CS	**Receiver Error Status (RE):** Indicates that the physical layer detected a receiver error.

22.8.33 Correctable Error Mask (CEM)—Offset 114h

When set, the corresponding error in the CES register is masked, and the logged error will cause no action. When cleared, the corresponding error is enabled. This register is only reset by a loss of core power.

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 0000E000h

31			28				24				20				16				12				8			4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																HLOEM	CIEM	ANFEM	RTT	RSVD			RNR	BD	BT	RSVD				RE

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RV	Reserved.
15	1b RWS	**Header Log Overflow Error Mask (HLOEM)**
14	1b RWS	**Correctable Internal Error Mask (CIEM)**
13	1b RWS	**Advisory Non-Fatal Error Mask (ANFEM):** When set, masks Advisory Non-Fatal errors from (a) signaling ERR_COR to the device control register and (b) updating the Uncorrectable Error Status register. This register is set by default to enable compatibility with software that does not comprehend Role-Based Error Reporting.
12	0b RWS	**Replay Timer Timeout Mask (RTTEM):**
11:9	0h RO	**Reserved**.
8	0b RWS	**Replay Number Rollover Mask (RNREM):**
7	0b RWS	**Bad DLLP Mask (BDLLPEM):.**
6	0b RWS	**Bad TLP Mask (BTLPEM):**
5:1	0h RO	**Reserved**.
0	0h RW/P	**Receiver Error Mask (REM):**

22.8.34 Advanced Error Capabilities and Control (AECC)—Offset 118h

This register gives the status and control for ECRC checks, and also the pointer to the first uncorrectable error that happened. This register is only reset by a loss of core power.

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																			CTPHLC	TLPPLP	MHRE	MHRC	ECE	ECC	EGE	EGC	FEP					

Bit Range	Default and Access	Field Name (ID): Description
31:13	0h RV	Reserved.
12	0b RO	**Completion Timeout Prefix/Header Log Capable (CTPHLC):** If Set, this bit indicates that the Function records the prefix/header of Request TLPs that experience a Completion Timeout error.
11	0b ROSV	**TLP Prefix Log Present (TLPPLP):** If Set and the First Error Pointer is valid, indicates that the TLP Prefix Log register contains valid information. If Clear or if First Error Pointer is invalid, the TLP Prefix Log register is undefined.
10	0b RO	**Multiple Header Recording Enable (MHRE):** Multiple Header recording is not supported
9	0b RO	**Multiple Header Recording Capable (MHRC):** Multiple header recording is not supported.
8	0b RWS	**ECRC Check Enable (ECE):**
7	0b RWL	**ECRC Check Capable (ECC):**
6	0b RWS	**ECRC Generation Enable (EGE):**
5	0b RWL	**ECRC Generation Capable (EGC):**.
4:0	0h ROSV	**First Error Pointer (FEP):** This field identifies the bit position of the first error reported in the UncorrectableError Status Register (xref). This register re-arms itself (which does not change its current value) as soon as the error status bit indicated by the pointer is cleared by the software by writing a 1 to that status bit.

22.8.35 Header Log Register (AERHDRLOG[1-4])—Offset 11C - 128h by 4h

This register logs the 4 Dwords of the transaction header for PCI Express Errors.

Type: CFG Register
(Size: 32 bits)

Device:
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
AERHDRLOG																															

Bit Range	Default and Access	Field Name (ID): Description
31:0	00000000h ROSV	**TLP Header Log (TLPHDRLOG)**: As soon as an error is logged in this register, it remains locked for further error-logging until the software clears the status bit that caused the header log, thus rearming the logging of errors

22.8.36 Uncorrectable Error Detect Mask Register (ERRUNCDETMSK)—Offset 140h

A value of 1 masks the field and prevents detection as well as logging of errors.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 04000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
RSVD					PTLPEDM	TPBEDM	AEBEDM	MCEDM	UIEDM	ACSEDM	UREDM	ECRCEDM	MTLPEDM	ROEDM	UCEDM	CAEDM	CTEDM	FCEDM	PTLPEDM	RSVD				SDEDM	DLPEDM	RSVD							

Bit Range	Default and Access	Field Name (ID): Description
31:27	0h RV	**Reserved**.
26	1b RWS	**Poisoned TLP Egress Blocked Detect Mask (PLTPEBDM)**
25	0b RO	**TLP Prefix Blocked Error Detect Mask (TPBEDM):** Not Supported
24	0b RWS	**AtomicOP Egress Blocked Error Detect Mask (AEBEDM)**
23	0b RWS	**MC Blocked TLP Error Detect Mask (MCEDM)**
22	0b RWS	**Uncorrectable Internal Error Detect Mask (UIEDM)**
21	0b RO	**ACS Violation Error Detect Mask (ACSEDM)**: Not Supported
20	0b RWS	**Unsupported Request Error Detect Mask (UREDM)**
19	0b RWS	**ECRC Check Error Mask (ECRCEDM)**
18	0b RWS	**Malformed TLP Error Detect Mask (MTLPEDM)**
17	0b RWS	**Receiver Overflow Error Detect Mask (ROEDM)**
16	0b RWS	**Unexpected Completion Error Detect Mask (UCEDM)**
15	0b RWS	**Completer Abort Error Detect Mask (CAEDM)**
14	0b RWS	**Completion Timeout Error Detect Mask (CTEDM)**
13	0b RWS	**Flow Control Error Detect Mask (FCEDM)**

Bit Range	Default and Access	Field Name (ID): Description
12	0b RWS	**Poisoned TLP Error Detect Mask (PTLPEDM)**
11:6	00h RV	**Reserved**
5	0b RO	**Surprise Down Error Detect Mask (SLDEDM):** Not Supported
4	0b RWS	**Data Link Protocol Error Detect Mask (DLPEDM)**
3:0	0h RV	**Reserved**

22.8.37 Correctable Error Detect Mask Register (ERRCORDETMSK)—Offset 144h

Type: CFG Register
(Size: 32 bits)

Device:
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																HLOEDM	CIEDM	ANFEDM	RTTEDM	RSVD			RNREDM	BDLLLPEDM	BTLPEDM	RSVD					REDM	

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RV	**Reserved**.
15	0b RWS	**Header Log Overflow Error Detect Mask (HLOEDM)**
14	0b RWS	**Correctable Internal Error DectMask (CIEDM)**
13	0b RWS	**Advisory Non-Fatal Error Detect Mask (ANFEDM):**
12	0b RWS	**Replay Timer Timeout Detect Mask (RTTEDM):**.
11:9	0h RO	**Reserved.**
8	0b RWS	**Replay Number Rollover Detect Mask (RNREDM):**
7	0b RWS	**Bad DLLP Detect Mask (BDLLPEDM):.**

Bit Range	Default and Access	Field Name (ID): Description
6	0b RWS	**Bad TLP Detect Mask (BTLPEDM):**
5:1	0h RO	**Reserved**.
0	0h RW/P	**Receiver Error Detect Mask (REDM):**

22.8.38 Secondary PCI Express Extended Capability Header (SECEXPCAP)—Offset 200h

Type: CFG Register
(Size: 32 bits)

Device: 00
Function: 0

Default: 00010019h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1
NCO													CV				PCIEECID															

Bit Range	Default and Access	Field Name (ID): Description
31:20	0h RW/O	**Next Capability Offset (NCO):** Next Capability Offset (NCO): This field contains the offset to the next PCI Express Capability structure or 000h if no other items exist in the linked list of Capabilities.
19:16	1b RO	**Capability Version (CV):** Capability Version (CV): This field is a PCI-SIG defined version number that indicates the version of the Capability structure present.
15:0	19h RO	**PCI Express Extended Capability ID (PCIEECID):** PCI Express Extended Capability ID for the Secondary PCI Express Extended Capability is 0019h

22.8.39 Link Control 3 (LCTL3)—Offset 204h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																ELSOSGV							RSVD							LERIE	PE

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	**Reserved**.
15:9	0h RW	**Enable Lower SKP OS Generation Vector (ELSOSGV):** Enable Lower SKP OS Generation Vector(ELSOSGV): When the Link is in L0 and the bit in this field corresponding to the current Link speed is Set, SKP Ordered Sets are scheduled at the rate defined for SRNS, overriding the rate required based on the clock tolerance architecture. Bit definitions within this field are: Bit 0 2.5 GT/s Bit 1 5.0 GT/s Bit 2 8.0 GT/s Bits 6:3 RsvdP Behavior is undefined if a bit is Set in this field and the corresponding bit in the Lower SKP OS Generation Supported Speeds Vector is not set.
8:2	0h RO	**Reserved.**
1	0h R0	**Link Equalization Request Interrupt Enable (LNKEQREQINTEN):** Not applicable for upstream ports
0	0h R0	**Perform Equalization (PE):** Not applicable for upstream ports

22.8.40 Lane Error Status Register (LANEERRSTS)—Offset 208h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																L15ERR	L14ERR	L13ERR	L12ERR	L11ERR	L10ERR	L9ERR	L8ERR	L7ERR	L6ERR	L5ERR	L4EERR	L3ERR	L2ERR	L1ERR	L0ERR	

Bit Range	Default and Access	Field Name (ID): Description
31:16	0000h RV	Reserved.
15	0b RW1CS	**Lane 15 Error (L15ERR)**: A value of 1b indicates that a lane based error was detected on this PCIe Express lane.
14	0b RW1CS	**Lane 14 Error (L14ERR)**: A value of 1b indicates that a lane based error was detected on this PCIe Express lane.
13	0b RW1CS	**Lane 13Error (L13ERR)**: A value of 1b indicates that a lane based error was detected on this PCIe Express lane.
12	0b RW1CS	**Lane 12 Error (L12ERR)**: A value of 1b indicates that a lane based error was detected on this PCIe Express lane.
11	0b RW1CS	**Lane 11 Error (L11ERR)**: A value of 1b indicates that a lane based error was detected on this PCIe Express lane.

Bit Range	Default and Access	Field Name (ID): Description
10	0b RW1CS	**Lane 10 Error (L10ERR)**: A value of 1b indicates that a lane based error was detected on this PCIe Express lane.
9	0b RW1CS	**Lane 9 Error (L9ERR)**: A value of 1b indicates that a lane based error was detected on this PCIe Express lane.
8	0b RW1CS	**Lane 8Error (L8ERR)**: A value of 1b indicates that a lane based error was detected on this PCIe Express lane.
7	0b RW1CS	**Lane 7Error (L7ERR)**: A value of 1b indicates that a lane based error was detected on this PCIe Express lane.
6	0b RW1CS	**Lane 6Error (L6ERR)**: A value of 1b indicates that a lane based error was detected on this PCIe Express lane.
5	0b RW1CS	**Lane 5Error (L5ERR)**: A value of 1b indicates that a lane based error was detected on this PCIe Express lane.
4	0b RW1CS	**Lane 4 Error (L4ERR)**: A value of 1b indicates that a lane based error was detected on this PCIe Express lane.
3	0b RW1CS	**Lane 3Error (L3ERR)**: A value of 1b indicates that a lane based error was detected on this PCIe Express lane.
2	0b RW1CS	**Lane 2Error (L2ERR)**: A value of 1b indicates that a lane based error was detected on this PCIe Express lane.
1	0b RW1CS	**Lane 1Error (L1ERR)**: A value of 1b indicates that a lane based error was detected on this PCIe Express lane.
0	0b RW1CS	**Lane 0 Error (L0ERR)**: A value of 1b indicates that a lane based error was detected on this PCIe Express lane.

22.8.41 Lane Equalization Control Register (LANEEQCTL[0-15])—Offset 20C - 22Ah by 2h (for up to 16 Upstream Lanes)

Each entry contains the values for the Lane with the corresponding default Lane number which is invariant to Link width and Lane reversal negotiation that occurs during Link training.

Type: CFG Register
(Size: 16bits)

Device:
Function: 0

Default: 7F00

15	14	13	12	11	10	9	8	7	6	5	3	2	2	1	0
0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
RSVD	UPRPH			UPTP				RSVD							

Bit Range	Default and Access	Field Name (ID): Description
15	0h RV	**Reserved**.
14:12	7h ROV	**Upstream Port Lane 1 Receiver Preset Hint (UPRPH):** Upstream Port Lane 1 Receiver Preset Hint (UPL1RPH): Field contains the Receiver Preset Hint value sent or received during Link Equalization. The settings are: 000b: -6 dB 001b: -7 dB 010b: -8 dB 011b: -9 dB 100b: -10 dB 101b: -11 dB 110b: -12 dB 111b: Reserved
11:8	Fh ROV	**Upstream Port Lane 1 Transmitter Preset (UPTP):** Upstream Port Lane 1 Transmitter Preset (UPL1TP): Field contains the Transmit Preset value sent or received during Link Equalization. The settings are: 0000b: -6 dB for de-emphasis, 0 dB for preshoot 0001b: -3.5 dB for de-emphasis, 0 dB for preshoot 0010b: -4.5 dB for de-emphasis, 0 dB for preshoot 0011b: -2.5 dB for de-emphasis, 0 dB for preshoot 0100b: 0 dB for de-emphasis, 0 dB for preshoot 0101b: 0 dB for de-emphasis, 2 dB for preshoot 0110b: 0 dB for de-emphasis, 2.5 dB for preshoot 0111b: -6 dB for de-emphasis, 3.5 dB for preshoot 1000b: -3.5 dB for de-emphasis, 3.5 dB for preshoot 1001b: 0 dB for de-emphasis, 3.5 dB for preshoot 1010b: The Transmitter Preset encoding of 1010b corresponds to the maximum Vtx-boost that the Transmitter advertises with FS and LF in both full swing and reduced swing modes with C_{-1} = 0 and C_{+1} = FS – (2*LF). All other encodings are reserved.
7:0	00h RV	Reserved.

22.8.42 Implementation Specific Registers

The following register lies in PCIe configuration space. It is not part of any PCIe specification register definition and is unique to the implementation on the Intel® C620 Series Chipset. With the normal register discovery mechanism this can not be found, and is only discovered by being listed here.

22.8.43 Personality Lock Key Control Register (PLKCTL) —Offset EAh

Type: CFG Register
(Size: 32 bits)

Device:
Function: 0

Default: 00000000

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																															CL

Bit Range	Default and Access	Field Name (ID): Description
31:1	0h Rv	Reserved.
0	0b RW-L	**Capability Lock (CL):** Lock key bit for all RWS-L bits (capabilities, next capability pointer, SSID/SVID, etc.) 1b = lock 0b = unlocked ***Note:*** This bit is self-locking. Once this bit is set to 1b, this key bit can not be unlocked. Writing a 0b has no effect on this bit.

22.9 PCI Express NP8 Uplink Port Configuration Registers Register Summary

The NP16 Uplink is a single endpoint PCIe controller. This controller is configurable into being x8, x4 or x1. However, it is not capable of being configured into two endpoints (2x4 for example). This port is muxed with the output of the SATA and PCIe Root Port controllers and is configured by a combination of fuses and soft straps. This muxing is static, meaning at start up the I/O is configured to be routed to one these 3 ports and does not dynamically change during run time.

Table 22-10. Summary of PCI Express Port Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	Identifiers (ID)—Offset 0h	37418086h
4h	7h	Device Command; Primary Status (CMD_PSTS)—Offset 4h	00100000h
8h	Bh	Revision ID;Class Code (RID_CC)—Offset 8h	06040000h
Ch	Fh	Cache Line Size; Primary Latency Timer; Header Type (CLS_PLT_HTYPE)—Offset Ch	00010000h
10h	17h	Express Port Memory Base Address Register (EXPPTMBAR_UX8) - Offset 10h	0000000000000000
18h	1Bh	Bus Numbers; Secondary Latency Timer (BNUM_SLT)—Offset 18h	000000h
1Ch	1Fh	I/O Base and Limit; Secondary Status (IOBL_SSTS)—Offset 1Ch	00000000h
20h	23h	Memory Base and Limit (MBL)—Offset 20h	00000000h
24h	27h	Prefetchable Memory Base and Limit (PMBL)—Offset 24h	00010001h
28h	2Bh	Prefetchable Memory Base Upper 32 Bits (PMBU32)—Offset 28h	00000000h
2Ch	2Fh	Prefetchable Memory Limit Upper 32 Bits (PMLU32)—Offset 2Ch	00000000h
34h	37h	Capabilities List Pointer (CAPP)—Offset 34h	00000040h
38h	3Bh	Expansion ROM Base Address Register (EXPROM) - Offset 38h	00000000
3Ch	3Fh	Interrupt Information; Bridge Control (INTR_BCTRL)—Offset 3Ch	00000100h
40h	43h	Capabilities List; PCI Express Capabilities (CLIST_XCAP)—Offset 40h	00528010
44h	47h	Device Capabilities (DCAP)—Offset 44h	00008002h
48h	4Bh	Device Control; Device Status (DCTL_DSTS)—Offset 48h	00002000
4Ch	4Fh	Link Capabilities (LCAP)—Offset 4Ch	0042C883h
50h	53h	Link Control; Link Status (LCTL_LSTS)—Offset 50h	00010000h
64h	67h	Device Capabilities 2 (DCAP2)—Offset 64h	00000000h
68h	6Bh	Device Control 2; Device Status 2 (DCTL2_DSTS2)—Offset 68h	00000000h
6Ch	6Fh	Link Capabilities 2 (LCAP2)—Offset 6Ch	0000000Eh

Table 22-10. Summary of PCI Express Port Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
70h	73h	Link Control 2; Link Status 2 (LCTL2_LSTS2)—Offset 70h	00000003h
80h	83h	Power Management Capability; PCI Power Management Capabilities (PMCAP_PMC) - Offset 80h	C8038801h
84h	87h	PCI Power Management Control and Status (PMCS) - Offset 84	00000008h
88h	8Bh	Subsystem Capability List Register (SVCAP) - Offset 88h	0000000Dh
8Ch	8Fh	Subsystem Vendor IDs (SVID) - Offset 8Ch	00008086h
EAh	EFh	Personality Lock Key Control Register (PLKCTL) - Offset EAh	00000000
100h	103h	Advanced Error Extended Reporting Capability Header (AECH)—Offset 100h	20010001h
104h	107h	Uncorrectable Error Status (UES)—Offset 104h	00000000h
108h	10Bh	Uncorrectable Error Mask (UEM)—Offset 108h	00400000h
10Ch	10Fh	Uncorrectable Error Severity (UEV)—Offset 10Ch	00262010h
110h	113h	Correctable Error Status (CES)—Offset 110h	00000000h
114h	117h	Correctable Error Mask (CEM)—Offset 114h	000D0000h
118h	11Bh	Advanced Error Capabilities and Control (AECC)—Offset 118h	00000000h
11Ch	128h	Header Log Register (AERHDRLOG[1-4]) - Offset 11C - 128h	00000000h
140h	143	Uncorrectable Error Detect Mask Register (ERRUNCDETMSK) - Offset 140h	00000000
144h	147h	Correctable Error Detect Mask Register (ERRCORDETMSK) - Offset 144h	00000000h
200h	203h	Secondary PCI Express Extended Capability Header (SECEXPCAP) - Offset 200h	00010019h
204h	207h	Link Control 3 (LCTL3)—Offset 204h	00000000h
208h	20Bh	Lane Error Status Register (LANEERRSTS) - Offset 208h	00000000h
20Ch	21A	Lane Equalization Control Register (LANEEQCTL[0-7]) Offset 20C - 21A	7F00

22.9.1 Identifiers (ID)—Offset 0h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 37C18086h

31			28				24				20				16				12				8				4				0
0	0	1	1	0	1	1	1	1	1	0	0	0	0	0	1	1	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0
DID																VID																

Bit Range	Default and Access	Field Name (ID): Description
31:16	37C1 RO/V	**Device Identification (DID):** Identifies this particular function as the x8 PCIe Upstream port in the Intel® C620 Series Chipset.
15:0	8086h RO	**Vendor Identification (VID):** Indicates Intel.

22.9.2 Device Command; Primary Status (CMD_PSTS)—Offset 4h

Type: CFG Register
(Size: 32 bits)

Device: 00
Function: 0

Default: 00100000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DPE	SSE	RMA	RTA	STA	PDTS		DPD	PFBC	RSVD	PC66	CLIST	IS	RSVD								ID	FBE	SEE	WCC	PERE	VGA_PSE	MWIE	SCE	BME	MSE	IOSE

Bit Range	Default and Access	Field Name (ID): Description
31	0b RW1C/V	**DPE - Detected Parity Error (DPE):** Set when the port receives a poisoned tlp from the PCIe interface of the internal backboner. This is set even if CMD.PERE is not set.
30	0b RW1C/V	**Signaled System Error (SSE):** This bit is set when ERR_FATAL or ERR_NONFATAL messages are sent to the root complex and the SERR enable bit in the PCICMD Register is set
29	0b RW1C/V	**Received Master Abort (RMA):** Set when the requester receives a completion with unsupported request status.
28	0b R/1C/V	**Received Target Abort (RTA):** Set when the root port receives a completion with completer abort.
27	0b RW1C/V	**Signaled Target Abort (STA):** This bit is Set when the port completes a Posted or Non-Posted Request as a Completer Abort error. This applies to a Function with a Type 1 Configuration header when the Completer Abort was generated by its Primary Side.
26:25	0h RO	**Primary DEVSEL# Timing Status (PDTS):** Reserved per *PCI Express Specification*.
24	0b RW1C/V	**Master Data Parity Error Detected (DPD):** This bit is set by a requester (primary side for type1 configuration headerfunctions) if the parity error response enable bit (PERE) in the Command Register is set and either of the following two conditions occur: • Requester receives a completion marked poisoned. • Requester sends a poisoned request (includes writes and messages) If the parity error bit is 0b, this bit is never set.
23	0b RO	**Primary Fast Back to Back Capable (PFBC):** Reserved per *PCI Express Specification*.
22	0b RV	**Reserved.**
21	0b RO	**Primary 66 MHz Capable (PC66):** Reserved per *PCI Express Specification*.
20	1b RO	**Capabilities List (CLIST):** Indicates the presence of a capabilities list.
19	0b RO/V	**Interrupt Status (IS):** When set indicates that an INTx emulation interrupt is pending internally for this function.
18:11	0h RV	**Reserved.**

Bit Range	Default and Access	Field Name (ID): Description
10	0b RW/V	**Interrupt Disable (ID):** This disables pin-based INTx# interrupts on enabled hot plug and power management events. This bit has no effect on MSI operation. When set, internal INTx# messages will not be generated. When cleared, internal INTx# messages are generated if there is an interrupt for hot plug or power management and MSI is not enabled. This bit does not affect interrupt forwarding from devices connected to the root port. Assert_INTx and Deassert_INTx messages will still be forwarded to the internal interrupt controllers if this bit is set. For PCI Bus Emulation Mode compatibility, if the PCIBEM register is set, this register is RO and returns a value of 0 when read, else it is RW with the functionality described above.
9	0b RO	**Fast Back to Back Enable (FBE):** Reserved per *PCI Express Specification*.
8	0b RW	**SERR# Enable (SEE):** When set, enables the root port to generate an SERR# message when PSTS.SSE is set.
7	0b RO	**Wait Cycle Control (WCC):** Reserved per *PCI Express Specification*.
6	0b RW	**Parity Error Response Enable (PERE):** Indicates that the device is capable of reporting parity errors as a master on the backbone.
5	0b RO	**VGA Palette Snoop (VGA_PSE):** Reserved per *PCI Express Specification*.
4	0b RO	**Memory Write and Invalidate Enable (MWIE):** Reserved per *PCI Express Specification*.
3	0b RO	**Special Cycle Enable (SCE):** Reserved per *PCI Express Specification* and *PCI Bridge Specification*.
2	0b RW	**Bus Master Enable (BME):.**This bit controls the ability of the Function to issue Memory and I/O read or write requests, and the ability of Root or Switch port to forward memory and I/O read or write requests in the upstream direction. When this bit is 0b, memory and I/O requests received at the root port or downstream side of a switch port (secondary side) must be handled as an Unsupported Request (UR). For Non-posted requests, a completion with UR completion status must be returned. For an endpoint, when this bit is Set, the PCI Express Function is allowed to issue Memory or I/O Requests. When this bit is Clear, the PCI Express Function is not allowed to issue any Memory or I/O Requests.
1	0b RW	**Memory Space Enable (MSE):** When set, memory cycles within the range specified by the memory base and limit registers can be forwarded to the PCI-Express device. When cleared, these memory cycles are responded to on the primary interface with an unsupported request completion.
0	0b RW	**I/O Space Enable (IOSE):** When set, I/O cycles within the range specified by the I/O base and limit registers can be forwarded to the PCI-Express device. When cleared, these cycles are responded to on the primary interface with an unsupported request completion.

22.9.3 Revision ID;Class Code (RID_CC)—Offset 8h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 06040000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
BCC								SCC								PI								RID								

Bit Range	Default and Access	Field Name (ID): Description
31:24	06h RO	**Base Class Code (BCC):** Indicates the device is a bridge device.
23:16	04h RO/V	**Sub-Class Code (SCC):** The default indicates the device is a PCI-to-PCI bridge. If the MPC.BT register is set to a '1' for a Host Bridge, this register reads 00h.
15:8	00h RO/V	**Programming Interface (PI):** This bit indicates that this device is standard (non-subtractive) PCI-to-PCI Bridge.
7:0	00h RO/V	**Revision ID (RID):** Indicates the revision of the function.

22.9.4 Cache Line Size; Primary Latency Timer; Header Type (CLS_PLT_HTYPE)—Offset Ch

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00010000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD								MFD	HTYPE							CT					RSVD				LS							

Bit Range	Default and Access	Field Name (ID): Description
31:24	0h Rv	**Reserved.**
23	0b RO	**Multi-function Device (MFD):** This bit is '0' to indicate a single function device.
22:16	1h RO/V	**Header Type (HTYPE):** The default mode identifies the header layout of the configuration space, which is a PCI-to-PCI bridge. If the MPC.BT register is set to a '1' for a Host Bridge, this register reads 00h.
15:11	0h RO	**Latency Count (CT):** Reserved per *PCI Express Specification*.
10:8	0h Rv	**Reserved**.
7:0	0h RW	**Line Size (LS):** This is read/write but contains no functionality, per *PCI Express Specification*.

22.9.5 Express Port Memory Base Address Register (EXPPTMBAR_UX8) —Offset 10h

Type: CFG Register
(Size: 64 bits)

Device: 0
Function: 0

Default: 0000000000000000h

63			60				56				52				48				44				40				36				32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BADDR																															

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BADDR															RSVD														PREF	TYPE		MIS

Bit Range	Default and Access	Field Name (ID): Description
63:17	00000000 0000h RW	**Base Address (BADDR):**
16:4	000h RV	**Reserved**
3	0b RO	**Prefetchable Memory (Pref)**: Not prefetchable memory space.
2:1	10b RO	**Type (type):** The memory mapped space can be placed anywhere in the 64-bit addressable region of the device
0	0b RO	**Memory Space Indicator (MSI):** 0b = Memory space, 1b - I/O space. Indicates the bus number of the backbone.

22.9.6 Bus Numbers (BusNUM)—Offset 18h

Access Method

Type: CFG Register
(Size: 24bits)

Device: 0
Function: 0

Default: 000000h

			20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SBBN								SCBN								PBN							

Bit Range	Default and Access	Field Name (ID): Description
23:16	0h RW	**Subordinate Bus Number (SBBN):** Indicates the highest PCI bus number below the bridge.
15:8	0h RW	**Secondary Bus Number (SCBN):** Indicates the bus number secondary interface.
7:0	0h RW	**Primary Bus Number (PBN):** Indicates the PCI Express bus number

22.9.7 I/O Base and Limit; Secondary Status (IOBL_SSTS)—Offset 1Ch

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DPE	RSE	RMA	RTA	STA	DTS		MDPD	FBC	RSVD	SC66	RSVD				IOLA				IOLC				IOBA				IOBC				

Bit Range	Default and Access	Field Name (ID): Description
31	0h RW1C/V	**Detected Parity Error (DPE):** Set when the port receives a poisoned TLP.
30	0h RW1C/V	**Received System Error (RSE):** Set when the port receives an ERR_FATAL or ERR_NONFATAL message from the device.
29	0h RW1C/V	**Received Master Abort (RMA):** Set when the port receives a completion with 'Unsupported Request' status from the device.
28	0h RW1C/V	**Received Target Abort (RTA):** Set when the port receives a completion with 'Completion Abort' status from the device.
27	0h RW1C/V	**Signaled Target Abort (STA):** Set when the port generates a completion with 'Completion Abort' status to the device.
26:25	0h RO/V	**DEVSEL# Timing Status (DVT):** Reserved per *PCI Express Specification*.
24	0b RW1C/V	**Master Data Parity Error Detected (MDPD):** Set when the BCTRL.PERE, and either of the following two conditions occurs: Port receives completion marked poisoned. Port poisons a write request to the secondary side.
23	0b RO/V	**Fast Back to Back Capable (FBC):** Reserved per *PCI Express Specification*.
22	0b RV	**Reserved.**
21	0b RO	**Secondary 66 MHz Capable (SC66):** Reserved per *PCI Express Specification*.
20:16	0h RV	**Reserved.**

Bit Range	Default and Access	Field Name (ID): Description
15:12	0h RW	**I/O Address Limit (IOLA):** I/O Base bits corresponding to address lines 15:12 for 4KB alignment. Bits 11:0 are assumed to be padded to FFFh.
11:8	0h RO	**I/O Limit Address Capability (IOLC):** Indicates that the bridge does not support 32-bit I/O addressing.
7:4	0h RW	**I/O Base Address (IOBA):** I/O Base bits corresponding to address lines 15:12 for 4KB alignment. Bits 11:0 are assumed to be padded to 000h.
3:0	0h RO	**I/O Base Address Capability (IOBC):** Indicates that the bridge does not support 32-bit I/O addressing.

22.9.8 Memory Base and Limit (MBL)—Offset 20h

Accesses that are within the ranges specified in this register will be sent to the attached device if CMD.MSE is set. Accesses from the attached device that are outside the ranges specified will be forwarded to the backbone if CMD.BME is set. The comparison performed is MB [gt]= AD[lb]31:20[rb] [lt]= ML.

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ML												RSVD				MB												RSVD				

Bit Range	Default and Access	Field Name (ID): Description
31:20	0h RW	**Memory Limit (ML):** These bits are compared with bits [31:20] of the incoming address to determine the upper 1MB aligned value of the range.
19:16	0h RV	**Reserved.**
15:4	0h RW	**Memory Base (MB):** These bits are compared with bits [31:20] of the incoming address to determine the lower 1MB aligned value of the range.
3:0	0h RV	**Reserved**.

22.9.9 Prefetchable Memory Base and Limit (PMBL)—Offset 24h

Accesses that are within the ranges specified in this register will be sent to the device if CMD.MSE is set. Accesses from the device that are outside the ranges specified will be forwarded to the backbone if CMD.BME is set. The comparison performed is PMBU32:PMB [gt]= AD[lb]63:32[rb]:AD[lb]31:20[rb] [lt]= PMLU32:PML.

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00010001h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
PML																I64L				PMB									I64B			

Bit Range	Default and Access	Field Name (ID): Description
31:20	0h RW	**Prefetchable Memory Limit (PML):** These bits are compared with bits [31:20] of the incoming address to determine the upper 1 MB aligned value of the range.
19:16	1h RO	**64-bit Indicator (I64L):** Indicates support for 64-bit addressing.
15:4	0h RW	**Prefetchable Memory Base (PMB):** These bits are compared with bits [31:20] of the incoming address to determine the lower 1 MB aligned value of the range.
3:0	1h RO	**64-bit Indicator (I64B):** Indicates support for 64-bit addressing.

22.9.10 Prefetchable Memory Base Upper 32 Bits (PMBU32)—Offset 28h

Size:32 bits

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PMBU																															

Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Prefetchable Memory Base Upper Portion (PMBU):** Upper 32-bits of the prefetchable address base.

22.9.11 Prefetchable Memory Limit Upper 32 Bits (PMLU32)—Offset 2Ch

Size:32 bits

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PMLU																															

Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Prefetchable Memory Limit Upper Portion (PMLU):** Upper 32-bits of the prefetchable address limit.

22.9.12 Capabilities List Pointer (CAPP)—Offset 34h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000040h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
RSVD																								PTR							

Bit Range	Default and Access	Field Name (ID): Description
31:8	0h RV	**Reserved**.
7:0	40h RO	**Capabilities Pointer (PTR):** Indicates that the pointer for the first entry in the capabilities list. BIOS can determine which capabilities will be exposed by including or removing them from the capability linked list.

22.9.13 Expansion ROM Base Address Register (EXPROM))—Offset 38h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BADDR												RSVD																			EXPROMEN

Bit Range	Default and Access	Field Name (ID): Description
31:20	0h RW	**Base Address (BADDR)**: This marks the base address for the 1 MB expansion ROM for the endpoint device. Firmware should initialize this region prior to PCIe enumeration.
19:1	0h RV	Reserved
0	0b RW	**Expansion ROM Enable (EXPROMEN):** This field enables the expansion ROM base address to be used by the device. 0: Disables access to Expansion ROM 1: Enables Access to Expansion ROM.

22.9.14 Interrupt Information; Bridge Control (INTR_BCTRL)—Offset 3Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000100h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
RSVD				DTSE	DTS	SDT	PDT	FBE	SBR	MAM	V16	VE	IE	SE	PERE	IPIN								ILINE								

Bit Range	Default and Access	Field Name (ID): Description
31:28	0h RV	**Reserved.**
27	0h RO	**Discard Timer SERR# Enable (DTSE):** Reserved per *PCI Express Specification*.
26	0h RO	**Discard Timer Status (DTS):** Reserved per *PCI Express Specification*.
25	0h RO	**Secondary Discard Timer (SDT):** Reserved per *PCI Express Specification*.
24	0h RO	**Primary Discard Timer (PDT):** Reserved per *PCI Express Specification*.
23	0h RO	**Fast Back to Back Enable (FBE):** Reserved per *PCI Express Specification*.
22	0h RW	**Secondary Bus Reset (SBR):** Triggers a Hot Reset on the PCI-Express port.
21	0h RO	**Master Abort Mode (MAM):** Reserved per *PCI Express Specification*.
20	0h RW	**VGA 16-Bit Decode (V16):** When set, indicates that the I/O aliases of the VGA range (see BCTRL:VE definition below), are not enabled. 0: Execute 10-bit address decode on VGA I/O accesses. 1: Execute 16-bit address decode on VGA I/O accesses.
19	0h RW	**VGA Enable (VE):** When set to 1b, the following ranges will be forwarded from the primary to the secondary side of the bridge regardless of the value of the I/O base and limit registers. Memory ranges A0000h-BFFFFh I/O ranges 3B0h 3BBh and 3C0h 3DFh, and all aliases of bits [15:10] in any combination of 1's
18	0h RW	**ISA Enable (IE):** This bit only applies to I/O addresses that are enabled by the I/O Base and I/O Limit registers and are in the first 64KB of PCI I/O space. If this bit is set, the PCIe bridge will block any forwarding from the primary to the secondary sided I/O transactions addressing the last 768 bytes in each 1KB block (offsets 100h to 3FFh). In the opposite direction (secondary to primary) I/O transactions will be forwarded if they address the last 768B in each 1KB block. 1: Forward upstream ISA I/O addresses in the address range defined by the I/O Base and I/O Limit registers that are in the firsts 64KB of PCI I/O address space (Top 768B of each 1K block). 0: Forward downstream all I/O addresses in the address range defined by the I/O Base and I/O Limit registers.
17	0h RW	**SERR# Enable (SE):** This bit controls the forwarding of ERR_COR, ERR_NONFATAL AND ERR_FATAL messages to the primary side of the PCIe port. 1: Enables forwarding of ERR_COR, ERR_NONFATA, ERR_FATAL messages. 0: Disables forwarding of ERR_COR, ERR_NONFATA, ERR_FATAL messages.
16	0h RW	**Parity Error Response Enable (PERE):** This bit controls the response to poisoned TLPs in the PCI Express port. 1: Enables reporting of poisoned TLP errors. 0: Disables reporting of poisoned TLP errors.
15:8	01h RW-L	**Interrupt Pin (IPIN):** This register tells which interrupt pin the function uses. 01h: Generate INTA 02h: Generate INTB 03h: Generate INTC 04h: Generate INTD Others: Reserved BIOS has the ability to write this register during boot to setup the correct interrupt for the function. After it is locked this register can not be updated until another PLTRST#
7:0	0h RW	**Interrupt Line (ILINE):** Software written value to indicate which interrupt line (vector) the interrupt is connected to. No hardware action is taken on this register.

22.9.15 Capabilities List; PCI Express Capabilities (CLIST_XCAP)—Offset 40h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00528010h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0
RSVD			IMN				SI	DT				CV				NEXT								CID								

Bit Range	Default and Access	Field Name (ID): Description
31:30	0h RO	Reserved.
29:25	0h RO	**Interrupt Message Number (IMN):** This field indicates the interrupt message number that is generated from the PCI Express port. When there is more than one MSI interrupt number, this register is required to contain the offset between the base Message Data and the MSI Message that is generated when the status bits in the slot status register or root port status registers are set. The chipset is required to update this field if the number of MSI messages change
24	0h RO	**Slot Implemented (SI):** Hardwired to 0b
23:20	5h RO	**Device / Port Type (DT):** Indicates this the upstream port of a PCIe switch
19:16	2h RO	**Capability Version (CV):** Version 2.0 indicates devices compliant to the PCI Express 2.0 specification which incorporates the Register Expansion ECN.
15:8	80h RO	**Next Capability (NEXT):** Indicates the location of the next capability.
7:0	10h RO	**Capability ID (CID):** Indicates this is a PCI Express capability

22.9.16 Device Capabilities (DCAP)—Offset 44h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00008002h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
RSVD			FLRC	CSPS			CSPV								RSVD	RBER	RSVD			E1AL			E0AL			ETFS	PFS		MPS		

Bit Range	Default and Access	Field Name (ID): Description
31:29	0h RV	**Reserved**.
28	0h RO	**Function Level Reset Capable (FLRC):** Reserved, not applicable.
27:26	0h RO	**Captured Slot Power Limit Scale (CSPS):** This value is set by the set_slot_power_limit message.
25:18	0h RO	**Captured Slot Power Limit Value (CSPV):** This value is set by the set_slot_power_limit message
17:16	0h RV	**Reserved.**
15	1h RO	**Role Based Error Reporting (RBER):** The PCIe cluster supports Role-based Error reporting.
14:12	0h RV	**Reserved.**
11:9	0h RO	**Endpoint L1 Acceptable Latency (E1AL):** Reserved
8:6	0h RO	**Endpoint L0 Acceptable Latency (E0AL):** Reserved
5	0h RO	**Extended Tag Field Supported (ETFS):** 0: The PCIe supports 5-bit extended tag 1: The PCIe supports 8-bit extended tag.
4:3	0h RO	**Phantom Functions Supported (PFS):** No phantom functions supported
2:0	2h RWL	**Max Payload Size Supported (MPS):** BIOS should write to this field during system initialization. The Max Payload Size supported is 512 bytes. 000b: 128 bytes max payload size. 001b: 256 bytes max payload size. 010b: 512 bytes max payload size. 011b: 1024 bytes max payload size. 100b: 2048 bytes max payload size. 101b: 4096 bytes max payload size. 110b: Reserved. 111b: Reserved.

22.9.17 Device Control; Device Status (DCTL_DSTS)—Offset 48h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00002000

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD										TDP	APD	URD	FED	NFED	CED	RSVD	MRRS			ENS	APME	PFE	ETFE	MPS			ERO	URE	FEE	NFE	CEE	

Bit Range	Default and Access	Field Name (ID): Description
31:22	0h RV	Reserved. Software must always write a "0" to these bits.
21	0b RO-V	**Transactions Pending (TDP):** Not supported.
20	0b RO	**AUX Power Detected (APD):** Auxiliary power is not supported
19	0b RW1C/V	**Unsupported Request Detected (URD):** Indicates an unsupported request was detected from the PCI Express link.
18	0b RW1C/V	**Fatal Error Detected (FED):** Indicates a fatal error was detected. Errors are logged in this register regardless of whether error reporting is enabled or not in the Device Control Register
17	0b RW1C/V	**Non-Fatal Error Detected (NFED):** Indicates a non-fatal error was detected. Errors are logged in this register regardless of whether error reporting is enabled or not in the Device Control Register
16	0b RW1C/V	**Correctable Error Detected (CED):** Indicates a correctable error was detected. Errors are logged in this register regardless of whether error reporting is enabled or not in the Device Control Register
15	0b RV	Reserved.
14:12	2h RW	**Max Read Request Size (MRRS):** This field sets the maximum Read Requests size of the function as a requester. The Function must not generate read requests with size exceeding the set value. 000b: 128 bytes maximum Read Request size 001b: 256 bytes maximum Read Request size 010b: 512 bytes maximum Read Request size 011b: 1024 bytes maximum Read Request size 100b: 2048 bytes maximum Read Request size 101b: 4096 bytes maximum Read Request size Others: Reserved
11	0b RO	**Enable No Snoop (ENS):** Not supported.
10	0b RO	**Aux Power PM Enable (APME):** Not supported.
9	0b RO	**Phantom Functions Enable (PFE):** Not supported
8	0b RO	**Extended Tag Field Enable (ETFE):** Not supported
7:5	0h RW	**Max Payload Size (MPS):** This field sets maximum TLP payload size for the function. As a receiver, the function must handle TLPs as large as the set value. As a Transmitter, the function must not generate TLPs exceeding the set value. 000b: 128 bytes maximum payload size (default) 001b: 256 bytes maximum payload size 010b: 512 bytes maximum payload size 011b: 1024 bytes maximum payload size 100b: 2048 bytes maximum payload size 101b: 4096 bytes maximum payload size Others: Reserved
4	0b RO	**Enable Relaxed Ordering (ERO):** Not supported
3	0b RW	**Unsupported Request Reporting Enable (URE):** This bit controls the enabling of ERR_CORR, ERR_NONFATAL or ERR_FATAL messages on PCI Express for reporting "Unsupported Request" errors.

Bit Range	Default and Access	Field Name (ID): Description
2	0b RW	**Fatal Error Reporting Enable (FEE):.**When this bit is set, generation of the ERR_FATAL message is enabled
1	0b RW	**Non-Fatal Error Reporting Enable (NFE):** When this bit is set, generation of the ERR_NONFATAL message is enabled.
0	0b RW	**Correctable Error Reporting Enable (CEE):** When this bit is set, generation of the ERR_CORR message is enabled.

22.9.18 Link Capabilities (LCAP)—Offset 4Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 0042C883h

31			28				24				20				16			12				8				4				0
0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	1	0	0	1	0	0	0	1	0	0	0	0	0	1	1
PN								RSVD	ASPMOC	LBNC	LARC	SDERC	CPM	EL1			EL0			APMS		MLW				MLS				

Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RW-L	**Port Number (PN):** Indicates the port number assigned to the uplink
23	0b RO	Reserved.
22	1b RW/O	**ASPM Optionality Compliance (ASPMOC):** ASPM Optionality Compliance(ASPMOC): This bit must be set to 1b for PCIe 3.0 compliant port. Components implemented against certain earlier versions of this specification will have this bit set to 0b. Software is permitted to use the value of this bit to help determine whether to enable ASPM or whether to run ASPM compliance tests.
21	0b RO	**Link Bandwidth Notification Capability (LBNC):** reserved, not supported
20	0b RO	**Link Active Reporting Capable (LARC):** reserved, not supported
19	0b RO	**Surprise Down Error Reporting Capable (SDERC):** Set to '0' to indicate the uplink does not support Surprise Down Error Reporting
18	0b RO	**Clock Power Management (CPM):** '0' Indicates that the upstream port does not support the CLKREQ# mechanism.
17:15	5h RW-L	**L1 Exit Latency (EL1):** Indicates an exit latency of 2 us to 4 us. 000b Less than 1 us 001b 1 us to less than 2 us 010b 2 us to less than 4 us 011b 4 us to less than 8 us 100b 8 us to less than 16 us 101b 16 us to less than 32 us 110b 32 us to 64 us 111b More than 64 us

Bit Range	Default and Access	Field Name (ID): Description
14:12	4h RW-L	**L0s Exit Latency (EL0):** This field indicates the L0s exit latency for the given PCI-Express Link. It indicates the length of time this port requires to complete transition from L0s to L0. 000b: Less than 64 ns 001b: 64 ns to less than 128 ns 010b: 128 ns to less than 256 ns 011b: 256 ns to less than 512 ns 100b 512 ns to less than 1 µs 101b: 1 us to less than 2 us 110b: 2 us to less than 4 us 111b: More than 4 us
11:10	2h RW-L	**Active State Link PM Support (APMS):** Indicates the level of active state power management on this link Bits Definition 00 No ASPM Supported 01 L0s Supported 10 L1 Supported 11 L0s and L1 supported ***Note:*** L0s should not be advertised when operating with SRIS.
9:4	08h RO/V	**Maximum Link Width (MLW):** This field indicates the maximum link width implemented by the given PCI Express Link. 00h: Reserved 01h: x1 02h: x2 04h: x4 08h: x8 10h: x16 20h: x32 (Unsupported) Others Reserved ***Note:*** Default value is determined by HW after port training is initiated for ports that bifurcate. Hence effective MLW can be smaller than indicated by default.
3:0	3h RO/V	**Max Link Speeds (MLS):** Indicates the supported link speeds of the Uplink Port. This field indicates the supported link speed(s) of the associated port. 0001b: 2.5 Gb/s link speed is supported 0010b: 5.0 Gb/s and 2.5 Gb/s link speed supported 0011b: 8.0 Gb/s and 5.0 and 2.5 Gb/s link speed supported Others: Reserved. ***Note:*** The encoding is the binary value of the bit location in the Supported Link Speeds Vector (in the Link Capabilities 2 register) that corresponds to the maximum Link speed.

22.9.19 Link Control; Link Status (LCTL_LSTS)—Offset 50h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00010000h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
LABS	LBMS	LA	SCC	LT	RSVD	NLW						CLS				RSVD				LABIE	LBMIE	HAWD	ECPM	ES	CCC	RL	LD	RCBC	RSVD	ASPM	

Bit Range	Default and Access	Field Name (ID): Description
31	0b RO	**Link Autonomous Bandwidth Status (LABS):** This bit is not applicable and hardwired to 0b
30	0b RO	**Link Bandwidth Management Status (LBMS):** This bit is not applicable and hardwired to 0b
29	0h RO/V	**Link Active (LA):** Set to 1b when the Data Link Control and Management State Machine is in the DL_Active state, 0b otherwise.
28	1b RW-L	**Slot Clock Configuration (SCC):** When to 1b, this bit indicates that the component uses the same physical reference clock that he platform provides on the connector. If the device uses an independent clock irrespective of the presence of a reference on the connector, this bit must be clear. 1: Indicates same reference clock. 0: Indicates independent reference clock
27	0b RO	**Link Training (LT):** This bit is not applicable for an upstream port and is hardwired to 0
26	0h RO	Reserved.
25:20	0h RO/V	**Negotiated Link Width (NLW):** This field indicates the negotiated width of the PCI Express link. 00 0001b: x1 00 0010b: X2 00 0100b: x4 00 1000b: X8 00 1100b: X12—not supported 01 0000b: X16 10 0000b: X32—not supported All other values are reserved. ***Note:*** The value in this field is undefined when the link is not up.
19:16	1b RO/V	**Current Link Speed (CLS):** This field indicates the negotiated link speed of the given PCI Express link. 0001b: 2.5 Gb/s PCI Express Link 0010b: 5.0 Gb/s PCI Express Link 0011b: 8.0 Gb/s PCI Express Link Others: Reserved ***Note:*** The encoding is the binary value of the bit location in the Supported Link Speeds Vector (in the Link Capabilities 2 register) that corresponds to the current Link speed. ***Note:*** The value in this field is undefined when the link is not up.
15:12	0h RO	**Reserved.**
11	0b RO	**Link Autonomous Bandwidth Interrupt Enable (LABIE):** this is not applicable so hardwired to 0b
10	0b RO	**Link Bandwidth Management Interrupt Enable (LBMIE):** The uplink does not implement the Link Bandwidth Notification Capability so this is hardwired to 0.
9	0h RW	**Hardware Autonomous Width Disable (HAWD):** When Set, this bit disables hardware from changing the Link width for reasons other than attempting to correct unreliable Link operation by reducing Link width.
8	0b RO	**Enable Clock Power Management (ECPM):** Reserved. Not supported by the upstream port.
7	0b RW	**Extended Synch (ES):** When set, forces extended transmission of FTS ordered sets in FTS and extra TS2 at exit from L1 prior to entering L0.
6	0b RW	**Common Clock Configuration (CCC):** When set, indicates that the root port and device are operating with a distributed common reference clock. A value of 0b indicates that this component and the component at the opposite end of this link are operating with asynchronous reference clock. After changing the value in this bit in both components on a link, software must trigger the link to retrain by writing a 1b to the Retrain Link bit of the downstream port.

Bit Range	Default and Access	Field Name (ID): Description
5	0b RO	**Retrain Link (RL):** Read only of 0b for an upstream port.
4	0b RO	**Link Disable (LD):** Read only of 0b for an upstream port.
3	0b RO	**Read Completion Boundary Control (RCBC):** Indicates the read completion boundary is 64 bytes.
2	0b RO	**Reserved**.
1:0	00b RW	**Active State Link PM Control (ASPM):** Indicates whether the upstream port should enter L0s or L1 or both. Bits Definition 00 Disabled 01 L0s Entry Enabled 10 L1 Entry Enabled 11 L0s and L1 Entry Enabled

22.9.20 Device Capabilities 2 (DCAP2)—Offset 64h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	1	1
RSVD												OBFFS		RSVD				TPHCS	LTRMS	NROEPRPASS	AD128ACS	AD64ACS	AD32ACS	ARS	AFS	CTDS	CTRS				

Bit Range	Default and Access	Field Name (ID): Description
31:20	0h RO	**Reserved.**
19:18	00b RO	**Optimized Buffer Flush/Fill Supported (OBFFS):** 00b - OBFF is not supported.
17:14	0h RO	**Reserved**.
13:12	00b RO	**TPH Completer Supported (TPHCS):** 00: TPH and Extended TPH Completer not supported
11	0b RW-L	**LTR Mechanism Supported (LTRMS):** A value of 1b indicates support for the optional Latency Tolerance Reporting (LTR) mechanism capability.
10	0b RO	**No RO-enabled PR-PR Passing (NROEPRPASS):** Set to 0b as this feature is not supported.
9	0b RO	**AD128 CAS Completer Supported (AD128ACS):** Not supported

Bit Range	Default and Access	Field Name (ID): Description
8	0b RO	**AD64-bit AtomicOP Completer Supported (AD64ACS):** Not supported
7	0b RO	**AD32 bit AtomicOP Completere Supported (AD32ACS):** not supported
6	0b RW-L	**AtomicOP Routing Supported (ARS)** Not supported. Leave at 0b
5	0b RO	**ARI Forwarding Supported (AFS):** Not Supported
4	0b RO	**Completion Timeout Disable Supported (CTDS):** A value of 1b indicates support for the Completion Timeout Disable mechanism.
3:0	0h RO	**Completion Timeout Ranges Supported (CTRS):** Not applicable to an upstream port, set to 0h

22.9.21 Device Control 2; Device Status 2 (DCTL2_DSTS2)—Offset 68h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																		OBFFEN		RSVD		LTREN	IDOCE	IDORE	AEB	ARE	AFE	CTD	CTV			

Bit Range	Default and Access	Field Name (ID): Description
31:15	0h RO	**Reserved**.
14:13	00b RO	**Optimized Buffer Flush/Fill Enable (OBFFEN):** Not supported so RO.
12:11	00b RO	**Reserved.**
10	0b RW	**LTR Mechanism Enable (LTREN):** Not supported
9	0b RO	**IDO Completion Enable (IDOCE):**. Not supported
8	0b RO	**IDO Request Enable (IDORE):** Not supported
7	0b RO	**AtomicOp Egress Blocking (AEB):** Not supported
6	0b RO	**AtomicOp Requester Enable (ARE):** Not supported

Bit Range	Default and Access	Field Name (ID): Description
5	0b RO	**ARI Forwarding Enable (AFE):** Not applicable
4	0b RO	**Completion Timeout Disable (CTD):** Not supported
3:0	0h RO	**Completion Timeout Value (CTV):** Not supported

22.9.22 Link Capabilities 2 (LCAP2)—Offset 6Ch

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 0000000Eh

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0
RSVD											LSOSRSS								LSOSGSSV				CS				SLSV				RSVD	

Bit Range	Default and Access	Field Name (ID): Description
31:23	0h RO	Reserved.
22:16	0h RW-L	**Lower SKP OS Reception Supported Speeds Vector (LSOSRSS):** If this field is non-zero, it indicates that the Port, when operating at the indicated speed(s) supports SRIS and also supports receiving SKP OS at the rate defined for SRNS while running in SRIS. Bit definitions within this field are: Bit 0 2.5 GT/s Bit 1 5.0 GT/s Bit 2 8.0 GT/s Bits 6:3 RsvdP ***Note:*** Currently, design does not support reception of low frequency SOS in L0.
15:9	0h RW-L	**Lower SKP OS Generation Supported Speeds Vector (LSOSGSSV):** If this field is non-zero, it indicates that the Port, when operating at the indicated speed(s) supports SRIS and also supports software control of the SKP Ordered Set transmission scheduling rate. Bit definitions within this field are: Bit 0 2.5 GT/s Bit 1 5.0 GT/s Bit 2 8.0 GT/s Bits 6:3 RsvdP

Bit Range	Default and Access	Field Name (ID): Description
8	0b RW-L	**Crosslink Supported (CS):** Not supported
7:1	07h RO/V	**Supported Link Speeds Vector (SLSV):** This field indicates the supported Link speed(s) of the associated Port. For each bit, a value of 1b indicates that the corresponding Link speed is supported; otherwise, the Link speed is not supported. Bit definitions are: bit 1 2.5 GT/s bit 2 5.0 GT/s bit 3 8.0 GT/s Others: Reserved
0	0h RO	Reserved.

22.9.23 Link Control 2; Link Status 2 (LCTL2_LSTS2)—Offset 70h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000003h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
RSVD										LER	EQP3S	EQP2S	EQP1S	EqC	CDL	CD				CSOS	EMC	TM			SD	HASD	EC	TLS				

Bit Range	Default and Access	Field Name (ID): Description
31:22	0h RO	Reserved.
21	0b RW1CS	**Link Equalization Request (LER):** This bit is set by hardware to request the Link equalization process to be performed on the Link.
20	0b ROS-V	**Equalization Phase 3 Successful (EQP3S):** When set to 1, this bit indicates that Phase 3 of the Transmitter Equalization procedure has successfully completed.
19	0b ROS-V	**Equalization Phase 2 Successful (EQP2S):** When set to 1, this bit indicates that Phase 2 of the Transmitter Equalization procedure has successfully completed.
18	0b ROS-V	**Equalization Phase 1 Successful (EQP1S):** When set to 1, this bit indicates that Phase 1 of the Transmitter Equalization procedure has successfully completed.
17	0b ROS-V	**Equalization Complete (EqC):** When set to 1, this bit indicates that the Transmitter Equalization procedure has completed
16	0b RO V	**Current De-emphasis Level (CDL):** When the Link is operating at 5.0 GT/s speed, this bit reflects the level of de-emphasis. Encodings: 1b -3.5 dB 0b -6 dB The value in this bit is undefined when the Link is not operating at 5.0 GT/s speed.

Bit Range	Default and Access	Field Name (ID): Description
15:12	0h RWS	**Compliance Preset/De-emphasis (CD):** For 8 GT/s and 5 GT/s: This bit sets the transmitter preset level forPolling.Compliance state if the entry occurred due to the Enter Compliance bit being 1b. The Encoding are defined as follows: 8 GT/s Rate: 0000b: -6 dB for de-emphasis, 0 dB for preshoot 0001b: -3.5 dB for de-emphasis, 0 dB for preshoot 0010b: -4.5 dB for de-emphasis, 0 dB for preshoot 0011b: -2.5 dB for de-emphasis, 0 dB for preshoot 0100b: 0 dB for de-emphasis, 0 dB for preshoot 0101b: 0 dB for de-emphasis, 2 dB for preshoot 0110b: 0 dB for de-emphasis, 2.5 dB for preshoot 0111b: -6 dB for de-emphasis, 3.5 dB for preshoot 1000b: -3.5 dB for de-emphasis, 3.5 dB for preshoot 1001b: 0 dB for de-emphasis, 3.5 dB for preshoot 1010b: The Transmitter Preset encoding of 1010b corresponds to the maximum Vtx-boost that the Transmitter advertises with FS and LF in both full swing and reduced swing modes with C_{-1} = 0 and C_{+1} = FS – (2*LF). All other encodings are reserved. 5 GT/s Rate: 001b: -3.5dB 000b: -6 dB All other are reserved. When the link is operating at 2.5 Gb/s, the setting of this bit has no effect.
11	0h RWS	**Compliance SOS (CSOS):** When set to 1b, the LTSSM is required to send SKP Ordered Sets periodically in between the (modified) compliance patterns. The default value of this bit is 0b.
10	0b RWS	**Enter Modified Compliance (EMC):** When this bit is set to 1b, the device transmits Modified Compliance Pattern if the LTSSM enters Polling.Compliance substate. Default value of this bit is 0b. This register is intended for debug, compliance testing purposes only. System firmware and software is allowed to modify this register only during debug or compliance testing. In all other cases, the system must ensure that this register is set to the default value.
9:7	0h RWS-V	**Transmit Margin (TM):** This field controls the value of the non-de-emphasized voltage level at the transmitter pins. This field is reset to 000b on entry to the LTSSM Polling.Configuration substate. 000b: Normal operating range 001b: 800-1200mV for full swing and 400-700mV for half-swing Others: Reserved This register is intended for debug, compliance testing purposes only. System firmware and software is allowed to modify this register only during debug or compliance testing. In all other cases, the system must ensure that this register is set to the default value.
6	0b RO	**Selectable De-emphasis (SD):** Not supported.

Bit Range	Default and Access	Field Name (ID): Description
5	0b RWS	**Hardware Autonomous Speed Disable (HASD):** When set, this bit disables hardware from changing the link speed for device specific reasons other than attempting to correct unreliable link operation by reducing link speed for device-specific reasons other than attempting to correct unreliable link operations by reducing link speed. Initial transition to the highest supported common link speed is not blocked by this bit.
4	0b RWS-V	**Enter Compliance (EC):** Software is permitted to force a Link to enter Compliance mode at the speed indicated in the Target Link Speed field by setting this bit to 1b in both components on a Link and then initiating a hot reset on the Link. Default value of this bit following Fundamental Reset is 0b. This bit is intended for debug, compliance testing purposes only. System firmware and software is allowed to modify this bit only during debug or compliance testing. In all other cases, the system must ensure that this bit is set to the default value. This bit is intended for debug, compliance testing purposes only. System firmware and software is allowed to modify this bit only during debug or compliance testing. In all other cases, the system must ensure that this bit is set to the default value.
3:0	3h RWS-V	**Target Link Speed (TLS):** For downstream ports and root ports, this field sets an upper limit on link operational speed by restricting the values advertised by the Downstream Port in its training sequences. 0001b: 2.5 Gb/s Target Link Speed 0010b: 5.0 Gb/s Target Link Speed 0011b: 8.0 Gb/s Target Link Speed Others: Reserved ***Note:*** The encoding is the binary value of the bit in the Supported Link Speeds Vector (in the Link Capabilities 2 register) that corresponds to the desired target Link speed. All other encodings are reserved If a value is written to this field that does not correspond to a speed included in the Supported Link Speeds field, the result is undefined. ***Note:*** The default value of this field is the highest Link speed supported by the component (as reported in the Max Link Speed field of the Link Capabilities register) unless the corresponding platform/form factor requires a different default value. For both Upstream and Downstream Ports, this field is used to set the target compliance mode speed when software is using the Enter Compliance bit to force a Link into compliance mode.

22.9.24 Power Management Capability; PCI Power Management Capabilities (PMCAP_PMC)—Offset 80h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: C8038801h

31			28				24				20				16				12				8				4				0
1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	1
PMES					D2S	D1S	AC			DSI	RSVD	PMEC	VS				NEXT								CID							

Bit Range	Default and Access	Field Name (ID): Description
31:27	19h RO	**PME Support (PMES):** Indicates PME# is supported for states D0, $D3_{HOT}$ and $D3_{COLD}$. There is no PME support for $D3_{COLD}$ as the part does not have the capability, but for PCIe compliance purposes, bit 15is set.
26	0b RO	**D2_Support (D2S):** The D2 state is not supported.
25	0b RO	**D1_Support (D1S):** The D1 state is not supported.
24:22	0h RO	**Aux_Current (AC):** Reports 0 mA (self-powered).
21	0b RO	**Device Specific Initialization (DSI):** Indicates that no device-specific initialization is required.
20	0h RO	**Reserved.**
19	0h RO	**PME Clock (PMEC):** Indicates that PCI clock is not required to generate PME#.
18:16	3h RO	**Version (VS):** Indicates support for Revision 1.2 of the *PCI Power Management Specification*.
15:8	88h RO	**Next Capability (NEXT):** Contains the offset of the next item in the Capabilities List.
7:0	1h RO	**Capability Identifier (CID):** Value of 01h indicates this is a PCI power management capability.

22.9.25 PCI Power Management Control and Status (PMCS)—Offset 84h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000008h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
RSVD								BPCE	B23S	RSVD						PMES	DSC	DSEL				PMEE	RSVD				NSR	RSVD	PS		

Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	**Reserved**
23	0b RO	**Bus Power / Clock Control Enable (BPCE):** Reserved per *PCI Express Specification*.
22	0b RO	**B2/B3 Support (B23S):** Reserved per *PCI Express Specification*.
21:16	0h RO	**Reserved.**
15	0b RW1CS	**PME Status (PMES):** Indicates a PME was received.

Bit Range	Default and Access	Field Name (ID): Description
14:13	0h RO	**Data Scale (DSC):** Reserved
12:9	0h RO	**Data Select (DSEL):** Reserved
8	0b RWS	**PME Enable (PMEE):** Gates assertion of the PME message.
7:4	0h RO	**Reserved.**
3	1b RW/O	**No Soft Reset (NSR):** When set to 1 this bit indicates that devices transitioning from $D3_{HOT}$ to D0 because of Power State commands do not perform an internal reset. Configuration context is preserved. Upon transition from $D3_{HOT}$ to D0 Initialized state, no additional operating system intervention is required to preserve Configuration Context beyond writing the Power State bits. When clear, devices do perform an internal reset upon transitioning from $D3_{HOT}$ to D0 via software control of the Power State bits. Configuration Context is lost when performing the soft reset. Upon transition from $D3_{HOT}$ to D0 state, full reinitialization sequence is needed to return the device to D0 Initialized.
2	0b RO	**Reserved.**
1:0	0h RW	**Power State (PS):** This field is used both to determine the current power state of the upstream port and to set a new power state. The values are: 00 D0 state 11 $D3_{HOT}$ state If software attempts to write a '10' or '01' to these bits, the write will be ignored.

22.9.26 Subsystem Capability List Register (SVCAP)—Offset 88h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 0000000Dh

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1
RSVD																NEXT												CID			

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:8	0h RO	**Next Capability (NEXT):** For upstream ports there is no MSI structure so the capability is terminated
7:0	Dh RO	**Capability Identifier (CID):** Value of 0Dh indicates the function as subsystem identification capable.

22.9.27 Subsystem Vendor IDs (SVID)—Offset 8Ch

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00008086h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0
SID																SVID															

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RW-L	**Subsystem Identifier (SID):** Indicates the subsystem as identified by the vendor. This field is write once and is locked down until a bridge reset occurs (not the PCI bus reset).
15:0	8086h RW-L	**Subsystem Vendor Identifier (SVID):** Indicates the manufacturer of the subsystem. This field is write once and is locked down until a bridge reset occurs (not the PCI bus reset).

22.9.28 Advanced Error Extended Reporting Capability Header (AECH)—Offset 100h

Size:32 bits
The AER capability can optionally be included or excluded from the capabilities list. The full AER is supported.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 20010001h

31			28				24				20				16				12				8				4				0
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
NCO															CV	CID															

Bit Range	Default and Access	Field Name (ID): Description
31:20	200h RW-L	**Next Capability Offset (NCO):** Contains the offset of the next structure in the extended capabilities list.
19:16	1h RO	**Capability Version (CV):** Indicates the version of the capability structure present.
15:0	0001h RO	**Capability ID (CID):** Identifies the function as Advanced Error Reporting capable.

22.9.29 Uncorrectable Error Status (UES)—Offset 104h

This register must maintain its state through a platform reset. It loses its state upon loss of core power.

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD					PTLLPEBS	TPBES	AEBE	MCE	UIE	AVS	URE	EE	MT	RO	UC	CA	CT	FCPE	PT	RSVD					SDE	DLPE	RSVD			TE		

Bit Range	Default and Access	Field Name (ID): Description
31:27	000h RV	**Reserved**.
26	0b RO	**Poisoned TLP Egress Blocked Status (PLTLPEBS)**
25	0b RO	**TLP Prefix Blocked Error Status (TPBES)**
24	0b RW1CS	**Atomic Egress Blocked Status (AEBE):** This bit will never be set.
23	0b RW1CS	**MC Blocked TLP Status (MCE):** This bit is set whenever a Multicast TLP is blocked
22	0b RW1CS	**Uncorrectable Internal Error Status (UIE):** This bit is set whenever an uncorrectable internal error is detected.
21	0b RO	**ACS Violation Status (AVS):** Reserved, not supported
20	0b RW1CS	**Unsupported Request Error Status (URE):** Indicates an unsupported request was received.
19	0b RW1CS	**ECRC Error Status (EE):** Set to 1b when an ECRC is detected.
18	0b RW1CS	**Malformed TLP Status (MT):** Indicates a malformed TLP was received.
17	0b RW1CS	**Receiver Overflow Status (RO):** Indicates a receiver overflow occurred.
16	0b RW1CS	**Unexpected Completion Status (UC):** Indicates an unexpected completion was received.
15	0b RW1CS	**Completor Abort Status (CA):** Indicates a completer abort was received
14	0b RO	**Completion Timeout Status (CT):** not applicable as the upstream port will not issue non-posted requests on it's one behalf.
13	0b RW1CS	**Flow Control Protocol Error Status (FCPE):** This bit is set when a flow control error protocol is detected.

Bit Range	Default and Access	Field Name (ID): Description
12	0b RW1CS	**Poisoned TLP Status (PT):** Indicates a poisoned TLP was received.
11:6	0h RV	**Reserved**. Software must write '0' to these bits
5	0b RO	**Surprise Down Error Status (SDE):** Surprise Down is not supported.
4	0b RW1CS	**Data Link Protocol Error Status (DLPE):** Indicates a data link protocol error occurred.
3:-	0h RV	**Reserved.**

22.9.30 Uncorrectable Error Mask (UEM)—Offset 108h

When set, the corresponding error in the UES register is masked, and the logged error will cause no action. When cleared, the corresponding error is enabled. This register is only reset by a loss of core power.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00400000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD					PTLTPEBM	TPBES	AEBEM	MCEM	UI	AVM	URE	EE	MT	RO	UC	CM	CT	FCPE	PT	RSVD					SDE	DLPE	RSVD			TE		

Bit Range	Default and Access	Field Name (ID): Description
31:27	0h RV	**Reserved.**
26	0b RO	**Poisoned TLP Egress Blocked Mask (PLTLPEBM)**: Not supported
25	0b RO	**TLP Prefix Blocked Error Mask (TPBES)**: not supported
24	0b RWS	**AtomicOP Egress Blocked Mask (AEBEM)**
23	0b RWS	**MC Blocked TLP Mask (MCEM):**
22	1b RWS	**Uncorrectable Internal Error Mask (UIEM)**
21	0b RO	**ACS Violation Mask (AVM):** Reserved. Access Control Services are not supported

Bit Range	Default and Access	Field Name (ID): Description
20	0b RWS	**Unsupported Request Error Mask (URE):**
19	0b RWS	**ECRC Error Mask (EE):**
18	0b RWS	**Malformed TLP Mask (MT):**
17	0b RWS	**Receiver Overflow Mask (RO):**
16	0b RWS	**Unexpected Completion Mask (UC):**
15	0b RWS	**Completor Abort Mask (CM):**
14	0b RWS	**Completion Timeout Mask (CT):** Not supported.
13	0b RWS	**Flow Control Protocol Error Mask (FCPE):**
12	0b RWS	**Poisoned TLP Mask (PT):**
11:6	0h RV	**Reserved.**
5	0h RO	**Surprise Down Error Mask (SDE):** not supported.
4	0b RWS	**Data Link Protocol Error Mask (DLPE):**
3:1	0h RO	**Reserved.**
0	0h RO	**Training Error Mask (TE):** Not supported.

22.9.31 Uncorrectable Error Severity (UEV)—Offset 10Ch

This register gives the option to make an uncorrectable error fatal or non-fatal. An error is fatal if the bit is set. An error is non-fatal if the bit is cleared. This register is only reset by a loss of core power.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00262010h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	1
RSVD						PTLBEBS	TPBES	AEBES	MCES	UIES	AVS	URE	EE	MT	RO	UC	CA	CT	FCPE	PT	RSVD							SDE	DLPE	RSVD				TE

Bit Range	Default and Access	Field Name (ID): Description
31:27	000h RV	**Reserved.**
26	0b RO	**Poisoned TLP Egress Blocked Severity (PTLBEBS)**
25	0b RO	**TLP Prefix Blocked Error Severity (TPBES)**
24	0b RWS	**AtomicOP Egress Blocked Severity (AEBES)**
23	0b RWS	**MC Blocked TLP Severity (MCES)**
22	1b RWS	**Uncorrectable Internal Error Severity (UIES)**
21	0b RO	**ACS Violation Severity (AVS):** Not supported
20	0b RWS	**Unsupported Request Error Severity (URE):**
19	0b RWS	**ECRC Error Severity (EE):**
18	1h RWS	**Malformed TLP Severity (MT):**
17	1b RWS	**Receiver Overflow Severity (RO):**
16	0b RWS	**Unexpected Completion Severity (UC):**
15	0b RWS	**Completor Abort Severity (CA):**
14	0b RO	**Completion Timeout Severity (CT):** Not Supported
13	1b RWS	**Flow Control Protocol Error Severity (FCPE):**
12	0b RWS	**Poisoned TLP Severity (PT):**
11:6	0h RV	**Reserved.**
5	0b RO	**Surprise Down Error Severity (SDE):** Not supported.
4	1b RWS	**Data Link Protocol Error Severity (DLPE):**
3:0	0h RV	**Reserved.**

22.9.32 Correctable Error Status (CES)—Offset 110h

This register is only reset by a loss of core power

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																HLOE	CIE	ANFES	RTT	RSVD			RNR	BD	BT	RSVD				RE		

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RV	**Reserved**. Software must write a "0" to these bits.
15	0b RW1CS	**Header Log Overflow Error (HLOE)**
14	0b RW1CS	**Correctable Internal Error (CIE)**
13	0b RW1CS	**Advisory Non-Fatal Error Status (ANFES):**
12	0b RW1CS	**Replay Timer Timeout Status (RTT):** Indicates the replay timer timed out.
11:9	0h RV	**Reserved.** Software must write '0' to these bits
8	0b RW1CS	**Replay Number Rollover Status (RNR):** Indicates the replay number roll over from 11 to 00.
7	0b RW1CS	**Bad DLLP Status (BD):** Indicates a bad DLLP was received.
6	0b RW1CS	**Bad TLP Status (BT):** Indicates a bad TLP was received.
5:1	0h RV	**Reserved.** Software must write '0' to these bits
0	0b RW1CS	**Receiver Error Status (RE):** Indicates that the physical layer detected a receiver error.

22.9.33 Correctable Error Mask (CEM)—Offset 114h

When set, the corresponding error in the CES register is masked, and the logged error will cause no action. When cleared, the corresponding error is enabled. This register is only reset by a loss of core power.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 000D0000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																HLOEM	CIEM	ANFEM	RTT	RSVD			RNR	BD	BT	RSVD				RE	

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RV	Reserved.
15	1b RWS	**Header Log Overflow Error Mask (HLOEM)**
14	1b RWS	**Correctable Internal Error Mask (CIEM)**
13	1b RWS	**Advisory Non-Fatal Error Mask (ANFEM):** When set, masks Advisory Non-Fatal errors from (a) signaling ERR_COR to the device control register and (b) updating the Uncorrectable Error Status register. This register is set by default to enable compatibility with software that does not comprehend Role-Based Error Reporting.
12	0b RWS	**Replay Timer Timeout Mask (RTTEM):**
11:9	0h RO	Reserved.
8	0b RWS	**Replay Number Rollover Mask (RNREM):**
7	0b RWS	**Bad DLLP Mask (BDLLPEM):.**
6	0b RWS	**Bad TLP Mask (BTLPEM):**
5:1	0h RO	Reserved.
0	0h RW/P	**Receiver Error Mask (REM):**

22.9.34 Advanced Error Capabilities and Control (AECC)—Offset 118h

This register gives the status and control for ECRC checks, and also the pointer to the first uncorrectable error that happened. This register is only reset by a loss of core power.

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																			CTPHLC	TLPPLP	MHRE	MHRC	ECE	ECC	EGE	EGC	FEP					

Bit Range	Default and Access	Field Name (ID): Description
31:13	0h RV	Reserved.
12	0b RO	**Completion Timeout Prefix/Header Log Capable (CTPHLC):** If Set, this bit indicates that the Function records the prefix/header of Request TLPs that experience a Completion Timeout error.
11	0b ROSV	**TLP Prefix Log Present (TLPPLP):** If Set and the First Error Pointer is valid, indicates that the TLP Prefix Log register contains valid information. If Clear or if First Error Pointer is invalid, the TLP Prefix Log register is undefined.
10	0b RO	**Multiple Header Recording Enable (MHRE):** Multiple Header recording is not supported
9	0b RO	**Multiple Header Recording Capable (MHRC):** Multiple header recording is not supported.
8	0b RWS	**ECRC Check Enable (ECE):**
7	0b RWL	**ECRC Check Capable (ECC):** ECRC is not supported.
6	0b RWS	**ECRC Generation Enable (EGE):**
5	0b RWL	**ECRC Generation Capable (EGC):**.
4:0	0h ROSV	**First Error Pointer (FEP):** This field identifies the bit position of the first error reported in the UncorrectableError Status Register (xref). This register re-arms itself (which does not change its current value) as soon as the error status bit indicated by the pointer is cleared by the software by writing a 1 to that status bit.

22.9.35 Header Log Register (AERHDRLOG[1-4])—Offset 11C - 128h by 4h

This register logs the 4 Dwords of the transaction header for PCI Express Errors.

Type: CFG Register
(Size: 32 bits)

Device:
Function: 0

Default: 00000000h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
AERHDRLOG																															

Bit Range	Default and Access	Field Name (ID): Description
31:0	00000000h ROSV	**TLP Header Log (TLPHDRLOG)**: As soon as an error is logged in this register, it remains locked for further error-logging until the software clears the status bit that caused the header log, thus rearming the logging of errors

22.9.36 Uncorrectable Error Detect Mask Register (ERRUNCDETMSK)—Offset 140h

A value of 1 masks the field and prevents detection as well as logging of errors.

Default: 00000000h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
RSVD					PTLPEDM	TPBEDM	AEBEDM	MCEDM	UIEDM	ACSEDM	UREDM	ACRCEDM	MTLPEDM	ROEDM	UCEDM	CAEDM	CTEDM	FCEDM	PTLPEDM	RSVD							SDEDM	DLPEDM	RSVD				

Bit Range	Default and Access	Field Name (ID): Description
31:27	0h RV	**Reserved**.
26	1b RWS	**Poisoned TLP Egress Blocked Detect Mask (PLTPEBDM)**
25	0b RO	**TLP Prefix Blocked Error Detect Mask (TPBEDM):** Not Supported
24	0b RWS	**AtomicOP Egress Blocked Error Detect Mask (AEBEDM)**
23	0b RWS	**MC Blocked TLP Error Detect Mask (MCEDM)**

Bit Range	Default and Access	Field Name (ID): Description
22	0b RWS	**Uncorrectable Internal Error Detect Mask (UIEDM)**
21	0b RO	**ACS Violation Error Detect Mask (ACSEDM)**: Not Supported
20	0b RWS	**Unsupported Request Error Detect Mask (UREDM)**
19	0b RWS	**ECRC Check Error Mask (ECRCEDM)**
18	0b RWS	**Malformed TLP Error Detect Mask (MTLPEDM)**
17	0b RWS	**Receiver Overflow Error Detect Mask (ROEDM)**
16	0b RWS	**Unexpected Completion Error Detect Mask (UCEDM)**
15	0b RWS	**Completer Abort Error Detect Mask (CAEDM)**
14	0b RWS	**Completion Timeout Error Detect Mask (CTEDM)**
13	0b RWS	**Flow Control Error Detect Mask (FCEDM)**
12	0b RWS	**Poisoned TLP Error Detect Mask (PTLPEDM)**
11:6	00h RV	**Reserved**
5	0b RO	**Surprise Down Error Detect Mask (SLDEDM):** Not Supported
4	0b RWS	**Data Link Protocol Error Detect Mask (DLPEDM)**
3:0	0h RV	**Reserved**

22.9.37 Correctable Error Detect Mask Register (ERRCORDETMSK)—Offset 144h

Type: CFG Register
(Size: 32 bits)

Device:
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																								T	E	U	C	R	B	V	

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RV	**Reserved.**
15	0b RWS	**Header Log Overflow Error Detect Mask (HLOEDM)**
14	0b RWS	**Correctable Internal Error DectMask (CIEDM)**
13	0b RWS	**Advisory Non-Fatal Error Detect Mask (ANFEDM):**
12	0b RWS	**Replay Timer Timeout Detect Mask (RTTEDM):**
11:9	0h RO	**Reserved.**
8	0b RWS	**Replay Number Rollover Detect Mask (RNREDM):**
7	0b RWS	**Bad DLLP Detect Mask (BDLLPEDM):**
6	0b RWS	**Bad TLP Detect Mask (BTLPEDM):**
5:1	0h RO	**Reserved.**
0	0h RWS	**Receiver Error Detect Mask (REDM):**

22.9.38 Secondary PCI Express Extended Capability Header (SECEXPCAP)—Offset 200h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00010019h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1
NCO												CV				PCIEECID															

Bit Range	Default and Access	Field Name (ID): Description
31:20	0h RW/O	**Next Capability Offset (NCO):** Next Capability Offset (NCO): This field contains the offset to the next PCI Express Capability structure or 000h if no other items exist in the linked list of Capabilities.
19:16	1b RO	**Capability Version (CV):** Capability Version (CV): This field is a PCI-SIG defined version number that indicates the version of the Capability structure present.
15:0	19h RO	**PCI Express Extended Capability ID (PCIEECID):** PCI Express Extended Capability ID for the Secondary PCI Express Extended Capability is 0019h.

22.9.39 Link Control 3 (LCTL3)—Offset 204h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																ELSOSGV							RSVD							LERIE	PE

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:9	0h RW	**Enable Lower SKP OS Generation Vector (ELSOSGV):** Enable Lower SKP OS Generation Vector(ELSOSGV): When the Link is in L0 and the bit in this field corresponding to the current Link speed is Set, SKP Ordered Sets are scheduled at the rate defined for SRNS, overriding the rate required based on the clock tolerance architecture. Bit definitions within this field are: Bit 0 2.5 GT/s Bit 1 5.0 GT/s Bit 2 8.0 GT/s Bits 6:3 RsvdP Behavior is undefined if a bit is Set in this field and the corresponding bit in the Lower SKP OS Generation Supported Speeds Vector is not set.
8:2	0h RO	Reserved.
1	0h R0	**Link Equalization Request Interrupt Enable (LNKEQREQINTEN):** Not applicable for upstream ports
0	0h R0	**Perform Equalization (PE):** Not applicable for upstream ports

22.9.40 Lane Error Status Register (LANEERRSTS)—Offset 208h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1
RSVD																								L7ERR	L6ERR	L5ERR	L4EERR	L3ERR	L2ERR	L1ERR	L0ERR

Bit Range	Default and Access	Field Name (ID): Description
31:8	0000h RV	Reserved.
7	0b RW1CS	**Lane 7Error (L7ERR)**: A value of 1b indicates that a lane based error was detected on this PCIe Express lane.
6	0b RW1CS	**Lane 6Error (L6ERR)**: A value of 1b indicates that a lane based error was detected on this PCIe Express lane.
5	0b RW1CS	**Lane 5Error (L5ERR)**: A value of 1b indicates that a lane based error was detected on this PCIe Express lane.
4	0b RW1CS	**Lane 4 Error (L4ERR)**: A value of 1b indicates that a lane based error was detected on this PCIe Express lane.
3	0b RW1CS	**Lane 3Error (L3ERR)**: A value of 1b indicates that a lane based error was detected on this PCIe Express lane.
2	0b RW1CS	**Lane 2Error (L2ERR)**: A value of 1b indicates that a lane based error was detected on this PCIe Express lane.
1	0b RW1CS	**Lane 1Error (L1ERR)**: A value of 1b indicates that a lane based error was detected on this PCIe Express lane.
0	0b RW1CS	**Lane 0 Error (L0ERR)**: A value of 1b indicates that a lane based error was detected on this PCIe Express lane.

22.9.41 Lane Equalization Control Register (LANEEQCTL[0-7]))—Offset 20C - 21Ah by 2h (for up to 8 upstream lanes)

Each entry contains the values for the Lane with the corresponding default Lane number which is invariant to Link width and Lane reversal negotiation that occurs during Link training.

Type: CFG Register
(Size: 16bits)

Device: 0
Function: 0

Default: 7F00

15	14	13	12	11	10	9	8	7	6	5	3	2	2	1	0
0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
RSVD	UPRPH			UPTP				RSVD							

Bit Range	Default and Access	Field Name (ID): Description
15	0h RV	Reserved.
14:12	7h ROV	**Upstream Port Lane 1 Receiver Preset Hint (UPRPH):** Upstream Port Lane 1 Receiver Preset Hint (UPL1RPH): Field contains the Receiver Preset Hint value sent or received during Link Equalization. The settings are: 000b: -6 dB 001b: -7 dB 010b: -8 dB 011b: -9 dB 100b: -10 dB 101b: -11 dB 110b: -12 dB 111b: Reserved
11:8	Fh ROV	**Upstream Port Lane 1 Transmitter Preset (UPTP):** Upstream Port Lane 1 Transmitter Preset (UPL1TP): Field contains the Transmit Preset value sent or received during Link Equalization. The settings are: 0000b: -6 dB for de-emphasis, 0 dB for preshoot 0001b: -3.5 dB for de-emphasis, 0 dB for preshoot 0010b: -4.5 dB for de-emphasis, 0 dB for preshoot 0011b: -2.5 dB for de-emphasis, 0 dB for preshoot 0100b: 0 dB for de-emphasis, 0 dB for preshoot 0101b: 0 dB for de-emphasis, 2 dB for preshoot 0110b: 0 dB for de-emphasis, 2.5 dB for preshoot 0111b: -6 dB for de-emphasis, 3.5 dB for preshoot 1000b: -3.5 dB for de-emphasis, 3.5 dB for preshoot 1001b: 0 dB for de-emphasis, 3.5 dB for preshoot 1010b: The Transmitter Preset encoding of 1010b corresponds to the maximum Vtx-boost that the Transmitter advertises with FS and LF in both full swing and reduced swing modes with C_{-1} = 0 and C_{+1} = FS – (2*LF). All other encodings are reserved.
7:0	00h RV	Reserved.

22.9.42 Implementation Specific Registers

The following register lies in PCIe configuration space. It is not part of any PCIe specification register definition and is unique to the implementation on the Intel® C620 Series Chipset. With the normal register discovery mechanism this can not be found, and is only discovered by being listed here.

22.9.43 Personality Lock Key Control Register. (PLKCTL) —Offset EAh

Type: CFG Register
(Size: 16 bits)

Device: 0
Function: 0

Default: 00000000h

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD															cl

Bit Range	Default and Access	Field Name (ID): Description
31:1	0h Rv	Reserved.
0	0b RW-L	**Capability Lock (CL):** Lock key bit for all RWS-L bits (capabilities, next capability pointer, SSID/SVID, etc.) 1b = lock 0b = unlocked ***Note:*** This bit is self-locking. Once this bit is set to 1b, this key bit can not be unlocked. Writing a 0b has no effect on this bit.

22.10 PCI Express Virtual Switch Port (VSP)[5:0] Configuration Registers Summary

There are six virtual switch ports integrated into the PCH. The PCIe configuration space is identical for all VSP[5:0] except for the respective VSP Device ID.

Table 22-11. Summary of PCI Express Port Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	Identifiers (ID)—Offset 0h	VSP0: 37C28086h VSP1: 37C38086h VSP2: 37C48086h VSP3: 37C58086h VSP4: 37C68086h VSP5: 37C78086h
4h	7h	Device Command; Primary Status (CMD_PSTS)—Offset 4h	0010 0000h
8h	Bh	Revision ID;Class Code (RID_CC)—Offset 8h	0604 0000h
Ch	Fh	Cache Line Size; Primary Latency Timer; Header Type (CLS_PLT_HTYPE)—Offset Ch	0001 0000h
10h	17h	Express Port Memory Base Address Register (EXPPTMBAR_VSP) — Offset 10h	0000 0000 0000 0004h
18h	1Bh	Bus Numbers (BusNUM)—Offset 18h	0000 0000h
1Ch	1Fh	I/O Base and Limit; Secondary Status (IOBL_SSTS)—Offset 1Ch	0000 0000h
20h	23h	Memory Base and Limit (MBL)—Offset 20h	0000 0000h
24h	27h	Prefetchable Memory Base and Limit (PMBL)—Offset 24h	0001 0001h
28h	2Bh	Prefetchable Memory Base Upper 32 Bits (PMBU32)—Offset 28h	0000 0000h
2Ch	2Fh	Prefetchable Memory Limit Upper 32 Bits (PMLU32)—Offset 2Ch	0000 0000h
34h	37h	Capabilities List Pointer (CAPP)—Offset 34h	0000 0040h
3Ch	3Fh	Interrupt Information; Bridge Control (INTR_BCTRL)—Offset 3Ch	0000 0100h
40h	43h	Capabilities List; PCI Express Capabilities (CLIST_XCAP)—Offset 40h	0062 8010h
44h	47h	Device Capabilities (DCAP)—Offset 44h	0000 8002h
48h	4Bh	Device Control; Device Status (DCTL_DSTS)—Offset 48h	0000 2000h
4Ch	4Fh	Link Capabilities (LCAP)—Offset 4Ch	0800 0C11h
50h	53h	Link Control; Link Status (LCTL_LSTS)—Offset 50h	1011 0000h
64h	67h	Device Capabilities 2 (DCAP2)—Offset 64h	0000 0000h
68h	6Bh	Device Control 2; Device Status 2 (DCTL2_DSTS2)—Offset 68h	0000 0000h
6Ch	6Fh	Link Capabilities 2 (LCAP2)—Offset 6Ch	0000 0002h

Table 22-11. Summary of PCI Express Port Configuration Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
70h	73h	Link Control 2; Link Status 2 (LCTL2_LSTS2)—Offset 70h	0000 0001h
80h	83h	Power Management Capability; PCI Power Management Capabilities (PMCAP_PMC)—Offset 80h	C803 8801h
84h	87h	PCI Power Management Control and Status (PMCS)—Offset 84h	0000 0008h
88h	8Bh	Subsystem Capability List Register (SVCAP)—Offset 88h	0000 000Dh
8C	8F	Subsystem Vendor IDs (SVID)—Offset 8Ch	0000 8086h
100h	103h	Advanced Error Extended Reporting Capability Header (AECH)—Offset 100h	1381 0001h
104h	107h	Uncorrectable Error Status (UES)—Offset 104h	0000 0000h
108h	10Bh	Uncorrectable Error Mask (UEM)—Offset 108h	0040 0000h
10Ch	10Fh	Uncorrectable Error Severity (UEV)—Offset 10Ch	0044 0010h
110h	113h	Correctable Error Status (CES)—Offset 110h	0000 0000h
114h	117h	Correctable Error Mask (CEM)—Offset 114h	0000 2000h
118h	11Bh	Advanced Error Capabilities and Control (AECC)—Offset 118h	0000 0000h
11Ch	128h	Header Log Register (AERHDRLOG[1-4])—Offset 11C - 128h by 4h	0000 0000h
138h	13Bh	Access Control Services Extended Capability Header (ACSCAPHDR)—Offset 138h	0001 000Dh
13Ch	13Fh	Access Control Services Capability and Control Register (ACSCAP_ACSCTL)—Offset 13Ch	0000 005Fh
140h	143h	Correctable Error Detect Mask Register (ERRCORDETMSK)—Offset 144h	0400 0000h
144h	147h	Correctable Error Detect Mask Register (ERRCORDETMSK)—Offset 144h	0000 0000h

22.10.1 Identifiers (ID)—Offset 0h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 37Cx8086h

31			28				24				20				16				12				8				4				0
0	0	1	1	0	1	1	1	1	1	0	0	0	x	x	x	1	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0
DID																VID															

Bit Range	Default and Access	Field Name (ID): Description
31:16	VSP[4:0] Refer to Description RO/V	**Device Identification (DID):** Identifies this particular function: VSP0: 37C2h associated with Intel QuickAssist Technology EP0 VSP1: 37C3h associated with Intel QuickAssist Technology EP1 VSP2: 37C4h associated with Intel QuickAssist Technology EP2 VSP3: 37C5h associated with 10 GbE VSP4: 37C6h associated Thermal Sensor in Endpoint Only (EPO) mode VSP5: 37C7h
15:0	8086h RO	**Vendor Identification (VID):** Indicates Intel.

22.10.2 Device Command; Primary Status (CMD_PSTS)—Offset 4h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00100000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DPE	SSE	RMA	RTA	STA	PDTS		DPD	PFBC	RSVD	PC66	CLIST	IS	RSVD								ID	FBE	SEE	WCC	PERE	VGA_PSE	MWIE	SCE	BME	MSE	IOSE

Bit Range	Default and Access	Field Name (ID): Description
31	0b RW1C	**DPE - Detected Parity Error (DPE):** Set when the port receives a poisoned tlp from the PCIe interface of the internal backboner. This is set even if CMD.PERE is not set.
30	0b RW1C	**Signaled System Error (SSE):** This bit is set when ERR_FATAL or ERR_NONFATAL messages are sent to the root complex and the SERR enable bit in the PCICMD Register is set
29	0b RW1C	**Received Master Abort (RMA):** Set when the requester receives a completion with unsupported request status.
28	0b RW1C	**Received Target Abort (RTA):** Set when the root port receives a completion with completer abort.
27	0b RW1C	**Signaled Target Abort (STA):** This bit is Set when the port completes a Posted or Non-Posted Request as a Completer Abort error. This applies to a Function with a Type 1 Configuration header when the Completer Abort was generated by its Primary Side.
26:25	0b RO	**Primary DEVSEL# Timing Status (PDTS):** Reserved per *PCI Express Specification*.
24	0b RW1C/V	**Master Data Parity Error Detected (DPD):** This bit is set by a requester (primary side for type1 configuration headerfunctions) if the parity error response enable bit (PERE) in the Command Register is set and either of the following two conditions occur: • Requester receives a completion marked poisoned. • Requester sends a poisoned request (includes writes and messages) If the parity error bit is 0b, this bit is never set.

Bit Range	Default and Access	Field Name (ID): Description
23	0b RO	**Primary Fast Back to Back Capable (PFBC):** Reserved per *PCI Express Specification*.
22	0b RV	**Reserved**.
21	0b RO	**Primary 66 MHz Capable (PC66):** Reserved per *PCI Express Specification*.
20	1b RO	**Capabilities List (CLIST):** Indicates the presence of a capabilities list.
19	0b RO/V	**Interrupt Status (IS):** When set indicates that an INTx emulation interrupt is pending internally for this function.
18:11	0h RV	**Reserved**.
10	0b RW	**Interrupt Disable (ID):** This disables pin-based INTx# interrupts on enabled hot plug and power management events. This bit has no effect on MSI operation. When set, internal INTx# messages will not be generated. When cleared, internal INTx# messages are generated if there is an interrupt for hot plug or power management and MSI is not enabled. This bit does not affect interrupt forwarding from devices connected to the root port. Assert_INTx and Deassert_INTx messages will still be forwarded to the internal interrupt controllers if this bit is set. For PCI Bus Emulation Mode compatibility, if the PCIBEM register is set, this register is RO and returns a value of 0 when read, else it is RW with the functionality described above.
9	0b RO	**Fast Back to Back Enable (FBE):** Reserved per *PCI Express Specification*.
8	0b RW	**SERR# Enable (SEE):** When set, enables the root port to generate an SERR# message when PSTS.SSE is set.
7	0b RO	**Wait Cycle Control (WCC):** Reserved per *PCI Express Specification*.
6	0b RW	**Parity Error Response Enable (PERE):** Indicates that the device is capable of reporting parity errors as a master on the backbone.
5	0b RO	**VGA Palette Snoop (VGA_PSE):** Reserved per *PCI Express Specification*.
4	0b RO	**Memory Write and Invalidate Enable (MWIE):** Reserved per *PCI Express Specification*.
3	0b RO	**Special Cycle Enable (SCE):** Reserved per *PCI Express Specification* and *PCI Bridge Specification*.
2	0b RW	**Bus Master Enable (BME):.**This bit controls the ability of the Function to issue Memory and I/O read or write requests, and the ability of Root or Switch port to forward memory and I/O read or write requests in the upstream direction. When this bit is 0b, memory and I/O requests received at the root port or downstream side of a switch port (secondary side) must be handled as an Unsupported Request (UR). For Non-posted requests, a completion with UR completion status must be returned. For an endpoint, when this bit is Set, the PCI Express Function is allowed to issue Memory or I/O Requests. When this bit is Clear, the PCI Express Function is not allowed to issue any Memory or I/O Requests.
1	0b RW	**Memory Space Enable (MSE):** When set, memory cycles within the range specified by the memory base and limit registers can be forwarded to the PCI-Express device. When cleared, these memory cycles are responded to on the primary interface with an unsupported request completion.
0	0b RW	**I/O Space Enable (IOSE):** When set, I/O cycles within the range specified by the I/O base and limit registers can be forwarded to the PCI-Express device. When cleared, these cycles are responded to on the primary interface with an unsupported request completion.

22.10.3 Revision ID;Class Code (RID_CC)—Offset 8h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 06040000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
BCC								SCC								PI								RID								

Bit Range	Default and Access	Field Name (ID): Description
31:24	06h RW/L	**Base Class Code (BCC):** Indicates the device is a bridge device.
23:16	04h RW/L	**Sub-Class Code (SCC):** The default indicates the device is a PCI-to-PCI bridge. If the MPC.BT register is set to a '1' for a Host Bridge, this register reads 00h.
15:8	0h RW/L	**Programming Interface (PI):** This bit indicates that this device is standard (non-subtractive) PCI-to-PCI Bridge.
7:0	00h RO/V	**Revision ID (RID):** Indicates the revision of the function.

22.10.4 Cache Line Size; Primary Latency Timer; Header Type (CLS_PLT_HTYPE)—Offset Ch

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00010000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BIST_TST								MFD	HTYPE							CT								LS								

Bit Range	Default and Access	Field Name (ID): Description
31:24	00h RO	**BIST tests (BIST_TST).** Not supported, hard wired to 00h
23	0b RW/L	**Multi-function Device (MFD):** This bit is '0' to indicate a single function device.

Bit Range	Default and Access	Field Name (ID): Description
22:16	01h RW/L	**Header Type (HTYPE):** The default mode identifies the header layout of the configuration space, which is a PCI-to-PCI bridge. If the MPC.BT register is set to a '1' for a Host Bridge, this register reads 00h.
15:8	00h RO	**Latency Count (CT):** Reserved per *PCI Express Specification*.
7:0	00h RW	**Line Size (LS):** This is read/write but contains no functionality, per *PCI Express Specification*.

22.10.5 Express Port Memory Base Address Register (EXPPTMBAR_VSP) —Offset 10h

Access Method

Type: CFG Register
(Size: 64 bits)

Device: 0
Function: 0

Default: 0000000000000004h

63			60				56				52				48				44				40				36				32
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BADDR																															

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
BADDR															RSVD														PREF	TYPE		MIS

Bit Range	Default and Access	Field Name (ID): Description
63:17	00000000 0000h RW	**Base Address (BADDR):**
16:4	000h Rv	**Reserved**
3	0b RO	**Prefetchable Memory (Pref)**: Not prefetchable memory space.
2:1	10b RO	**Type (type):** The memory mapped space can be placed anywhere in the 64-bit addressable region of the device
0	0b RO	**Memory Space Indicator (MSI):** 0b = Memory space, 1b - I/O space. Indicates the bus number of the backbone.

22.10.6 Bus Numbers (BusNUM)—Offset 18h

Type: CFG Register
(Size: 24bits)

Device: 0
Function: 0

Default: 0h

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SBBN								SCBN								PBN							

Bit Range	Default and Access	Field Name (ID): Description
23:16	0h RW	**Subordinate Bus Number (SBBN):** Indicates the highest PCI bus number below the bridge.
15:8	0h RW	**Secondary Bus Number (SCBN):** Indicates the bus number secondary interface.
7:0	0h RW	**Primary Bus Number (PBN):** Indicates the PCI Express bus number

22.10.7 I/O Base and Limit; Secondary Status (IOBL_SSTS)—Offset 1Ch

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000000h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DPE	RSE	RMA	RTA	STA	SDTS		MDPD	FBC	RSVD	SC66	RSVD				IOLA				IOLC				IOBA				IOBC				

Bit Range	Default and Access	Field Name (ID): Description
31	0b RW1C/V	**Detected Parity Error (DPE):** Set when the port receives a poisoned TLP.
30	0b RW1C/V	**Received System Error (RSE):** Set when the port receives an ERR_FATAL or ERR_NONFATAL message from the device.
29	0b RW1C/V	**Received Master Abort (RMA):** Set when the port receives a completion with 'Unsupported Request' status from the device.
28	0b RW1C/V	**Received Target Abort (RTA):** Set when the port receives a completion with 'Completion Abort' status from the device.
27	0b RW1C/V	**Signaled Target Abort (STA):** Set when the port generates a completion with 'Completion Abort' status to the device.

Bit Range	Default and Access	Field Name (ID): Description
26:25	0h RO	**DEVSEL# Timing Status (DVT):** Reserved per *PCI Express Specification*.
24	0b RW1C/V	**Master Data Parity Error Detected (MDPD):** Set when the BCTRL.PERE, and either of the following two conditions occurs: Port receives completion marked poisoned. Port poisons a write request to the secondary side.
23	0h RO	**Fast Back to Back Capable (FBC):** Reserved per *PCI Express Specification*.
22	0b RV	**Reserved**.
21	0b RO	**Secondary 66 MHz Capable (SC66):** Reserved per *PCI Express Specification*.
20:16	0h RV	**Reserved.**
15:12	0h RW	**I/O Address Limit (IOLA):** I/O Base bits corresponding to address lines 15:12 for 4KB alignment. Bits 11:0 are assumed to be padded to FFFh.
11:8	0h RO	**I/O Limit Address Capability (IOLC):** Indicates that the bridge does not support 32-bit I/O addressing.
7:4	0h RW	**I/O Base Address (IOBA):** I/O Base bits corresponding to address lines 15:12 for 4KB alignment. Bits 11:0 are assumed to be padded to 000h.
3:0	0h RO	**I/O Base Address Capability (IOBC):** Indicates that the bridge does not support 32-bit I/O addressing.

22.10.8 Memory Base and Limit (MBL)—Offset 20h

Accesses that are within the ranges specified in this register will be sent to the attached device if CMD.MSE is set. Accesses from the attached device that are outside the ranges specified will be forwarded to the backbone if CMD.BME is set. The comparison performed is MB [gt]= AD[lb]31:20[rb] [lt]= ML.

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ML												RSVD				MB												RSVD				

Bit Range	Default and Access	Field Name (ID): Description
31:20	0h RW	**Memory Limit (ML):** These bits are compared with bits [31:20] of the incoming address to determine the upper 1 MB aligned value of the range.

Bit Range	Default and Access	Field Name (ID): Description
19:16	0h RV	**Reserved**.
15:4	0h RW	**Memory Base (MB):** These bits are compared with bits [31:20] of the incoming address to determine the lower 1 MB aligned value of the range.
3:0	0h RV	**Reserved**.

22.10.9 Prefetchable Memory Base and Limit (PMBL)—Offset 24h

Accesses that are within the ranges specified in this register will be sent to the device if CMD.MSE is set. Accesses from the device that are outside the ranges specified will be forwarded to the backbone if CMD.BME is set. The comparison performed is PMBU32:PMB [gt]= AD[lb]63:32[rb]:AD[lb]31:20[rb] [lt]= PMLU32:PML.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00010001h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
PML												I64L				PMB												I64B			

Bit Range	Default and Access	Field Name (ID): Description
31:20	000h RW	**Prefetchable Memory Limit (PML):** These bits are compared with bits [31:20] of the incoming address to determine the upper 1 MB aligned value of the range.
19:16	1h RO	**64-bit Indicator (I64L):** Indicates support for 64-bit addressing.
15:4	000h RW	**Prefetchable Memory Base (PMB):** These bits are compared with bits [31:20] of the incoming address to determine the lower 1 MB aligned value of the range.
3:0	1h RO	**64-bit Indicator (I64B):** Indicates support for 64-bit addressing.

22.10.10 Prefetchable Memory Base Upper 32 Bits (PMBU32)—Offset 28h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PMBU																															

Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Prefetchable Memory Base Upper Portion (PMBU):** Upper 32-bits of the prefetchable address base.

22.10.11 Prefetchable Memory Limit Upper 32 Bits (PMLU32)—Offset 2Ch

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PMLU																															

Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Prefetchable Memory Limit Upper Portion (PMLU):** Upper 32-bits of the prefetchable address limit.

22.10.12 Capabilities List Pointer (CAPP)—Offset 34h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000040h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
RSVD																								PTR							

Bit Range	Default and Access	Field Name (ID): Description
31:8	0h RO	Reserved.
7:0	40h RW/O	**Capabilities Pointer (PTR):** Indicates that the pointer for the first entry in the capabilities list. BIOS can determine which capabilities will be exposed by including or removing them from the capability linked list.

22.10.13 Interrupt Information; Bridge Control (INTR_BCTRL)—Offset 3Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000100h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
RSVD				DTSE	DTS	SDT	PDT	FBE	SBR	MAM	V16	VE	IE	SE	PERE	IPIN								ILINE							

Bit Range	Default and Access	Field Name (ID): Description
31:28	0h RV	**Reserved**.
27	0b RO	**Discard Timer SERR# Enable (DTSE):** Reserved per *PCI Express Specification*.
26	0b RO	**Discard Timer Status (DTS):** Reserved per *PCI Express Specification*.
25	0b RO	**Secondary Discard Timer (SDT):** Reserved per *PCI Express Specification*.

Bit Range	Default and Access	Field Name (ID): Description
24	0b RO	**Primary Discard Timer (PDT):** Reserved per *PCI Express Specification*.
23	0b RO	**Fast Back to Back Enable (FBE):** Reserved per *PCI Express Specification*.
22	0b RW	**Secondary Bus Reset (SBR):** Triggers a Hot Reset on the PCI-Express port.
21	0b RO	**Master Abort Mode (MAM):** Reserved per *PCI Express Specification*.
20	0b RW	**VGA 16-Bit Decode (V16):** When set, indicates that the I/O aliases of the VGA range (see BCTRL:VE definition below), are not enabled. 0: Execute 10-bit address decode on VGA I/O accesses. 1: Execute 16-bit address decode on VGA I/O accesses.
19	0b RW	**VGA Enable (VE):** When set to 1b, the following ranges will be forwarded from the primary to the secondary side of the bridge regardless of the value of the I/O base and limit registers. Memory ranges A0000h-BFFFFh I/O ranges 3B0h 3BBh and 3C0h 3DFh, and all aliases of bits [15:10] in any combination of 1's
18	0b RW	**ISA Enable (IE):** This bit only applies to I/O addresses that are enabled by the I/O Base and I/O Limit registers and are in the first 64KB of PCI I/O space. If this bit is set, the PCIe bridge will block any forwarding from the primary to the secondary sided I/O transactions addressing the last 768 bytes in each 1KB block (offsets 100h to 3FFh). In the opposite direction (secondary to primary) I/O transactions will be forwarded if they address the last 768B in each 1KB block. 1: Forward upstream ISA I/O addresses in the address range defined by the I/O Base and I/O Limit registers that are in the firsts 64 KB of PCI I/O address space (Top 768B of each 1K block). 0: Forward downstream all I/O addresses in the address range defined by the I/O Base and I/O Limit registers.
17	0b RW	**SERR# Enable (SE):** This bit controls the forwarding of ERR_COR, ERR_NONFATAL AND ERR_FATAL messages to the primary side of the PCIe port. 1: Enables forwarding of ERR_COR, ERR_NONFATA, ERR_FATAL messages. 0: Disables forwarding of ERR_COR, ERR_NONFATA, ERR_FATAL messages.
16	0b RW	**Parity Error Response Enable (PERE):** This bit controls the response to poisoned TLPs in the PCI Express port. 1: Enables reporting of poisoned TLP errors. 0: Disables reporting of poisoned TLP errors.
15:8	01h RW/L	**Interrupt Pin (IPIN):** This register tells which interrupt pin the function uses. 01h: Generate INTA 02h: Generate INTB 03h: Generate INTC 04h: Generate INTD Others: Reserved BIOS has the ability to write this register during boot to setup the correct interrupt for the function. Once the bits are locked they can not be updated until another PLTRST#
7:0	0h RW	**Interrupt Line (ILINE):** Software written value to indicate which interrupt line (vector) the interrupt is connected to. No hardware action is taken on this register.

22.10.14 Capabilities List; PCI Express Capabilities (CLIST_XCAP)—Offset 40h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00628010h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
RSVD			IMN				SI	DT				CV				NEXT								CID								

Bit Range	Default and Access	Field Name (ID): Description
31:30	0h RV	**Reserved.**
29:25	0h RO	**Interrupt Message Number (IMN):** This field indicates the interrupt message number that is generated from the PCI Express port. When there is more than one MSI interrupt number, this register is required to contain the offset between the base Message Data and the MSI Message that is generated when the status bits in the slot status register or root port status registers are set. The chipset is required to update this field if the number of MSI messages change.
24	0b RO	**Slot Implemented (SI):** Hardwired to 0b.
23:20	6h RO	**Device / Port Type (DT):** Indicates this a Virtual Switch Port of a PCIe switch.
19:16	2h RO	**Capability Version (CV):** Version 2.0 indicates devices compliant to the *PCI Express 2.0 Specification* which incorporates the Register Expansion ECN.
15:8	80h RO	**Next Capability (NEXT):** Indicates the location of the next capability.
7:0	10h RO	**Capability ID (CID):** Indicates this is a PCI Express capability.

22.10.15 Device Capabilities (DCAP)—Offset 44h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00008002h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
RSVD			FLRC	CSPS			CSPV								RSVD	RBER	RSVD			E1AL			E0AL			ETFS	PFS			MPS		

Bit Range	Default and Access	Field Name (ID): Description
31:29	0h RV	**Reserved**.
28	0b RV	**Function Level Reset Capable (FLRC):** Reserved, not applicable.
27:26	0h RO	**Captured Slot Power Limit Scale (CSPS):** This value is set by the set_slot_power_limit message.
25:18	0h RO	**Captured Slot Power Limit Value (CSPV):** This value is set by the set_slot_power_limit message.
17:16	0h RV	**Reserved.**
15	1b RO	**Role Based Error Reporting (RBER):** The PCIe cluster supports Role-based Error reporting.
14:12	0h RV	**Reserved**.
11:9	0h RO	**Endpoint L1 Acceptable Latency (E1AL):** Reserved
8:6	0h RO	**Endpoint L0 Acceptable Latency (E0AL):** Reserved
5	0h RO	**Extended Tag Field Supported (ETFS):** 0: The PCIe supports 5-bit extended tag 1: The PCIe supports 8-bit extended tag.
4:3	0h RO	**Phantom Functions Supported (PFS):** No phantom functions supported
2:0	2h RW/L	**Max Payload Size Supported (MPS):** BIOS should write to this field during system initialization. The Max Payload Size supported is 512 bytes. 000b: 128 bytes max payload size. 001b: 256 bytes max payload size. 010b: 512 bytes max payload size. 011b: 1024 bytes max payload size. 100b: 2048 bytes max payload size. 101b: 4096 bytes max payload size. 110b: Reserved. 111b: Reserved.

22.10.16 Device Control; Device Status (DCTL_DSTS)—Offset 48h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00002000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD										TDP	APD	URD	FED	NFED	CED	RSVD	MRRS			ENS	APME	PFE	ETFE	MPS				ERO	URE	FEE	NFE	CEE

Bit Range	Default and Access	Field Name (ID): Description
31:22	0h RV	Reserved. Software must always write a "0" to these bits.
21	0b RO-V	**Transactions Pending (TDP):** Not applicable
20	0b RO	**AUX Power Detected (APD):** Auxiliary power is not supported
19	0b RW/1C/V	**Unsupported Request Detected (URD):** Indicates an unsupported request was detected from the PCI Express link.
18	0b RW/1C/V	**Fatal Error Detected (FED):** Indicates a fatal error was detected. Errors are logged in this register regardless of whether error reporting is enabled or not in the Device Control Register.
17	0b RW/1C/V	**Non-Fatal Error Detected (NFED):** Indicates a non-fatal error was detected. Errors are logged in this register regardless of whether error reporting is enabled or not in the Device Control Register.
16	0b RW/1C/V	**Correctable Error Detected (CED):** Indicates a correctable error was detected. Errors are logged in this register regardless of whether error reporting is enabled or not in the Device Control Register.
15	0h RV	Reserved.
14:12	2h RW	**Max Read Request Size (MRRS):** This field sets the maximum Read Requests size of the function as a requester. The Function must not generate read requests with size exceeding the set value. 000b: 128 bytes maximum Read Request size 001b: 256 bytes maximum Read Request size 010b: 512 bytes maximum Read Request size 011b: 1024 bytes maximum Read Request size 100b: 2048 bytes maximum Read Request size 101b: 4096 bytes maximum Read Request size Others: Reserved
11	0h RO	**Enable No Snoop (ENS):** Not supported.
10	0h RO	**Aux Power PM Enable (APME):** Not supported.
9	0h RO	**Phantom Functions Enable (PFE):** Not supported.
8	0h RO	**Extended Tag Field Enable (ETFE):** Not supported.

Bit Range	Default and Access	Field Name (ID): Description
7:5	0h RW	**Max Payload Size (MPS):** This field sets maximum TLP payload size for the function. As a receiver, the function must handle TLPs as large as the set value. As a Transmitter, the function must not generate TLPs exceeding the set value. 000b: 128 bytes maximum payload size (default) 001b: 256 bytes maximum payload size 010b: 512 bytes maximum payload size 011b: 1024 bytes maximum payload size 100b: 2048 bytes maximum payload size 101b: 4096 bytes maximum payload size Others: Reserved
4	0h RO	**Enable Relaxed Ordering (ERO):** Not supported.
3	0h RW	**Unsupported Request Reporting Enable (URE):** This bit controls the enabling of ERR_CORR, ERR_NONFATAL or ERR_FATAL messages on PCI Express for reporting "Unsupported Request" errors.
2	0h RW	**Fatal Error Reporting Enable (FEE):** When this bit is set, generation of the ERR_FATAL message is enabled
1	0h RW	**Non-Fatal Error Reporting Enable (NFE):** When this bit is set, generation of the ERR_NONFATAL message is enabled.
0	0h RW	**Correctable Error Reporting Enable (CEE):** When this bit is set, generation of the ERR_CORR message is enabled.

22.10.17 Link Capabilities (LCAP)—Offset 4Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 08000C11h

31			28				24				20				16				12				8				4				0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	1	0	0	0	1
PN								RSVD			LBNC	LARC	SDERC	CPM	EL1			EL0			APMS		MLW					MLS				

Bit Range	Default and Access	Field Name (ID): Description
31:24	08h RW-L	**Port Number (PN):** Indicates the port number assigned to this link for Virtual Switch Port.
23:22	0b RV	**Reserved**.
21	0b RO	**Link Bandwidth Notification Capability (LBNC):** Reserved, not supported
20	0b RO	**Link Active Reporting Capable (LARC):** Reserved, not supported

Bit Range	Default and Access	Field Name (ID): Description
19	0b RO	**Surprise Down Error Reporting Capable (SDERC):** Set to '0' to indicate the uplink does not support Surprise Down Error Reporting.
18	0b RO	**Clock Power Management (CPM):** '0' Indicates that the upstream port does not support the CLKREQ# mechanism.
17:15	000b RO	**L1 Exit Latency (EL1):** Indicates an exit latency of 2 us to 4 us. 000b Less than 1 us 001b 1 us to less than 2 us 010b 2 us to less than 4 us 011b 4 us to less than 8 us 100b 8 us to less than 16 us 101b 16 us to less than 32 us 110b 32 us to 64 us 111b More than 64 us
14:12	000b RO	**L0s Exit Latency (EL0):** This field indicates the L0s exit latency for the given PCI-Express Link. It indicates the length of time this port requires to complete transition from L0s to L0. 000b: Less than 64 ns 001b: 64 ns to less than 128 ns 010b: 128 ns to less than 256 ns 011b: 256 ns to less than 512 ns 100b 512 ns to less than 1 µs 101b: 1 us to less than 2 us 110b: 2 us to less than 4 us 111b: More than 4 us
11:10	11b RO	**Active State Link PM Support (APMS):** Indicates the level of active state power management on this link Bits Definition 00 No ASPM Supported 01 L0s Supported 10 L1 Supported 11 L0s and L1 supported ***Note:*** L0s should not be advertised when operating with SRIS.
9:4	1h RO	**Maximum Link Width (MLW):** This field indicates the maximum link width implemented by the given PCI Express Link. 00h: Reserved 01h: x1 02h: x2 04h: x4 08h: x8 10h: x16 20h: x32 (Unsupported) Others Reserved ***Note:*** Default value is determined by HW after port training is initiated for ports that bifurcate. Hence effective MLW can be smaller than indicated by default
3:0	1h RO	**Max Link Speeds (MLS):** This field indicates the supported link speed(s) of the associated port. 0001b: 2.5 Gb/s link speed is supported 0010b: 5.0 Gb/s and 2.5 Gb/s link speed supported 0011b: 8.0 Gb/s and 5.0 and 2.5 Gb/s link speed supported Others: Reserved. ***Note:*** The encoding is the binary value of the bit location in the Supported Link Speeds Vector (in the Link Capabilities 2 register) that corresponds to the maximum Link speed.

22.10.18 Link Control; Link Status (LCTL_LSTS)—Offset 50h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 10110000h

31			28				24				20				16				12				8				4				0
0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
LABS	LBMS	LA	SCC	LT	RSVD	NLW						CLS				RSVD				LABIE	LBMIE	HAWD	ECPM	ES	CCC	RL	LD	RCBC	RSVD	ASPM		

Bit Range	Default and Access	Field Name (ID): Description
31	0b RO	**Link Autonomous Bandwidth Status (LABS):** This bit is not applicable and hardwired to 0b.
30	0b RO	**Link Bandwidth Management Status (LBMS):** This bit is not applicable and hardwired to 0b.
29	0h RO/V	**Link Active (LA):** Set to 1b when the Data Link Control and Management State Machine is in the DL_Active state, 0b otherwise.
28	1b RW-L	**Slot Clock Configuration (SCC):** When to 1b, this bit indicates that the component uses the same physical reference clock that he platform provides on the connector. If the device uses an independent clock irrespective of the presence of a reference on the connector, this bit must be clear. 1: Indicates same reference clock. 0: Indicates independent reference clock
27	0b RO	**Link Training (LT):** This bit is not applicable for an upstream port and is hardwired to 0.
26	0b RV	**Reserved.**
25:20	1h RO/V	**Negotiated Link Width (NLW):** This field indicates the negotiated width of the PCI Express link. 00 0001b: x1 00 0010b: X2 00 0100b: x4 00 1000b: X8 00 1100b: X12—not supported 01 0000b: X16 10 0000b: X32—not supported All other values are reserved. ***Note:*** The value in this field is undefined when the link is not up.
19:16	1h RO/V	**Current Link Speed (CLS):** This field indicates the negotiated link speed of the given PCI Express link. 0001b: 2.5 Gb/s PCI Express Link 0010b: 5.0 Gb/s PCI Express Link 0011b: 8.0 Gb/s PCI Express Link Others: Reserved ***Note:*** The encoding is the binary value of the bit location in the Supported Link Speeds Vector (in the Link Capabilities 2 register) that corresponds to the current Link speed. ***Note:*** The value in this field is undefined when the link is not up.
15:12	0h RO	**Reserved**.

Bit Range	Default and Access	Field Name (ID): Description
11	0b RO	**Link Autonomous Bandwidth Interrupt Enable (LABIE):** Link Autonomous Bandwidth Interrupt Enable - this is not applicable so hardwired to 0b.
10	0b RO	**Link Bandwidth Management Interrupt Enable (LBMIE):** The uplink does not implement the Link Bandwidth Notification Capability so this is hardwired to 0.
9	0b RW	**Hardware Autonomous Width Disable (HAWD):** When Set, this bit disables hardware from changing the Link width for reasons other than attempting to correct unreliable Link operation by reducing Link width.
8	0b RO	**Enable Clock Power Management (ECPM):** Reserved. Not supported by the upstream port.
7	0h RW	**Extended Synch (ES):** When set, forces extended transmission of FTS ordered sets in FTS and extra TS2 at exit from L1 prior to entering L0.
6	0b RW	**Common Clock Configuration (CCC):** When set, indicates that the root port and device are operating with a distributed common reference clock. A value of 0b indicates that this component and the component at the opposite end of this link are operating with asynchronous reference clock. After changing the value in this bit in both components on a link, software must trigger the link to retrain by writing a 1b to the Retrain Link bit of the downstream port.
5	0b RO	**Retrain Link (RL):** Read only of 0b for an upstream port.
4	0b RO	**Link Disable (LD):** Read only of 0b for an upstream port.
3	0b RO	**Read Completion Boundary Control (RCBC):** Indicates the read completion boundary is 64 bytes.
2	0b RV	**Reserved**.
1:0	00b RW	**Active State Link PM Control (ASPM):** Indicates whether the upstream port should enter L0s or L1 or both. Bits Definition 00 Disabled 01 L0s Entry Enabled 10 L1 Entry Enabled 11 L0s and L1 Entry Enabled

22.10.19 Device Capabilities 2 (DCAP2)—Offset 64h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD												OBFFS			RSVD			TPHCS	LTRMS	RSVD						AFS	CTDS	CTRS			

Bit Range	Default and Access	Field Name (ID): Description
31:20	0h Rv	**Reserved**.
19:18	00b RO	**Optimized Buffer Flush/Fill Supported (OBFFS):** 00b - OBFF is not supported.
17:14	0h RO	**Reserved**.
13:12	00b RO	**TPH Completer Supported (TPHCS):** 00: TPH and Extended TPH Completer not supported
11	0b RW-L	**LTR Mechanism Supported (LTRMS):** A value of 1b indicates support for the optional Latency Tolerance Reporting (LTR) mechanism capability.
10	0b RO	**No RO-enabled PR-PR Passing (NROEPRPASS):** Set to 0b as this feature is not supported.
9	0b RO	**AD128 CAS Completer Supported (AD128ACS):** Not supported
8	0b RO	**AD64-bit AtomicOP Completer Supported (AD64ACS):** Not supported
7	0b RO	**AD32 bit AtomicOP Completere Supported (AD32ACS):** not supported
6	0b RW-L	**AtomicOP Routing Supported (ARS)** This function is not supported. Any writes should be “0”
5	0b RO	**ARI Forwarding Supported (AFS):** Not Supported
4	0b RO	**Completion Timeout Disable Supported (CTDS):** A value of 1b indicates support for the Completion Timeout Disable mechanism.
3:0	0h RO	**Completion Timeout Ranges Supported (CTRS):** Not applicable to an upstream port, set to 0h

22.10.20 Device Control 2; Device Status 2 (DCTL2_DSTS2)—Offset 68h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000000h

31			28				24				20				16				12			8			4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																		OBFFEN		RSVD		LTREN	IDOCE	IDORE	AEB	AFE	AFE	CTD	CTV	

Bit Range	Default and Access	Field Name (ID): Description
31:15	0h RV	Reserved.
14:13	00b RO	**Optimized Buffer Flush/Fill Enable (OBFFEN):** Not supported so RO.
12:11	0h RO	**Reserved**.
10	0h RW	**LTR Mechanism Enable (LTREN):** Not supported
9	0b RO	**IDO Completion Enable (IDOCE):**. Not supported
8	0b RO	**IDO Request Enable (IDORE):** Not supported
7	0b RO	**AtomicOp Egress Blocking (AEB):** Not supported
6	0b RO	**AtomicOp Requester Enable (ARE):** Not supported
5	0b RO	**ARI Forwarding Enable (AFE):** Not applicable
4	0b RO	**Completion Timeout Disable (CTD):** Not supported
3:0	0h RO	**Completion Timeout Value (CTV):** Not supported

22.10.21 Link Capabilities 2 (LCAP2)—Offset 6Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000002h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
RSVD											LSOSRSS					LSOSGSSV							CS	SLSV							RSVD	

Bit Range	Default and Access	Field Name (ID): Description
31:23	0h RV	**Reserved**.
22:16	0h RW-L	**Lower SKP OS Reception Supported Speeds Vector (LSOSRSS):** If this field is non-zero, it indicates that the Port, when operating at the indicated speed(s) supports SRIS and also supports receiving SKP OS at the rate defined for SRNS while running in SRIS. Bit definitions within this field are: Bit 0 2.5 GT/s Bit 1 5.0 GT/s Bit 2 8.0 GT/s Bits 6:3 RsvdP
15:9	0h RW-L	**Lower SKP OS Generation Supported Speeds Vector (LSOSGSSV):** If this field is non-zero, it indicates that the Port, when operating at the indicated speed(s) supports SRIS and also supports software control of the SKP Ordered Set transmission scheduling rate. Bit definitions within this field are: Bit 0 2.5 GT/s Bit 1 5.0 GT/s Bit 2 8.0 GT/s Bits 6:3 RsvdP
8	0b RW-L	**Crosslink Supported (CS):** Not supported
7:1	01h RO/V	**Supported Link Speeds Vector (SLSV):** This field indicates the supported Link speed(s) of the associated Port. For each bit, a value of 1b indicates that the corresponding Link speed is supported; otherwise, the Link speed is not supported. Bit definitions are: bit 1 2.5 GT/s bit 2 5.0 GT/s bit 3 8.0 GT/s Others: Reserved
0	0b RV	**Reserved**.

22.10.22 Link Control 2; Link Status 2 (LCTL2_LSTS2)—Offset 70h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000001h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
RSVD										LER	EQP3S	EQP2S	EQP1S	EqC	CDL	CD				CSOS	EMC	TM			SD	HASD	EC	TLS				

Bit Range	Default and Access	Field Name (ID): Description
31:22	0h RV	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
21	0b RO	**Link Equalization Request (LER):** This bit is set by hardware to request the Link equalization process to be performed on the Link.
20	0b RO	**Equalization Phase 3 Successful (EQP3S):** When set to 1, this bit indicates that Phase 3 of the Transmitter Equalization procedure has successfully completed.
19	0b RO	**Equalization Phase 2 Successful (EQP2S):** When set to 1, this bit indicates that Phase 2 of the Transmitter Equalization procedure has successfully completed.
18	0b RO	**Equalization Phase 1 Successful (EQP1S):** When set to 1, this bit indicates that Phase 1 of the Transmitter Equalization procedure has successfully completed.
17	0b RO	**Equalization Complete (EqC):** When set to 1, this bit indicates that the Transmitter Equalization procedure has completed.
16	0b RO	**Current De-emphasis Level (CDL):** When the Link is operating at 5.0 GT/s speed, this bit reflects the level of de-emphasis. Encodings: 1b -3.5 dB 0b -6 dB The value in this bit is undefined when the Link is not operating at 5.0 GT/s speed.
15:12	0h RO	**Compliance Preset/De-emphasis (CD):** For 8 GT/s and 5 GT/s: This bit sets the transmitter preset level for Polling. Compliance state if the entry occurred due to the Enter Compliance bit being 1b. The Encoding are defined as follows: 8 GT/s Rate: 0000b: -6 dB for de-emphasis, 0 dB for preshoot 0001b: -3.5 dB for de-emphasis, 0 dB for preshoot 0010b: -4.5 dB for de-emphasis, 0 dB for preshoot 0011b: -2.5 dB for de-emphasis, 0 dB for preshoot 0100b: 0 dB for de-emphasis, 0 dB for preshoot 0101b: 0 dB for de-emphasis, 2 dB for preshoot 0110b: 0 dB for de-emphasis, 2.5 dB for preshoot 0111b: -6 dB for de-emphasis, 3.5 dB for preshoot 1000b: -3.5 dB for de-emphasis, 3.5 dB for preshoot 1001b: 0 dB for de-emphasis, 3.5 dB for preshoot 1010b: The Transmitter Preset encoding of 1010b corresponds to the maximum Vtx-boost that the Transmitter advertises with FS and LF in both full swing and reduced swing modes with C_{-1} = 0 and C_{+1} = FS – (2*LF). All other encodings are reserved. 5 GT/s Rate: 001b: -3.5dB 000b: -6 dB All other are reserved. When the link is operating at 2.5 Gb/s, the setting of this bit has no effect.
11	0h RO	**Compliance SOS (CSOS):** When set to 1b, the LTSSM is required to send SKP Ordered Sets periodically in between the (modified) compliance patterns. The default value of this bit is 0b.
10	0b RO	**Enter Modified Compliance (EMC):** When this bit is set to 1b, the device transmits Modified Compliance Pattern if the LTSSM enters Polling.Compliance substate. Default value of this bit is 0b. This register is intended for debug, compliance testing purposes only. System firmware and software is allowed to modify this register only during debug or compliance testing. In all other cases, the system must ensure that this register is set to the default value.

Bit Range	Default and Access	Field Name (ID): Description
9:7	0h RO	**Transmit Margin (TM):** This field controls the value of the non-de-emphasized voltage level at the transmitter pins. This field is reset to 000b on entry to the LTSSM Polling. Configuration substate. 000b: Normal operating range 001b: 800-120 0mV for full swing and 400-700 mV for half-swing Others: Reserved This register is intended for debug, compliance testing purposes only. System firmware and software is allowed to modify this register only during debug or compliance testing. In all other cases, the system must ensure that this register is set to the default value.
6	0b RO	**Selectable De-emphasis (SD):** not supported.
5	0b RO	**Hardware Autonomous Speed Disable (HASD):** When set, this bit disables hardware from changing the link speed for device specific reasons other than attempting to correct unreliable link operation by reducing link speed for device-specific reasons other than attempting to correct unreliable link operations by reducing link speed. Initial transition to the highest supported common link speed is not blocked by this bit.
4	0b RO	**Enter Compliance (EC):** Software is permitted to force a Link to enter Compliance mode at the speed indicated in the Target Link Speed field by setting this bit to 1b in both components on a Link and then initiating a hot reset on the Link. Default value of this bit following Fundamental Reset is 0b. This bit is intended for debug, compliance testing purposes only. System firmware and software is allowed to modify this bit only during debug or compliance testing. In all other cases, the system must ensure that this bit is set to the default value. This bit is intended for debug, compliance testing purposes only. System firmware and software is allowed to modify this bit only during debug or compliance testing. In all other cases, the system must ensure that this bit is set to the default value.
3:0	1h RO	**Target Link Speed (TLS):** For downstream ports and root ports, this field sets an upper limit on link operational speed by restricting the values advertised by the Downstream Port in its training sequences. 0001b: 2.5 Gb/s Target Link Speed 0010b: 5.0 Gb/s Target Link Speed 0011b: 8.0 Gb/s Target Link Speed Others: Reserved ***Note:*** The encoding is the binary value of the bit in the Supported Link Speeds Vector (in the Link Capabilities 2 register) that corresponds to the desired target Link speed. All other encodings are reserved. If a value is written to this field that does not correspond to a speed included in the Supported Link Speeds field, the result is undefined. ***Note:*** The default value of this field is the highest Link speed supported by the component (as reported in the Max Link Speed field of the Link Capabilities register) unless the corresponding platform/form factor requires a different default value. For both Upstream and Downstream Ports, this field is used to set the target compliance mode speed when software is using the Enter Compliance bit to force a Link into compliance mode.

22.10.23 Power Management Capability; PCI Power Management Capabilities (PMCAP_PMC)—Offset 80h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: C8038801h

31			28				24				20				16				12				8				4				0
1	1	0	0	1	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1
PMES					D2S	D1S	AC				DSI	RSVD	PMEC	VS					NEXT								CID					

Bit Range	Default and Access	Field Name (ID): Description
31:27	19h RO	**PME Support (PMES):** Indicates PME# is supported for states D0, $D3_{HOT}$ and $D3_{COLD}$. There is no PME support for $D3_{COLD}$ as the part does not have the capability, but for PCIe compliance purposes, bit 15 is set.
26	0b RO	**D2_Support (D2S):** The D2 state is not supported.
25	0b RO	**D1_Support (D1S):** The D1 state is not supported.
24:22	0h RO	**Aux_Current (AC):** Reports 0mA (self-powered),
21	0b RO	**Device Specific Initialization (DSI):** Indicates that no device-specific initialization is required.
20	0h RO	**Reserved**.
19	0h RO	**PME Clock (PMEC):** Indicates that PCI clock is not required to generate PME#.
18:16	3h RO	**Version (VS):** Indicates support for Revision 1.2 of the PCI Power Management Specification.
15:8	88h RO	**Next Capability (NEXT):** Contains the offset of the next item in the Capabilities List.
7:0	1h RO	**Capability Identifier (CID):** Value of 01h indicates this is a PCI power management capability.

22.10.24 PCI Power Management Control and Status (PMCS)—Offset 84h

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000008h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
RSVD								BPCE	B23S	RSVD							PMES	DSC	DSEL				PMEE	RSVD				NSR	RSVD	PS	

Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RV	**Reserved**
23	0b RO	**Bus Power / Clock Control Enable (BPCE):** Reserved per *PCI Express Specification*.
22	0b RO	**B2/B3 Support (B23S):** Reserved per *PCI Express Specification*.
21:16	0h RV	**Reserved.**
15	0b RW1CS	**PME Status (PMES):** Indicates a PME was received.
14:13	0h RO	**Data Scale (DSC):** Not supported
12:9	0h RO	**Data Select (DSEL):** Not supported
8	0b RWS	**PME Enable (PMEE):** Gates assertion of the PME message.
7:4	0h RV	**Reserved**.
3	1b RWL	**No Soft Reset (NSR):** When set to 1 this bit indicates that devices transitioning from $D3_{HOT}$ to D0 because of Power State commands do not perform an internal reset. Configuration context is preserved. Upon transition from $D3_{HOT}$ to D0 Initialized state, no additional operating system intervention is required to preserve Configuration Context beyond writing the Power State bits. When clear, devices do perform an internal reset upon transitioning from $D3_{HOT}$ to D0 via software control of the Power State bits. Configuration Context is lost when performing the soft reset. Upon transition from $D3_{HOT}$ to D0 state, full reinitialization sequence is needed to return the device to D0 Initialized.
2	0b RV	**Reserved**.
1:0	0h RW	**Power State (PS):** This field is used both to determine the current power state of the upstream port and to set a new power state. The values are: 00 D0 state 11 $D3_{HOT}$ state If software attempts to write a '10' or '01' to these bits, the write will be ignored.

22.10.25 Subsystem Capability List Register (SVCAP)—Offset 88h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 000000Dh

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	1	1	0	1
RSVD																NEXT								CID							

Bit Range	Default and Access	Field Name (ID): Description
31:16	0000h Rv	Reserved.
15:8	00h RO	**Next Capability (NEXT):** For upstream ports there is no MSI structure so the capability is terminated
7:0	0Dh RO	**Capability Identifier (CID):** Value of 0Dh indicates the function as subsystem identification capable.

22.10.26 Subsystem Vendor IDs (SVID)—Offset 8Ch

Size:32 bits

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00008086h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0
SID																SVID															

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RW-L	**Subsystem Identifier (SID):** Indicates the subsystem as identified by the vendor. This field is write once and is locked down until a bridge reset occurs (not the PCI bus reset).
15:0	8086h RW-L	**Subsystem Vendor Identifier (SVID):** Indicates the manufacturer of the subsystem. This field is write once and is locked down until a bridge reset occurs (not the PCI bus reset).

22.10.27 Advanced Error Extended Reporting Capability Header (AECH)—Offset 100h

Size:32 bits
The AER capability can optionally be included or excluded from the capabilities list. The full AER is supported.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 13810001h

31			28				24				20				16				12				8				4				0
0	0	0	1	0	0	1	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
NCO													CV			CID																

Bit Range	Default and Access	Field Name (ID): Description
31:20	138h RW-L	**Next Capability Offset (NCO):** Contains the offset of the next structure in the extended capabilities list.
19:16	1h RO	**Capability Version (CV):** Indicates the version of the capability structure present.
15:0	0001h RO	**Capability ID (CID):** Identifies the function as Advanced Error Reporting capable.

22.10.28 Uncorrectable Error Status (UES)—Offset 104h

This register must maintain its state through a platform reset. It loses its state upon loss of core power.

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD					PLTLPEBS	TPEBS	AEBE	MCE	UIE	AVS	URE	EE	MT	RO	UC	CA	CT	FCPE	PT	RSVD							SDE	DLPE	RSVD			

Bit Range	Default and Access	Field Name (ID): Description
31:27	000h RV	Reserved.
26	0b RO	**Poisoned TLP Egress Blocked Status (PLTLPEBS)**
25	0b RO	**TLP Prefix Blocked Error Status (TPBES)**
24	0b RW1CS	**Atomic Egress Blocked Status (AEBE):** This bit will never be set
23	0b RW1CS	**MC Blocked TLP Status (MCE):** This bit is set whenever a Multicast TLP is blocked
22	0b RW1CS	**Uncorrectable Internal Error Status (UIE):** This bit is set whenever an uncorrectable internal error is detected.
21	0b RO	**ACS Violation Status (AVS):** Reserved, not supported
20	0b RW1CS	**Unsupported Request Error Status (URE):** Indicates an unsupported request was received.
19	0b RW1CS	**ECRC Error Status (EE):** Set to 1b when an ECRC is detected
18	0b RW1CS	**Malformed TLP Status (MT):** Indicates a malformed TLP was received.
17	0b RW1CS	**Receiver Overflow Status (RO):** Indicates a receiver overflow occurred.
16	0b RW1CS	**Unexpected Completion Status (UC):** Indicates an unexpected completion was received.
15	0b RW1CS	**Completor Abort Status (CA):** Indicates a completer abort was received
14	0b RO	**Completion Timeout Status (CT):** not applicable as the upstream port will not issue non-posted requests on it's one behalf.
13	0b RW1CS	**Flow Control Protocol Error Status (FCPE):** This bit is set when a flow control error protocol is detected.
12	0b RW1CS	**Poisoned TLP Status (PT):** Indicates a poisoned TLP was received.
11:6	0h RV	**Reserved.** Software must write '0' to these bits
5	0b RO	**Surprise Down Error Status (SDE):** Surprise Down is not supported.
4	0b RW1CS	**Data Link Protocol Error Status (DLPE):** Indicates a data link protocol error occurred.
3:0	0h RV	**Reserved.**

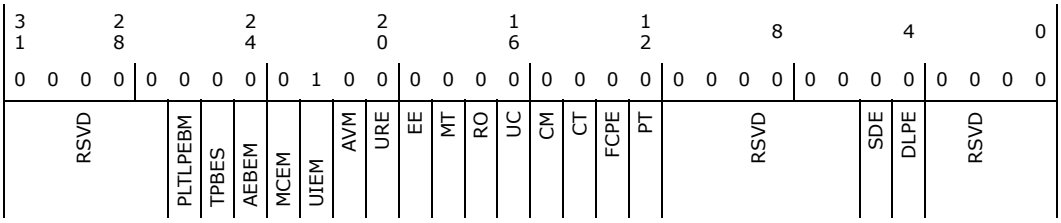
22.10.29 Uncorrectable Error Mask (UEM)—Offset 108h

When set, the corresponding error in the UES register is masked, and the logged error will cause no action. When cleared, the corresponding error is enabled. This register is only reset by a loss of core power.

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00400000h



Bit Range	Default and Access	Field Name (ID): Description
31:27	0h RV	**Reserved.**
26	0b RO	**Poisoned TLP Egress Blocked Mask (PLTLPEBM)**: Not supported
25	0b RO	**TLP Prefix Blocked Error Mask (TPBES)**: not supported
24	0b RWS	**AtomicOP Egress Blocked Mask (AEBEM)**
23	0b RWS	**MC Blocked TLP Mask (MCEM):**
22	1b RWS	**Uncorrectable Internal Error Mask (UIEM)**
21	0b RO	**ACS Violation Mask (AVM):** Reserved. Access Control Services are not supported
20	0b RWS	**Unsupported Request Error Mask (URE):.**
19	0b RWS	**ECRC Error Mask (EE):**
18	0b RWS	**Malformed TLP Mask (MT):**
17	0b RWS	**Receiver Overflow Mask (RO):**
16	0b RWS	**Unexpected Completion Mask (UC):**
15	0b RWS	**Completor Abort Mask (CM):**
14	0b RWS	**Completion Timeout Mask (CT):** Not supported.
13	0b RWS	**Flow Control Protocol Error Mask (FCPE):**

Bit Range	Default and Access	Field Name (ID): Description
12	0b RWS	**Poisoned TLP Mask (PT):**
11:6	0h RV	**Reserved.**
5	0h RO	**Surprise Down Error Mask (SDE):** not supported.
4	0b RWS	**Data Link Protocol Error Mask (DLPE):**
3:0	0h RV	**Reserved.**

22.10.30 Uncorrectable Error Severity (UEV)—Offset 10Ch

This register gives the option to make an uncorrectable error fatal or non-fatal. An error is fatal if the bit is set. An error is non-fatal if the bit is cleared. This register is only reset by a loss of core power.

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00440010h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0
RSVD					PTLBEBS	TPBES	AEBES	MCES	UIES	AVS	URE	EE	MT	RO	UC	CA	CT	FCPE	PT	RSVD							SDE	DLPE	RSVD			

Bit Range	Default and Access	Field Name (ID): Description
31:27	000h RV	**Reserved.**
26	0b RO	**Poisoned TLP Egress Blocked Severity (PTLBEBS)**
25	0b RO	**TLP Prefix Blocked Error Severity (TPBES)**
24	0b RWS	**AtomicOP Egress Blocked Severity (AEBES)**
23	0b RWS	**MC Blocked TLP Severity (MCES)**
22	1b RWS	**Uncorrectable Internal Error Severity (UIES)**
21	0b RO	**ACS Violation Severity (AVS):** Not supported
20	0b RWS	**Unsupported Request Error Severity (URE):**

Bit Range	Default and Access	Field Name (ID): Description
19	0b RWS	**ECRC Error Severity (EE):**
18	1h RWS	**Malformed TLP Severity (MT):**
17	0b RWS	**Receiver Overflow Severity (RO):**
16	0b RWS	**Unexpected Completion Severity (UC):**
15	0b RWS	**Completor Abort Severity (CA):**
14	0b RO	**Completion Timeout Severity (CT):** Not Supported
13	0b RWS	**Flow Control Protocol Error Severity (FCPE):**
12	0b RWS	**Poisoned TLP Severity (PT):**
11:6	0h RV	Reserved.
5	0b RO	**Surprise Down Error Severity (SDE):** Not supported.
4	1b RWS	**Data Link Protocol Error Severity (DLPE):**
3:0	0h RV	Reserved.

22.10.31 Correctable Error Status (CES)—Offset 110h

This register is only reset by a loss of core power.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 28
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																HLOE	CIE	ANFES	RTT	RSVD			RNR	BD	BT	RSVD				RE		

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RV	**Reserved**. Software must write a "0" to these bits.
15	0b RO	**Header Log Overflow Error (HLOE)**
14	0b RO	**Correctable Internal Error (CIE)**
13	0b RW1CS	**Advisory Non-Fatal Error Status (ANFES):**
12	0b RO	**Replay Timer Timeout Status (RTT):** Indicates the replay timer timed out.
11:9	0h RV	**Reserved.** Software must write '0' to these bits
8	0b RO	**Replay Number Rollover Status (RNR):** Indicates the replay number roll over from 11 to 00.
7	0b RW1CS	**Bad DLLP Status (BD):** Indicates a bad DLLP was received.
6	0b RO	**Bad TLP Status (BT):** Indicates a bad TLP was received.
5:1	0h RV	**Reserved**. Software must write '0' to these bits.
0	0b RO	**Receiver Error Status (RE):** Indicates that the physical layer detected a receiver error.

22.10.32 Correctable Error Mask (CEM)—Offset 114h

When set, the corresponding error in the CES register is masked, and the logged error will cause no action. When cleared, the corresponding error is enabled. This register is only reset by a loss of core power.

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00002000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																HLOEM	CIEM	ANFEM	RTT	RSVD			RNR	BD	BT	RSVD				RE		

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RV	Reserved.
15	0b RWS	**Header Log Overflow Error Mask (HLOEM)**
14	0b RWS	**Correctable Internal Error Mask (CIEM)**
13	1b RWS	**Advisory Non-Fatal Error Mask (ANFEM):** When set, masks Advisory Non-Fatal errors from (a) signaling ERR_COR to the device control register and (b) updating the Uncorrectable Error Status register. This register is set by default to enable compatibility with software that does not comprehend Role-Based Error Reporting.
12	0b RWS	**Replay Timer Timeout Mask (RTTEM):**.
11:9	0h RO	**Reserved**.
8	0b RWS	**Replay Number Rollover Mask (RNREM):**
7	0b RWS	**Bad DLLP Mask (BDLLPEM):.**
6	0b RWS	**Bad TLP Mask (BTLPEM):**
5:1	0h RO	**Reserved**.
0	0h RW/P	**Receiver Error Mask (REM):**

22.10.33 Advanced Error Capabilities and Control (AECC)—Offset 118h

This register gives the status and control for ECRC checks, and also the pointer to the first uncorrectable error that happened. This register is only reset by a loss of core power.

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																			CTPHLC	TLPPLP	MHRE	MHRC	ECE	ECC	EGE	EGC	FEP					

Bit Range	Default and Access	Field Name (ID): Description
31:13	0h RV	Reserved.
12	0b RO	**Completion Timeout Prefix/Header Log Capable (CTPHLC):** If Set, this bit indicates that the Function records the prefix/header of Request TLPs that experience a Completion Timeout error.
11	0b ROSV	**TLP Prefix Log Present (TLPPLP):** If Set and the First Error Pointer is valid, indicates that the TLP Prefix Log register contains valid information. If Clear or if First Error Pointer is invalid, the TLP Prefix Log register is undefined.
10	0b RO	**Multiple Header Recording Enable (MHRE):** Multiple Header recording is not supported
9	0b RO	**Multiple Header Recording Capable (MHRC):** Multiple header recording is not supported.
8	0b RWS	**ECRC Check Enable (ECE):**
7	0b RWL	**ECRC Check Capable (ECC):**
6	0b RWS	**ECRC Generation Enable (EGE):**
5	0b RWL	**ECRC Generation Capable (EGC):**.
4:0	0h ROSV	**First Error Pointer (FEP):** This field identifies the bit position of the first error reported in the UncorrectableError Status Register (xref). This register re-arms itself (which does not change its current value) as soon as the error status bit indicated by the pointer is cleared by the software by writing a 1 to that status bit.

22.10.34 Header Log Register (AERHDRLOG[1-4])—Offset 11C - 128h by 4h

This register logs the 4 Dwords of the transaction header for PCI Express Errors.

Type: CFG Register
(Size: 32 bits)

Device:
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
AERHDRLOG																															

Bit Range	Default and Access	Field Name (ID): Description
31:0	00000000h ROSV	**TLP Header Log (TLPHDRLOG)**: As soon as an error is logged in this register, it remains locked for further error-logging until the software clears the status bit that caused the header log, thus rearming the logging of errors.

22.10.35 Access Control Services Extended Capability Header (ACSCAPHDR)—Offset 138h

Size:32 bits

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 0001000Dh

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1
NCO												CV				CID																

Bit Range	Default and Access	Field Name (ID): Description
31:20	000h RW-L	**Next Capability Offset (NCO):** Contains the offset of the next structure in the extended capabilities list.
19:16	1h RO	**Capability Version (CV):** Indicates the version of the capability structure present.
15:0	000Dh RO	**Extended Capability ID (ECID):** Identifies the function as Access Control Services capable.

22.10.36 Access Control Services Capability and Control Register (ACSCAP_ACSCTL)—Offset 13Ch

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 0000005Fh

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1	1	1
RSVD									ACSDTP2PE	ACSP2PECE	ACSUFE	ACSP2CRE	ACSP2PRRE	ACSTBE	ACSSVE	ECVS								RSVD	ACSDTP2P	ACSP2PEC	ACSUF	ACSP2PCR	ACSP2PRR	ACSTB	ACSSV

Bit Range	Default and Access	Field Name (ID): Description
31:23	0000h RO	Reserved
22	0b RW	**ACS Direct Translated P2P Enable (T) (ACSP2PECE):** When Set, overrides the ACS P2P Request Redirect and ACS P2P Egress Control mechanisms with peer-to-peer Memory Requests whose Address Translation (AT) field indicates a Translated address ***Note:*** This bit is ignored if ACS Translation Blocking (B) is enabled.
21	0b RO	**ACS P2P Egress Control Enable (E) (ACSP2PECE):** This is hardwired to 0b as the component does not implement ACS P2P Egress Control.
20	0b RW	**ACS Upstream Forwarding Enable (U) (ACSUFE):** When set, the component forwards upstream any Request or Completion TLPs it receives that were redirected upstream by a component lower in the hierarchy. Note that the U bit only applies to upstream TLPs arriving at a Downstream Port, and whose normal routing targets the same Downstream Port.
19	0b RW	**ACS P2P Completion Redirect Enable (C) (ACSP2PCRE):** Determines when the component redirects peer-to-peer Completions upstream; applicable only to Read Completions whose Relaxed Ordering Attribute is clear.
18	0b RW	**ACS P2P Request Redirect Enable (R) (ACSP2PRRE):** This bit determines when the component redirects peer-to-peer Requests upstream.
17	0b RW	**ACS Translation Blocking Enable (B) (ACSTBE):** When set, the component blocks all upstream Memory Requests whose Address Translation (AT) field is not set to the default value.
16	0b RW	**ACS Source Validation Enable (V) (ACSSVE):** When set, the component validates the Bus Number from the Requester ID of upstream Requests against the secondary / subordinate Bus Numbers.
15:8	00h RO	**Egress Control Vector Size (ECVS):** Indicates the number of bits in the Egress Control Vector. This is set to 00h as ACS P2P Egress Control (ACSP2PEC) bit 5 in this register is 0b.
7	0b RV	Reserved
6	1b RO	**ACS Direct Translated P2P (T) (ACSDTP2P):** Indicates that the component does implement ACS Direct Translated P2P. ***Note:*** Required for Root Ports that support Address Translation Services (ATS) and also support peer-to-peer traffic with other Root Ports; required for Switch Downstream Ports
5	0b RO	**ACS P2P Egress Control (E) (ACSP2PEC):** Hardwired to 0. Indicates that the component does not implement ACS P2P Egress Control.
4	1b RO	**ACS Upstream Forwarding (U) (ACSUF):** Indicates that the component implements ACS Upstream Forwarding.
3	1b RO	**ACS P2P Completion Redirect (C) (ACSP2PCR):** Indicates that the component implements ACS P2P Completion Redirect.
2	1b RO	**ACS P2P Request Redirect (R) (ACSP2PRR):** Indicates that the component implements ACS P2P Request Redirect.
1	1b RO	**ACS Translation Blocking (B) (ACSTB):** Indicates that the component implements ACS Translation Blocking.
0	1b RO	**ACS Source Validation (V) (ACSSV):** Indicates that the component implements ACS Source Validation.

22.10.37 Uncorrectable Error Detect Mask Register (ERRUNCDETMSK)—Offset 140h

A value of 1 masks the field and prevents detection as well as logging of errors.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 0
Function: 0

Default: 04000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD					PTLPEDM	TPBEDM	AEBEDM	MCEDM	UIEDM	ACSEDM	UREDM	ECRCEDM	MTLPEDM	ROEDM	UCEDM	CAEDM	CTEDM	FCEDM	PTLPEDM	RSVD				SDEDM	DLPEDM	RSVD							

Bit Range	Default and Access	Field Name (ID): Description
31:27	000h RV	**Reserved**.
26	1b RWS	**Poisoned TLP Egress Blocked Detect Mask (PLTPEBDM)**
25	0b RO	**TLP Prefix Blocked Error Detect Mask (TPBEDM):** Not Supported
24	0b RWS	**AtomicOP Egress Blocked Error Detect Mask (AEBEDM)**
23	0b RO	**MC Blocked TLP Error Detect Mask (MCEDM)**
22	0b RWS	**Uncorrectable Internal Error Detect Mask (UIEDM)**
21	0b RWS	**ACS Violation Error Detect Mask (ACSEDM)**: Not Supported
20	0b RWS	**Unsupported Request Error Detect Mask (UREDM)**
19	0b RO	**ECRC Check Error Mask (ECRCEDM)**
18	0b RWS	**Malformed TLP Error Detect Mask (MTLPEDM)**
17	0b RWS	**Receiver Overflow Error Detect Mask (ROEDM)**
16	0b RWS	**Unexpected Completion Error Detect Mask (UCEDM)**
15	0b RWS	**Completer Abort Error Detect Mask (CAEDM)**
14	0b RWS	**Completion Timeout Error Detect Mask (CTEDM)**
13	0b RWS	**Flow Control Error Detect Mask (FCEDM)**

Bit Range	Default and Access	Field Name (ID): Description
12	0b RWS	**Poisoned TLP Error Detect Mask (PTLPEDM)**
11:6	00h RV	**Reserved**
5	0b RO	**Surprise Down Error Detect Mask (SLDEDM):** Not Supported
4	0b RWS	**Data Link Protocol Error Detect Mask (DLPEDM)**
3:0	000b RV	**Reserved**

22.10.38 Correctable Error Detect Mask Register (ERRCORDETMSK)—Offset 144h

Type: CFG Register
(Size: 32 bits)

Device:
Function: 0

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																HLOEDM	CIEDM	ANFEDM	RTTEDM	RSVD			RNREDM	BDLLPEDM	BTLPEDM	RSVD					REDM	

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RV	**Reserved**.
15	0b RO	**Header Log Overflow Error Detect Mask (HLOEDM)**
14	0b RO	**Correctable Internal Error DectMask (CIEDM)**
13	0b RWS	**Advisory Non-Fatal Error Detect Mask (ANFEDM):**
12	0b RO	**Replay Timer Timeout Detect Mask (RTTEDM):**.
11:9	0h Rv	**Reserved.**
8	0b RO	**Replay Number Rollover Detect Mask (RNREDM):**
7	0b RO	**Bad DLLP Detect Mask (BDLLPEDM):**

Bit Range	Default and Access	Field Name (ID): Description
6	0b RO	**Bad TLP Detect Mask (BTLPEDM):**
5:1	0h Rv	**Reserved**.
0	0h R0	**Receiver Error Detect Mask (REDM):**

§

23 Real Time Clock

23.1 Acronyms

Acronyms	Description
GPI	General Purpose Input
RAM	Random Access Memory
RTC	Real Time Clock
UIP	Update in Progress

23.2 References

None

23.3 Overview

The PCH contains a Motorola* MC146818B-compatible real-time clock with 256 bytes of battery-backed RAM. The real-time clock performs two key functions—keeping track of the time of day and storing system data, even when the system is powered down. The RTC operates on a 32.768 kHz crystal and a 3V battery.

The RTC also supports two lockable memory ranges. By setting bits in the configuration space, two 8-byte ranges can be locked to read and write accesses. This prevents unauthorized reading of passwords or other system security information.

The RTC also supports a date alarm that allows for scheduling a wake up event up to 30 days in advance, rather than just 24 hours in advance.

23.4 Signal Description

Name	Type	Description
RTCX1	I	**Crystal Input 1:** This signal is connected to the 32.768 kHz crystal (Max 50K ESR). If using an external oscillator, the RTCX1 Vih must be within the range of 0.8V to 1.5V (1.5V max).
RTCX2	O	**Crystal Input 2:** This signal is connected to the 32.768 kHz crystal (Max 50K ESR). If using an external oscillator, RTCX2 should be left floating.
RTCRST#	I	An external RC circuit creates a time delay for the signal such that it will go high sometime after battery voltage is valid. If the battery is missing/weak, this signal can appear to be low (asserted) at boot just after the Vcc3P3 power rail is up since it will not have time to meet Vih when Vcc3P3 is high. Upon booting, BIOS should recognize that RTCRST# was asserted and clear CMOS RAM. 1. Unless CMOS is being cleared (only to be done in the G3 power state) or the battery is low, the signal input must always be high when RTC power is on. 2. In the case where the RTC battery is dead or missing on the platform, the signal should go high (de-asserted) before DSW_PWROK or RSMRST# go high (de-assert).

Name	Type	Description
SRTCRST#	I	**Secondary RTC Reset:** This signal resets the manageability register bits in the RTC well when the RTC battery is removed. An external RC circuit creates a time delay for the signal such that it will go high (de-assert) sometime after the battery voltage is valid. When asserted, this signal resets all register bits in the RTC well. ***Notes:*** 1. Unless CMOS is being cleared (only to be done in the G3 power state) the signal input must always be high when RTC power is on. 2. In the case where the RTC battery is dead or missing on the platform, the signal should go high (de-asserted) before DSW_PWROK or RSMRST# go high (de-assert).

23.5 Integrated Pull-Ups and Pull-Downs

None

23.6 I/O Signal Planes and States

Signal Name	Power Plane	During Reset	Immediately after Reset	S3/S4/S5	Deep Sx
RTCRST#	RTC	Undriven	Undriven	Undriven	Undriven
SRTCRST#	RTC	Undriven	Undriven	Undriven	Undriven

23.7 Functional Description

The Real Time Clock module provides a battery backed-up date and time keeping device with two banks of static RAM with 128 bytes each, although the first bank has 114 bytes for general purpose usage.

Three interrupt features are available: time of day alarm with once a second to once a month range, periodic rates of 122 – 500 ms, and end of update cycle notification. Seconds, minutes, hours, days, day of week, month, and year are counted. Daylight savings compensation is no longer supported.

The hour is represented in twelve or twenty-four hour format, and data can be represented in BCD or binary format. The design is functionally compatible with the Motorola MS146818B. The time keeping comes from a 32.768 kHz oscillating source, which is divided to achieve an update every second. The lower 14 bytes on the lower RAM block has very specific functions. The first ten are for time and date information. The next four (0Ah to 0Dh) are registers, which configure and report RTC functions.

The time and calendar data should match the data mode (BCD or binary) and hour mode (12 or 24 hour) as selected in register B. It is up to the programmer to make sure that data stored in these locations is within the reasonable values ranges and represents a possible date and time. The exception to these ranges is to store a value of C0–FFh in the Alarm bytes to indicate a don't care situation. All Alarm conditions must match to trigger an Alarm Flag, which could trigger an Alarm Interrupt if enabled.

The SET bit must be 1 while programming these locations to avoid clashes with an update cycle. Access to time and date information is done through the RAM locations. If a RAM read from the ten time and date bytes is attempted during an update cycle, the value read do not necessarily represent the true contents of those locations. Any RAM writes under the same conditions are ignored.

Note: The leap year determination for adding a 29th day to February does not take into account the end-of-the-century exceptions. The logic simply assumes that all years divisible by four are leap years. According to the Royal Observatory Greenwich, years that are divisible by 100 are typically not leap years. In every fourth century (years divisible by 400, like 2000), the 100-year-exception is over-ridden and a leap-year occurs.

Note: The year 2100 will be the first time in which the current RTC implementation would incorrectly calculate the leap-year.

The PCH does not implement month/year alarms.

23.7.1 Update Cycles

An update cycle occurs once a second, if the SET bit of register B is not asserted and the divide chain is properly configured. During this procedure, the stored time and date are incremented, overflow is checked, a matching alarm condition is checked, and the time and date are rewritten to the RAM locations.

The update cycle will start at least 488 µs after the UIP bit of register A is asserted, and the entire cycle does not take more than 1984 µs to complete. The time and date RAM locations (0–9) are disconnected from the external bus during this time.

To avoid update and data corruption conditions, external RAM access to these locations can safely occur at two times. When a updated-ended interrupt is detected, almost 999 ms is available to read and write the valid time and date data. If the UIP bit of Register A is detected to be low, there is at least 488 µs before the update cycle begins.

Warning: The overflow conditions for leap years adjustments are based on more than one date or time item. To ensure proper operation when adjusting the time, the new time and data values should be set at least two seconds before leap year occurs.

23.7.2 Interrupts

The real-time clock interrupt is internally routed within the PCH both to the I/O APIC and the 8259. It is mapped to interrupt vector 8. This interrupt does not leave the PCH, nor is it shared with any other interrupt. IRQ8# from the SERIRQ stream is ignored. However, the High Performance Event Timers can also be mapped to IRQ8#; in this case, the RTC interrupt is blocked.

23.7.3 Lockable RAM Ranges

The RTC battery-backed RAM supports two 8-byte ranges that can be locked using the configuration space. If the locking bits are set, the corresponding range in the RAM will not be readable or writable. A write cycle to those locations will have no effect. A read cycle to those locations will not return the location’s actual value (resultant value is undefined).

Once a range is locked, the range can be unlocked only by a hard reset, which will invoke the BIOS and allow it to relock the RAM range.

23.7.4 Century Rollover

The PCH detects a rollover when the Year byte transitions form 99 to 00. Upon detecting the rollover, the PCH sets the NEWCENTURY_STS bit.

If the system is in an S0 state, this causes an SMI#. The SMI# handler can update registers in the RTC RAM that are associated with century value.

If the system is in a sleep state (S1–S5) when the century rollover occurs, the PCH also sets the NEWCENTURY_STS bit, but no SMI# is generated. When the system resumes from the sleep state, BIOS should check the NEWCENTURY_STS bit and update the century value in the RTC RAM.

23.7.5 Clearing Battery-Backed RTC RAM

Clearing RTC well registers in a PCH-based platform can be done by using a jumper on RTCRST# or GPI. Implementations should not attempt to clear CMOS by using a jumper to pull VccRTC low. The RTC RAM contents are themselves not cleared by RTCRST#. To clear them to 0b, they must be written to.

23.7.5.1 Using RTCRST# to Clear CMOS

A jumper on RTCRST# can be used to clear CMOS values, as well as reset to default, the state of those configuration bits that reside in the RTC power well.

When the RTCRST# is strapped to ground, the RTC_PWR_STS bit will be set and those configuration bits in the RTC power well will be set to their default state. BIOS can monitor the state of this bit and manually clear the RTC CMOS array once the system is booted. The normal position would cause RTCRST# to be pulled up through a weak pull-up resistor. This RTCRST# jumper technique allows the jumper to be moved and then replaced—all while the system is powered off. Then, once booted, the RTC_PWR_STS can be detected in the set state.

23.7.5.2 Using a GPI to Clear CMOS

A jumper on a GPI can also be used to clear CMOS values. The BIOS would detect the setting of this GPI on system boot-up, and manually clear the CMOS array.

Note: The GPI strap technique to clear CMOS requires multiple steps to implement. The system is booted with the jumper in new position, then powered back down. The jumper is replaced back to the normal position, then the system is rebooted again.

Warning: Do not implement a jumper on VccRTC to clear CMOS.

23.7.6 External RTC Circuitry

The PCH implements an internal oscillator circuit that is sensitive to step voltage changes in VCCRTC. Refer to the Platform Design Guide for example external RTC circuitry guidelines.

Table 23-1. RTC Crystal Requirements

Parameter	Specification
Frequency	32.768 kHz
Typical Tolerance	20 ppm or better
ESR	≤ 50 kΩ

Table 23-2. External Crystal Oscillator Requirements

Parameter	Specification
Frequency	32.768 kHz
Typical Tolerance	20 ppm or better
RTCX1 Vin requirements	Vih min = 0.8V Vih max = 1.5V Vil min = -0.2V Vil max = 0.2V

23.8 RTC Indexed Registers Summary

The RTC contains two sets of indexed registers that are accessed using the two separate Index and Target registers (70h/71h or 72h/73h), as shown in the following table:

RTC (Standard) RAM Bank
Index Name
00h Seconds
01h Seconds Alarm
02h Minutes
03h Minutes Alarm
04h Hours
05h Hours Alarm
06h Day of Week
07h Day of Month
08h Month
09h Year
0Ah Register A
0Bh Register B
0Ch Register C
0Dh Register D
0Eh-7Fh 114 Bytes of User RAM

Table 23-3. Summary of RTC Indexed Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
1h	1h	Seconds Alarm (Sec_Alarm)—Offset 1h	0h
2h	2h	Minutes (Minutes)—Offset 2h	0h
3h	3h	Minutes Alarm (Minutes_Alarm)—Offset 3h	0h
4h	4h	Hours (Hours)—Offset 4h	0h
5h	5h	Hours Alarm (Hours_Alarm)—Offset 5h	0h
6h	6h	Day of Week (Day_of_Week)—Offset 6h	0h
7h	7h	Day of Month (Day_of_Month)—Offset 7h	0h
8h	8h	Month (Month)—Offset 8h	0h
9h	9h	Year (Year)—Offset 9h	0h
Ah	Ah	Register A (RTC_REGA)—Offset Ah	20h
Bh	Bh	Register B - General Configuration (Register_B)—Offset Bh	0h
Ch	Ch	Register C - Flag Register (Register_C)—Offset Ch	0h
Dh	Dh	Register D - Flag Register (Register_D)—Offset Dh	80h

23.8.1 Seconds Alarm (Sec_Alarm)—Offset 1h

Seconds Alarm

23.8.2 Minutes (Minutes)—Offset 2h

Minutes

23.8.3 Minutes Alarm (Minutes_Alarm)—Offset 3h

Minutes Alarm

23.8.4 Hours (Hours)—Offset 4h

Hours

23.8.5 Hours Alarm (Hours_Alarm)—Offset 5h

Hours Alarm

23.8.6 Day of Week (Day_of_Week)—Offset 6h

Day of Week

23.8.7 Day of Month (Day_of_Month)—Offset 7h

Day of Month

23.8.8 Month (Month)—Offset 8h

Month

23.8.9 Year (Year)—Offset 9h

Year

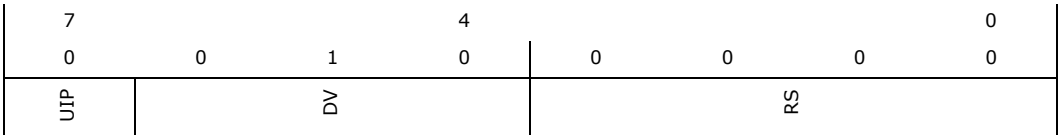
23.8.10 Register A (RTC_REGA)—Offset Ah

This register is used for general configuration of the RTC functions. None of the bits are affected by RSMRST# or any other reset signal.

Access Method

Type: Register (Size: 8 bits)	**Device:** **Function:**

Default: 20h



Bit Range	Default and Access	Field Name (ID): Description
7	0h RW	**UPDATE IN PROGRESS (UIP):** This bit may be monitored as a status flag. 0 = Update cycle will not start for at least 488 micro-seconds. The time, calendar, and alarm information in RAM is always available when the UIP bit is 0. 1 = The update is soon to occur or is in progress.
6:4	2h RW	**Division Chain Select (DV):** These three bits control the divider chain for the oscillator, and are not affected by RSMRST# or any other reset signal.DV2 corresponds to bit 6. DV2 DV1 DV0 Function 0 1 0 Normal Operation 1 1 X Divider Reset 1 0 1 Bypass 15 stages (test mode only) 1 0 0 Bypass 10 stages (test mode only) 0 1 1 Bypass 5 stages (test mode only) 0 0 1 Invalid 0 0 0 Invalid
3:0	0h RW	**Rate Select (RS):** Selects one of 13 taps of the 15 stage divider chain. The selected tap can generate a periodic interrupt if the PIE bit is set in Register B. Otherwise this tap will set the PF flag of Register C. If the periodic interrupt is not to be used, these bits should all be set to zero. RS3 corresponds to bit 3. RS3[4p] RS2 RS1 RS0 Periodic Rate 0 0 0 0 Interrupt never toggles 0 0 0 1 3.90625 ms 0 0 1 0 7.8125 ms 0 0 1 1 122.070 s 0 1 0 0 244.141 s 0 1 0 1 488.281 s 0 1 1 0 976.5625s 0 1 1 1 1.953125 ms 1 0 0 0 3.90625 ms 1 0 0 1 7.8125 ms 1 0 1 0 15.625 ms 1 0 1 1 31.25 ms 1 1 0 0 62.5 ms 1 1 0 1 125 ms 1 1 1 0 250 ms 1 1 1 1 500 ms

23.8.11 Register B - General Configuration (Register_B)—Offset Bh

Access Method

Type: Register
(Size: 8 bits)

Device:
Function:

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
SET	PIE	AIE	UIE	SQWE	DM	HOURFORM	DSE

Bit Range	Default and Access	Field Name (ID): Description
7	0h RW	**Update Cycle Inhibit (SET):** Enables/Inhibits the update cycles. 0 = Update cycle occurs normally once each second. 1 = A current update cycle will abort and subsequent update cycles will not occur until SET is returned to zero. When set is one, the BIOS may initialize time and calendar bytes safely. This bit is not affected by RSMRST# nor any other reset signal. ***Note:*** Software must ensure this bit is at least transitioned from '1' to '0' once whenever the RTC coin battery is inserted. This is to ensure that the internal RTC time updates occur properly.
6	0h RW	**Periodic Interrupt Enable (PIE):** 0 = Disabled. 1 = Enabled. Allows an interrupt to occur with a time base set with the RS bits of register A. This bit is cleared by RSMRST# assertion, but not on any other reset.
5	0h RW	**Alarm Interrupt Enable (AIE):** 0 = Disabled. 1 = Enabled. Allows an interrupt to occur when the AF is set from an alarm match from the update cycle. An alarm can occur once a second, one an hour, once a day, or once a month. This bit is cleared by RTCRST# assertion, but not on any other reset.
4	0h RW	**Update-ended Interrupt Enable: (UIE):** 0 = Disabled. 1 = Enabled. Allows an interrupt to occur when the update cycle ends. This bit is cleared by RSMRST# assertion, but not on any other reset.
3	0h RW	**Square Wave Enable (SQWE):** The Square Wave Enable bit serves no function in this device, yet is left in this register bank to provide compatibility with the Motorola 146818B. There is not SQW pin on this device. This bit is cleared by RSMRST# assertion, but not on any other reset.
2	0h RW	**Data Mode (DM):** This bit specifies either binary or BCD data representation. 0 = BCD. 1 = Binary. This bit is not affected by RSMRST# nor any other reset signal.
1	0h RW	**Hour Format (HOURFORM):** This bit indicates the hour byte format. 0 = Twelve-hour mode is selected.In twelve hour mode, the seventh bit represents AM as zero and PM as one. 1 = Twenty-four hour mode is selected. This bit is not affected by RSMRST# nor any other reset signal.
0	0h RW	**Daylight Savings Enable (DSE):** The Daylight Savings Enable bit triggers two special hour updates per year when set to one. One is on the first Sunday in April, where time increments from 1:59:59 AM to 3:00:00 AM. The other is the last Sunday in October when the time first reaches 1:59:59 AM, it is changed to 1:00:00 AM. The time must increment normally for at least two update cycles (seconds) previous to these conditions for the time change to occur properly. These special update conditions do not occur when the DSE bit is set to zero. The days for the hour adjustment are those specified in United States federal law as of 1987, which is different than previous years. If BUC.DSO bit is set, the DSE bit continues to be a R/W bit, but Daylight Saving is disabled regardless of the DSE bit. This bit is not affected by RSMRST# nor any other reset signal.

23.8.12 Register C - Flag Register (Register_C)—Offset Ch

Access Method

Type: Register
(Size: 8 bits)

Device:
Function:

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
IRQF	PF	AF	UF	RSVD			

Bit Range	Default and Access	Field Name (ID): Description
7	0h RO	**Interrupt Request Flag (IRQF):** Interrupt Request Flag = (PF * PIE) + (AF * AIE) + (UF * UFE). This also causes the RTC Interrupt to be asserted. This bit is cleared upon RSMRST# assertion or a read of Register C.
6	0h RO	**Periodic Interrupt Flag (PF):** Periodic interrupt Flag will be one when the tap as specified by the RS bits of register A is one. If no taps are specified, this flag bit will remain at zero. This bit is cleared upon RSMRST# assertion or a read of Register C.
5	0h RO	**Alarm Flag (AF):** Alarm Flag will be high after all Alarm values match the current time. This bit is cleared upon RTCRST# assertion or a read of Register C.
4	0h RO	**Update-ended Flag (UF):** Updated-ended flag will be high immediately following an update cycle for each second. The bit is cleared upon RSMRST# assertion or a read of Register C.
3:0	0h RO	Reserved.

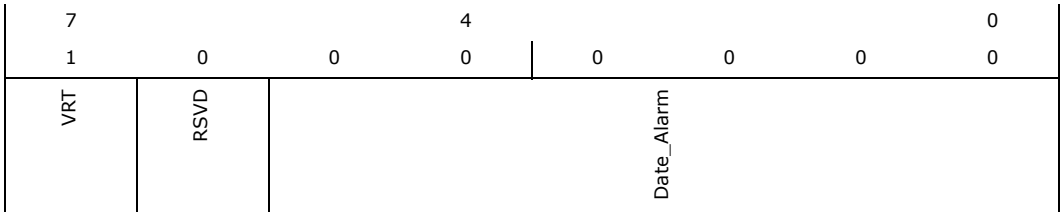
23.8.13 Register D - Flag Register (Register_D)—Offset Dh

Access Method

Type: Register
(Size: 8 bits)

Device:
Function:

Default: 80h



Bit Range	Default and Access	Field Name (ID): Description
7	1h RW	**Valid RAM and Time Bit (VRT):** This bit is hard-wired to 1 in the RTC power well. This bit should always be written as a 0 for write cycle, however it will return a 1 for read cycles.
6	0h RO	Reserved.
5:0	0h RW	**Date Alarm (Date_Alarm):** These bits store the date of month alarm value. If set to 000000, then a don't care state is assumed. The host must configure the dates alarm for these bits to do anything, yet they can be written at any time. If the date alarm is not enabled, these bits will return zeros to mimic the functionality of the Motorola 146818B. These bits are not affected by any reset assertion.

23.9 RTC Configuration Registers Summary

Table 23-4. Summary of RTC Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
3400h	3403h	RTC Configuration (RC)—Offset 3400h	0h
3414h	3417h	Backed Up Control (BUC)—Offset 3414h	0h

23.9.1 RTC Configuration (RC)—Offset 3400h

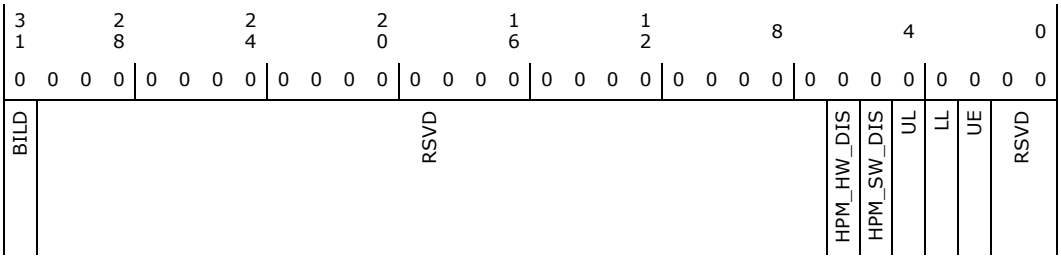
All bits in this register are in the Primary Well and cleared by PLTRST# assertion.

Access Method

Type: Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RWLO	**BIOS Interface Lock-Down (BILD):** When set, prevents RTC version of TS (BUC.TS) from being changed. This bit can only be written from 0 to 1 once. This BILD bit has a different function compared to LPC, SPI and eSPI version but BIOS should set all the corresponding bits after reset in order to lock down the BIOS interface correctly.
30:7	0h RO	Reserved.
6	0h RW	**RTC High Power Mode HW Disable (HPM_HW_DIS):** 0 = HW control of the RTC internal VRM is disabled. 1 = The internal VRM that generates the RTC well supply voltage in SUS mode is disabled when SLP_S0# is asserted.
5	0h RW	**RTC High Power Mode SW Disable (HPM_SW_DIS):** 0 = The internal VRM powers the RTC well when RSMRST# is '1'. (default) 1 = The internal VRM that generates the rtc well supply voltage in SUS mode is disabled.
4	0h RWLO	**Upper 128 Byte Lock (UL):** When set, bytes 38h-3Fh in the upper 128 byte bank of RTC RAM are locked and cannot be accessed. Writes will be dropped and reads will not return any guaranteed data. Bit reset on system reset.
3	0h RWLO	**Lower 128 Byte Lock (LL):** When set, bytes 38h-3Fh in the lower 128 byte bank of RTC RAM are locked and cannot be accessed. Writes will be dropped and reads will not return any guaranteed data. Bit reset on system reset.
2	0h RW	**Upper 128 Byte Enable (UE):** When set, the upper 128 byte bank of RTC RAM can be accessed.
1:0	0h RO	Reserved.

23.9.2 Backed Up Control (BUC)—Offset 3414h

All bits in this register are in the RTC well and only cleared by RTCRST# assertion.

Access Method

Type: Register
(Size: 8 bits)

Device:
Function:

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
RSVD		LanDisable	DSO	RSVD			TS

Bit Range	Default and Access	Field Name (ID): Description
7:6	0h RO	Reserved.
5	0h RW/RO/F/L	**LAN Disable (LanDisable):** 0 = LAN is Enabled 1 = LAN is Disabled. This register is reset to 0 on RTCRST# assertion. If the Function Disable SUS Well Lockdown register is set, this bit can not be changed by software.
4	0h RW	**Daylight Savings Override (DSO):** 0 = The DSE bit in the RTC register B bit(0) is a RW bit that is configurable by software to enable the daylight savings. 1 = The DSE bit in the RTC Register B bit(0) is a RW bit but has no effect, daylight savings is hard-disabled internally. System BIOS shall configure this bit accordingly during the boot process before RTC time is initialized.
3	0h RW	**Reserved**
2:1	0h RO	**Reserved.**
0	0h RW	**Top Swap (TS):** 0 = PCH will not invert A16, A17 or A18. 1 = PCH will invert A16, A17 or A18 for cycles going to the BIOS space. If booting from SPI LPC (FWH), then the boot-block size is 64 KB and A16 is inverted if Top Swap is enabled. If booting from SPI, then the Top Swap Block size soft strap determines if A16, A17 or A18 should be inverted if Top Swap is enabled. If PCH is strapped for Top Swap (GPP_B14/SPKR is high at rising edge of PCH_PWROK), then this bit cannot be cleared by software. The strap jumper should be removed and the system rebooted.

§

24 Serial ATA

The PCH has two integrated Serial ATA host controllers with independent DMA operation on up to14 ports.

24.1 Acronyms

Acronyms	Description
AHCI	Advanced Host Controller Interface
DMA	Direct Memory Access
DEVSLP	Device Sleep
IDE	Integrated Drive Electronics
RAID	Redundant Array of Independent Disks
SATA and SSATA	Serial Advanced Technology Attachment
Intel® RSTe	Intel® Rapid Storage Technology Enterprise

24.2 References

Specification	Location
Serial ATA Specification, Revision 3.2	https://www.sata-io.org
Serial ATA II: Extensions to Serial ATA 1.0, Revision 1.0	https://www.sata-io.org
Serial ATA II Cables and Connectors Volume 2 Gold	https://www.sata-io.org

24.3 Overview

The PCH has two integrated SATA host controller that supports independent DMA operation for up tofourteen ports for the Intel® C620 Series Chipsetand supports data transfer rates of up to 6 Gb/s on all ports. The fourteen ports are split between eight ports on one controller and six on the other. The SATA ports on the second controller are named SSATA ports to identify that they are at a different configuration space and address as the first eight, which are compatible with the client. Other than the difference in the number of SATA ports and residing at different Device:Function numbers, the two SATA controller are identical in capabilities and programming.

The PCH SATA controller contains one mode of operation—AHCI mode using memory space, and it also supports RAID mode. The PCH SATA controller no longer supports legacy mode using I/O space. Therefore, AHCI software is required. The PCH SATA controller supports the *Serial ATA Specification*, Revision 3.2.

See Section 1.3, “PCH I/O Features and Capabilities” for details on feature availability.

24.4 Signal Description

Name	Type	Description
GPP_E4_SATA_DEVSLP0	OD	**Serial ATA Port [0] Device Sleep**: This is an open-drain pin on the PCH side. PCH will tri-state this pin to signal to the SATA device that it may enter a lower power state (pin will go high due to pull-up that is internal to the SATA device, per DEVSLP specification). PCH will drive pin low to signal an exit from DEVSLP state. Design Constraint: As per the PDG, no external pull-up or pull-down termination required when used as DEVSLP. ***Note:*** This pin can be mapped to SATA Port 0.
GPP_E5_SATA_DEVSLP1	OD	**Serial ATA Port [1] Device Sleep:** This is an open-drain pin on the PCH side. PCH will tri-state this pin to signal to the SATA device that it may enter a lower power state (pin will go high due to pull-up that is internal to the SATA device, per DEVSLP specification). PCH will drive pin low to signal an exit from DEVSLP state. Design Constraint: As per the PDG, no external pull-up or pull-down termination required when used as DEVSLP. ***Note:*** This pin can be mapped to SATA Port 1.
GPP_E6_SATA_DEVSLP2	OD	**Serial ATA Port [2] Device Sleep:** This is an open-drain pin on the PCH side. PCH will tri-state this pin to signal to the SATA device that it may enter a lower power state (pin will go high due to pull-up that is internal to the SATA device, per DEVSLP specification). PCH will drive pin low to signal an exit from DEVSLP state. Design Constraint: As per the PDG, no external pull-up or pull-down termination required when used as DEVSLP. ***Note:*** This pin can be mapped to SATA Port 2.
GPP_F5_SATA_DEVSLP3	OD	**Serial ATA Port [3] Device Sleep**: This is an open-drain pin on the PCH side. PCH will tri-state this pin to signal to the SATA device that it may enter a lower power state (pin will go high due to pull-up that is internal to the SATA device, per DEVSLP specification). PCH will drive pin low to signal an exit from DEVSLP state. Design Constraint: As per the PDG, no external pull-up or pull-down termination required when used as DEVSLP. ***Note:*** This pin can be mapped to SATA Port 3.
GPP_F6_SATA_DEVSLP4	OD	**Serial ATA Port [4] Device Sleep**: This is an open-drain pin on the PCH side. PCH will tri-state this pin to signal to the SATA device that it may enter a lower power state (pin will go high due to pull-up that is internal to the SATA device, per DEVSLP specification). PCH will drive pin low to signal an exit from DEVSLP state. Design Constraint: As per the PDG, no external pull-up or pull-down termination required when used as DEVSLP. ***Note:*** This pin can be mapped to SATA Port 4.
GPP_F7_SATA_DEVSLP5	OD	**Serial ATA Port [5] Device Sleep**: This is an open-drain pin on the PCH side. PCH will tri-state this pin to signal to the SATA device that it may enter a lower power state (pin will go high due to pull-up that is internal to the SATA device, per DEVSLP specification). PCH will drive pin low to signal an exit from DEVSLP state. Design Constraint: As per the PDG, no external pull-up or pull-down termination required when used as DEVSLP. ***Note:*** This pin can be mapped to SATA Port 5.
GPP_F8_SATA_DEVSLP6	OD	**Serial ATA Port [6] Device Sleep**: This is an open-drain pin on the PCH side. PCH will tri-state this pin to signal to the SATA device that it may enter a lower power state (pin will go high due to pull-up that is internal to the SATA device, per DEVSLP specification). PCH will drive pin low to signal an exit from DEVSLP state. Design Constraint: As per the PDG, no external pull-up or pull-down termination required when used as DEVSLP. ***Note:*** This pin can be mapped to SATA Port 6.
GPP_F9_SATA_DEVSLP7	OD	**Serial ATA Port [7] Device Sleep**: This is an open-drain pin on the PCH side. PCH will tri-state this pin to signal to the SATA device that it may enter a lower power state (pin will go high due to pull-up that is internal to the SATA device, per DEVSLP specification). PCH will drive pin low to signal an exit from DEVSLP state. Design Constraint: As per the PDG, no external pull-up or pull-down termination required when used as DEVSLP. ***Note:*** This pin can be mapped to SATA Port 7.

Name	Type	Description
PCIE_12_UP_0_SATA0_TXP PCIE_12_UP_0_SATA0_TXN	O	**Serial ATA Differential Transmit Pair 0:** These outbound SATA Port 0 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 12 and PCIe Uplink Port 0. ***Note:*** Use FITC to allow selection of PCIe versus SATA based on strapping of SATA/PCIe Select GPIO polarity for SATA Port 0 with SATAXPCIE0.
PCIE_12_UP_0_SATA0_RXP PCIE_12_UP_0_SATA0_RXN	I	**Serial ATA Differential Receive Pair 0:** These inbound SATA Port 0 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 12 and PCIe Uplink Port 0. ***Note:*** Use FITC to allow selection of PCIe versus SATA based on strapping of SATA/PCIe Select GPIO polarity for SATA Port 0 with SATAXPCIE0.
PCIE_13_UP_1_SATA1_TXP PCIE_13_UP_1_SATA1_TXN	O	**Serial ATA Differential Transmit Pair 1:** These outbound SATA Port 1 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 13 and PCIe Uplink Port 1. ***Note:*** Use FITC to allow selection of PCIe versus SATA based on strapping of SATA/PCIe Select GPIO polarity for SATA Port 1 with SATAXPCIE1.
PCIE_13_UP_1_SATA1_RXP PCIE_13_UP_1_SATA1_RXN	I	**Serial ATA Differential Receive Pair 1 [First Instance]:** These inbound SATA Port 1 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 13 and PCIe Uplink Port 1. ***Note:*** Use FITC to allow selection of PCIe versus SATA based on strapping of SATA/PCIe Select GPIO polarity for SATA Port 1 with SATAXPCIE1.
PCIE_14_UP_2_SATA2_TXP PCIE_14_UP_2_SATA2_TXN	O	**Serial ATA Differential Transmit Pair 2:** These outbound SATA Port 2 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 14 and PCIe Uplink Port 2. ***Note:*** Use FITC to allow selection of PCIe versus SATA based on strapping of SATA/PCIe Select GPIO polarity for SATA Port 2 with SATAXPCIE2.
PCIE_14_UP_2_SATA2_RXP PCIE_14_UP_2_SATA2_RXN	I	**Serial ATA Differential Receive Pair 2:** These inbound SATA Port 2 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 14 and PCIe Uplink Port 2. ***Note:*** Use FITC to allow selection of PCIe versus SATA based on strapping of SATA/PCIe Select GPIO polarity for SATA Port 2 with SATAXPCIE2.
PCIE_15_UP_3_SATA3_TXP PCIE_15_UP_3_SATA3_TXN	O	**Serial ATA Differential Transmit Pair 3:** These outbound SATA Port 3 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 15 and PCIe Uplink Port 3. ***Note:*** Use FITC to allow selection of PCIe versus SATA based on strapping of SATA/PCIe Select GPIO polarity for SATA Port 3 with SATAXPCIE3.
PCIE_15_UP_3_SATA3_RXP PCIE_15_UP_3_SATA3_RXN	I	**Serial ATA Differential Receive Pair 3:** These inbound SATA Port 3 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 15 and PCIe Uplink Port 3. ***Note:*** Use FITC to allow selection of PCIe versus SATA based on strapping of SATA/PCIe Select GPIO polarity for SATA Port 3 with SATAXPCIE3.
PCIE_16_UP_4_SATA4_TXP PCIE_16_UP_4_SATA4_TXN	O	**Serial ATA Differential Transmit Pair 4:** These outbound SATA Port 4 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 16 and PCIe Uplink Port 4. ***Note:*** Use FITC to allow selection of PCIe versus SATA based on strapping of SATA/PCIe Select GPIO polarity for SATA Port 4 with SATAXPCIE4.
PCIE_16_UP_4_SATA4_RXP PCIE_16_UP_4_SATA4_RXN	I	**Serial ATA Differential Receive Pair 4:** These inbound SATA Port 4 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 16 and PCIe Uplink Port 4. ***Note:*** Use FITC to allow selection of PCIe versus SATA based on strapping of SATA/PCIe Select GPIO polarity for SATA Port 4 with SATAXPCIE4.

Name	Type	Description
PCIE_17_UP_5_SATA5_TXP PCIE_17_UP_5_SATA5_TXN	O	**Serial ATA Differential Transmit Pair 5:** These outbound SATA Port 5 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 17 and PCIe Uplink Port 5. ***Note:*** Use FITC to allow selection of PCIe versus SATA based on strapping of SATA/PCIe Select GPIO polarity for SATA Port 5 with SATAXPCIE5.
PCIE_17_UP_5_SATA5_RXP PCIE_17_UP_5_SATA5_RXN	I	**Serial ATA Differential Receive Pair 5:** These inbound SATA Port 5 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 17 and PCIe Uplink Port 5. ***Note:*** Use FITC to allow selection of PCIe versus SATA based on strapping of SATA/PCIe Select GPIO polarity for SATA Port 5 with SATAXPCIE5.
PCIE_18_UP_6_SATA6_TXP PCIE_18_UP_6_SATA6_TXN	O	**Serial ATA Differential Transmit Pair 6:** These outbound SATA Port 6 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 18 and PCIe Uplink Port 6. ***Note:*** Use FITC to allow selection of PCIe versus SATA based on strapping of SATA/PCIe Select GPIO polarity for SATA Port 6 with SATAXPCIE6.
PCIE_18_UP_6_SATA6_RXP PCIE_18_UP_6_SATA6_RXN	I	**Serial ATA Differential Receive Pair 6:** These inbound SATA Port 6 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 18 and PCIe Uplink Port 6. ***Note:*** Use FITC to allow selection of PCIe versus SATA based on strapping of SATA/PCIe Select GPIO polarity for SATA Port 6 with SATAXPCIE6.
PCIE_19_UP_7_SATA7_TXP PCIE_19_UP_7_SATA7_TXN	O	**Serial ATA Differential Transmit Pair 7:** These outbound SATA Port 7 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 19 and PCIe Uplink Port 7. ***Note:*** Use FITC to allow selection of PCIe versus SATA based on strapping of SATA/PCIe Select GPIO polarity for SATA Port 7 with SATAXPCIE7.
PCIE_19_UP_7_SATA7_RXP PCIE_19_UP_7_SATA7_RXN	I	**Serial ATA Differential Receive Pair 7:** These inbound SATA Port 7 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 19 and PCIe Uplink Port 7. ***Note:*** Use FITC to allow selection of PCIe versus SATA based on strapping of SATA/PCIe Select GPIO polarity for SATA Port 7 with / SATAXPCIE7.
GPP_E0_SATAXPCIE0_SATAGP0	I	**Serial ATA Port [0] General Purpose Inputs:** When configured as SATAGP_0, this is an input pin that is used as an interlock switch status indicator for SATA Port 0. Drive the pin to '0' to indicate that the switch is closed and to '1' to indicate that the switch is open.
GPP_E0_SATAXPCIE0_SATAGP0	I	S**ATA/PCIe selection**: When configured as SATAXPCIE0, this pin will configure the PCIE_12_UP_0_SATA0[TX/RX] pins to be either PCIe signals or SATAPort0 signals. The pin must be stable at the time PCH_PWROK goes high and remain in that state.
GPP_E1_SATAXPCIE1_SATAGP1	I	**Serial ATA Port [1] General Purpose Inputs:** When configured as SATAGP_1, this is an input pin that is used as an interlock switch status indicator for SATA Port 1. Drive the pin to '0' to indicate that the switch is closed and to '1' to indicate that the switch is open.
GPP_E1_SATAXPCIE1_SATAGP1	I	S**ATA/PCIe selection**: When configured as SATAXPCIE1, this pin will configure the PCIE_13_UP_1_SATA1[TX/RX] pins to be either PCIe signals or SATAPort1 signals. The pin must be stable at the time PCH_PWROK goes high and remain in that state.
GPP_E2_SATAXPCIE2_SATAGP2	I	**Serial ATA Port [2] General Purpose Inputs:** When configured as SATAGP_2, this is an input pin that is used as an interlock switch status indicator for SATA Port 2. Drive the pin to '0' to indicate that the switch is closed and to '1' to indicate that the switch is open.
GPP_E2_SATAXPCIE2_SATAGP2	I	S**ATA/PCIe selection**: When configured as SATAXPCIE2, this pin will configure the PCIE_14_UP_2_SATA2[TX/RX] pins to be either PCIe signals or SATAPort2 signals. The pin must be stable at the time PCH_PWROK goes high and remain in that state.
GPP_F0_SATAXCPIE3_SATAGP3	I	**Serial ATA Port [3] General Purpose Inputs:** When configured as SATAGP_3, this is an input pin that is used as an interlock switch status indicator for SATA Port 3. Drive the pin to '0' to indicate that the switch is closed and to '1' to indicate that the switch is open.

Name	Type	Description
GPP_F0_SATAXCPIE3_SATAGP3	I	S**ATA/PCIe selection**: When configured as SATAXPCIE3, this pin will configure the PCIE_15_UP_3_SATA3[TX/RX] pins to be either PCIe signals or SATAPort3 signals. The pin must be stable at the time PCH_PWROK goes high and remain in that state.
GPP_F1_SATAXPCIE4_SATAGP4	I	**Serial ATA Port [4] General Purpose Inputs:** When configured as SATAGP_4, this is an input pin that is used as an interlock switch status indicator for SATA Port 4. Drive the pin to '0' to indicate that the switch is closed and to '1' to indicate that the switch is open.
PP_F1_SATAXPCIE4_SATAGP4	I	S**ATA/PCIe selection**: When configured as SATAXPCIE4, this pin will configure the PCIE_16_UP_4_SATA4[TX/RX] pins to be either PCIe signals or SATAPort4 signals. The pin must be stable at the time PCH_PWROK goes high and remain in that state.
GPP_F2_SATAXPCIE5_SATAGP5	I	**Serial ATA Port [5] General Purpose Inputs:** When configured as SATAGP_5, this is an input pin that is used as an interlock switch status indicator for SATA Port 5. Drive the pin to '0' to indicate that the switch is closed and to '1' to indicate that the switch is open.
GPP_F2_SATAXPCIE5_SATAGP5	I	S**ATA/PCIe selection**: When configured as SATAXPCIE5, this pin will configure the PCIE_17_UP_5_SATA5[TX/RX] pins to be either PCIe signals or SATAPort5 signals. The pin must be stable at the time PCH_PWROK goes high and remain in that state.
GPP_F3_SATAXPCIE6_SATAGP6	I	**Serial ATA Port [6] General Purpose Inputs:** When configured as SATAGP_6, this is an input pin that is used as an interlock switch status indicator for SATA Port 6. Drive the pin to '0' to indicate that the switch is closed and to '1' to indicate that the switch is open.
GPP_F3_SATAXPCIE6_SATAGP6	I	S**ATA/PCIe selection**: When configured as SATAXPCIE6, this pin will configure the PCIE_18_UP_6_SATA6[TX/RX] pins to be either PCIe signals or SATAPort6 signals. The pin must be stable at the time PCH_PWROK goes high and remain in that state.
GPP_F4_SATAXPCIE7_SATAGP7	I	**Serial ATA Port [7] General Purpose Inputs:** When configured as SATAGP_7, this is an input pin that is used as an interlock switch status indicator for SATA Port 7. Drive the pin to '0' to indicate that the switch is closed and to '1' to indicate that the switch is open.
GPP_F3_SATAXPCIE6_SATAGP6	I	S**ATA/PCIe selection**: When configured as SATAXPCIE7, this pin will configure the PCIE_19_UP_7_SATA7[TX/RX] pins to be either PCIe signals or SATAPort7 signals. The pin must be stable at the time PCH_PWROK goes high and remain in that state.
GPP_E8_SATA_LED#	OD O	**Serial ATA LED:** This signal is an open-drain output pin driven during SATA command activity. It is to be connected to external circuitry that can provide the current to drive a platform LED. When active, the LED is on. When tri-stated, the LED is off. ***Note:*** An external pull-up resistor to VCC3_3 is required.
GPP_F10_SATA_SCLOCK	OD	**SGPIO Reference Clock:** The SATA controller uses rising edges of this clock to transmit serial data, and the target uses the falling edge of this clock to latch data. The SClock frequency supported is 32 kHz. If SGPIO interface is not used, this signal can be used as GPP_F10.
GPP_F11_SATA_SLOAD	OD	**SGPIO Load:** The controller drives a '1' at the rising edge of SCLOCK to indicate either the start or end of a bit stream. A 4-bit vendor specific pattern will be transmitted right after the signal assertion. If SGPIO interface is not used, this signal can be used as GPP_F11.
GPP_F13_SATA_SDATAOUT0	OD	**SGPIO Dataout0:** Driven by the controller to indicate the drive status in the following sequence: drive 0, 1,2, 3, 4, 5, 6, 7, 0, 1, 2... If SGPIO interface is not used, the signals can be used as GPP_F13.
GPP_F12_SATA_SDATAOUT1	OD	**SGPIO Dataout1:** Driven by the controller to indicate the drive status in the following sequence: drive 0, 1,2, 3, 4, 5, 6, 7, 0, 1, 2... If SGPIO interface is not used, the signals can be used as GPP_F12.

Name	Type	Description
GPP_G20_SSATA_DEVSLP0	OD	**SSATA Port [0] Device Sleep**: This is an open-drain pin on the PCH side. PCH will tri-state this pin to signal to the SATA device that it may enter a lower power state (pin will go high due to pull-up that is internal to the SATA device, per DEVSLP specification). PCH will drive pin low to signal an exit from DEVSLP state. Design Constraint: As per PDG, no external pull-up or pull-down termination required when used as DEVSLP. This pin can be mapped to SSATA Port 0.
GPP_G21_SSATA_DEVSLP1	OD	**SSATA Port [1] Device Sleep**: This is an open-drain pin on the PCH side. PCH will tri-state this pin to signal to the SATA device that it may enter a lower power state (pin will go high due to pull-up that is internal to the SATA device, per DEVSLP specification). PCH will drive pin low to signal an exit from DEVSLP state. Design Constraint: As per PDG, no external pull-up or pull-down termination required when used as DEVSLP. This pin can be mapped to SSATA Port 1.
GPP_G22_SSATA_DEVSLP2	OD	**SSATA Port [2] Device Sleep**: This is an open-drain pin on the PCH side. PCH will tri-state this pin to signal to the SATA device that it may enter a lower power state (pin will go high due to pull-up that is internal to the SATA device, per DEVSLP specification). PCH will drive pin low to signal an exit from DEVSLP state. Design Constraint: As per PDG, no external pull-up or pull-down termination required when used as DEVSLP. This pin can be mapped to SSATA Port 2.
GPP_D9_SSATA_DEVSLP3	OD	**SSATA Port [3] Device Sleep**: This is an open-drain pin on the PCH side. PCH will tri-state this pin to signal to the SATA device that it may enter a lower power state (pin will go high due to pull-up that is internal to the SATA device, per DEVSLP specification). PCH will drive pin low to signal an exit from DEVSLP state. Design Constraint: As per PDG, no external pull-up or pull-down termination required when used as DEVSLP. This pin can be mapped to SSATA Port 3.
GPP_D10_SSATA_DEVSLP4	OD	**SSATA Port [4] Device Sleep**: This is an open-drain pin on the PCH side. PCH will tri-state this pin to signal to the SATA device that it may enter a lower power state (pin will go high due to pull-up that is internal to the SATA device, per DEVSLP specification). PCH will drive pin low to signal an exit from DEVSLP state. Design Constraint: As per PDG, no external pull-up or pull-down termination required when used as DEVSLP. This pin can be mapped to SSATA Port 4.
GPP_D11_SSATA_DEVSLP5	OD	**SSATA Port [5] Device Sleep**: This is an open-drain pin on the PCH side. PCH will tri-state this pin to signal to the SATA device that it may enter a lower power state (pin will go high due to pull-up that is internal to the SATA device, per DEVSLP specification). PCH will drive pin low to signal an exit from DEVSLP state. Design Constraint: As per PDG, no external pull-up or pull-down termination required when used as DEVSLP. This pin can be mapped to SSATA Port 5.
PCIE_6_SSATA0_TXP PCIE_6_SSATA0_TXN	O	**SSATA Differential Transmit Pair 0:** These outbound SATA Port 0 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 6. ***Note:*** Use FITC to allow selection of PCIe versus SSATA based on strapping of SATA/PCIe Select GPIO polarity for SSATA Port 0 with SSATAXPCIE0.
PCIE_6_SSATA0_RXP PCIE_6_SSATA0_RXN	I	**SSATA Differential Receive Pair 0:** These outbound SATA Port 0 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 6. ***Note:*** Use FITC to allow selection of PCIe versus SSATA based on strapping of SATA/PCIe Select GPIO polarity for SSATA Port 0 with SSATAXPCIE0.

Name	Type	Description
PCIE_7_SSATA1_TXP PCIE_7_SSATA1_TXN	O	**SSATA Differential Transmit Pair 1:** These outbound SATA Port 0 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 7. ***Note:*** Use FITC to allow selection of PCIe versus SSATA based on strapping of SATA/PCIe Select GPIO polarity for SSATA Port 1 with SSATAXPCIE1.
PCIE_7_SSATA1_RXP PCIE_7_SSATA1_RXN	I	**SSATA Differential Receive Pair 1:** These outbound SATA Port 0 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 7. ***Note:*** Use FITC to allow selection of PCIe versus SSATA based on strapping of SATA/PCIe Select GPIO polarity for SSATA Port 1 with SSATAXPCIE1.
PCIE_8_SSATA2_GBE_TXP PCIE_8_SSATA2_GBE_TXN	O	**SSATA Differential Transmit Pair 2:** These outbound SATA Port 0 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 8 and the 1GbE MAC ***Note:*** Use FITC to allow selection of PCIe versus SSATA based on strapping of SATA/PCIe Select GPIO polarity for SSATA Port 2 with SSATAXPCIE2.
PCIE_8_SSATA2_GBE_RXP PCIE_8_SSATA2_GBE_RXN	I	**SSATA Differential Receive Pair 2:** These outbound SATA Port 0 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 8 and the 1GbE MAC. ***Note:*** Use FITC to allow selection of PCIe versus SSATA based on strapping of SATA/PCIe Select GPIO polarity for SSATA Port 2 with SSATAXPCIE2
PCIE_9_SSATA3_TXP PCIE_9_SSATA3_TXN	O	**SSATA Differential Transmit Pair 3:** These outbound SATA Port 0 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 9. ***Note:*** Use FITC to allow selection of PCIe versus SSATA based on strapping of SATA/PCIe Select GPIO polarity for SSATA Port 3 with SSATAXPCIE3.
PCIE_9_SSATA3_RXP PCIE_9_SSATA3_RXN	I	**SSATA Differential Receive Pair 3:** These outbound SATA Port 0 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 9. ***Note:*** Use FITC to allow selection of PCIe versus SSATA based on strapping of SATA/PCIe Select GPIO polarity for SSATA Port 3 with SSATAXPCIE3.
PCIE_10_SSATA4_TXP PCIE_10_SSATA4_TXN	O	**SSATA Differential Transmit Pair 4:**These outbound SATA Port 0 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 10 ***Note:*** Use FITC to allow selection of PCIe versus SSATA based on strapping of SATA/PCIe Select GPIO polarity for SSATA Port 4 with SSATAXPCIE4.
PCIE_10_SSATA4_RXP PCIE_10_SSATA4_RXN	I	**SSATA Differential Receive Pair 4:** These outbound SATA Port 0 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 10. ***Note:*** Use FITC to allow selection of PCIe versus SSATA based on strapping of SATA/PCIe Select GPIO polarity for SSATA Port 4 with SSATAXPCIE4.
PCIE_11_SSATA5_TXP PCIE_11_SSATA5_TXN	O	**SSATA Differential Transmit Pair 5:**These outbound SATA Port 0 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 11 ***Note:*** Use FITC to allow selection of PCIe versus SSATA based on strapping of SATA/PCIe Select GPIO polarity for SSATA Port 5 with SSATAXPCIE5.
PCIE_11_SSATA5_RXP PCIE_11_SSATA5_RXN	I	**SSATA Differential Receive Pair 5:** These outbound SATA Port 0 high-speed differential signals support 1.5 Gb/s, 3 Gb/s and 6 Gb/s. The signals are multiplexed with PCIe Root Port 11. ***Note:*** Use FITC to allow selection of PCIe versus SSATA based on strapping of SATA/PCIe Select GPIO polarity for SSATA Port 5 with SSATAXPCIE5.
GPP_G23_SSATAXPCIE0_SSATAGP0/	I	**SSATA Port [0] General Purpose Inputs:** When configured as SSATAGP_0, this is an input pin that is used as an interlock switch status indicator for SSATA Port 0. Drive the pin to '0' to indicate that the switch is closed and to '1' to indicate that the switch is open.

Name	Type	Description
GPP_G23_SSATAXPCIE0_SSATAGP0	I	**SSATA/PCIe selection**: When configured as SSATAXPCIE0, this pin will configure the PCIE_6_SSATA0[TX/RX] pins to be either PCIe signals or SSATAPort0 signals. The pin must be stable at the time PCH_PWROK goes high and remain in that state.
GPP_H19_SSATAXPCIE1_SSATAGP1/	I	**SSATA Port [1] General Purpose Inputs:** When configured as SSATAGP_1, this is an input pin that is used as an interlock switch status indicator for SSATA Port 1. Drive the pin to '0' to indicate that the switch is closed and to '1' to indicate that the switch is open.
GPP_H19_SSATAXPCIE1_SSATAGP1/	I	**SSATA/PCIe selection**: When configured as SSATAXPCIE1, this pin will configure the PCIE_7_SSATA1[TX/RX] pins to be either PCIe signals or SSATAPort1 signals.
GPP_H20_SSATAXPCIE2_SSATAGP2/	I	**SSATA Port [2] General Purpose Inputs:** When configured as SSATAGP_2, this is an input pin that is used as an interlock switch status indicator for SSATA Port 2. Drive the pin to '0' to indicate that the switch is closed and to '1' to indicate that the switch is open.
GPP_H20_SSATAXPCIE2_SSATAGP2	I	**SSATA/PCIe selection**: When configured as SSATAXPCIE2, this pin will configure the PCIE_8_SSATA2[TX/RX] pins to be either PCIe signals or SSATAPort2 signals. The pin must be stable at the time PCH_PWROK goes high and remain in that state.
GPP_H21_SSATAXPCIE3_SSATAGP3/	I	**SSATA Port [3] General Purpose Inputs:** When configured as SSATAGP_3, this is an input pin that is used as an interlock switch status indicator for SSATA Port 3. Drive the pin to '0' to indicate that the switch is closed and to '1' to indicate that the switch is open.
GPP_H21_SSATAXPCIE3_SSATAGP3/	I	**SSATA/PCIe selection**: When configured as SSATAXPCIE3, this pin will configure the PCIE_9_SSATA3[TX/RX] pins to be either PCIe signals or SSATAPort3 signals. The pin must be stable at the time PCH_PWROK goes high and remain in that state.
GPP_H22_SSATAXPCIE4_SSATAGP4/	I	**SSATA Port [4] General Purpose Inputs:** When configured as SSATAGP_4, this is an input pin that is used as an interlock switch status indicator for SSATA Port 4. Drive the pin to '0' to indicate that the switch is closed and to '1' to indicate that the switch is open.
GPP_H22_SSATAXPCIE4_SSATAGP4/	I	**SSATA/PCIe selection**: When configured as SSATAXPCIE4, this pin will configure the PCIE_10_SSATA4[TX/RX] pins to be either PCIe signals or SSATAPort4 signals. The pin must be stable at the time PCH_PWROK goes high and remain in that state.
GPP_H23_SSATAXPCIE4_SSATAGP4/	I	**SSATA Port [5] General Purpose Inputs:** When configured as SSATAGP_5, this is an input pin that is used as an interlock switch status indicator for SSATA Port 5. Drive the pin to '0' to indicate that the switch is closed and to '1' to indicate that the switch is open.
GPP_H23_SSATAXPCIE4_SSATAGP4	I	**SSATA/PCIe selection**: When configured as SSATAXPCIE5, this pin will configure the PCIE_11_SSATA5[TX/RX] pins to be either PCIe signals or SSATAPort5 signals. The pin must be stable at the time PCH_PWROK goes high and remain in that state.
GPP_F22_SSATA_SCLOCK/	OD	**SGPIO Reference Clock:** The SSATA controller uses rising edges of this clock to transmit serial data, and the target uses the falling edge of this clock to latch data. The SClock frequency supported is 32 kHz. If SGPIO interface is not used, this signal can be used as GPP_F22.
GPP_F23_SSATA_SLOAD/	OD	**SGPIO Load:** The controller drives a '1' at the rising edge of SCLOCK to indicate either the start or end of a bit stream. A 4-bit vendor specific pattern will be transmitted right after the signal assertion. If SGPIO interface is not used, this signal can be used as GPP_F22.
GPP_D15_SSATA_SDATAOUT0	OD	**SGPIO Dataout0:** Driven by the SSATA controller to indicate the drive status in the following sequence: drive 0, 1,2, 3, 4, 5, 6, 7, 0, 1, 2... If SGPIO interface is not used, the signals can be used as GPP_D15.
GPP_D12_SSATA_SDATAOUT1	OD	**SGPIO Dataout1:** Driven by the SSATA controller to indicate the drive status in the following sequence: drive 0, 1,2, 3, 4, 5, 6, 7, 0, 1, 2... If SGPIO interface is not used, the signals can be used as GPP_D12.

24.5 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Nominal Value	Notes
SATAXPCIE[2:0]	Pull-up	20 KΩ	1, 2
SATAXPCIE[7:3]	Pull-up	20 KΩ	1, 2

Note:
1. SATAGP[2:0]/SATAXPCIE[2:0]/GPP_E[2:0] and SATAGP[7:3]/SATAXPCIE[7:3]/GPP_F[4:0] has two native functions – the first native function (SATAXPCIEx) is selected if the Flex I/O soft strap SATA_PCIE_Select_for_Port_x = 11b. Setting SATA_PCIE_Select_for_Port_x = 11b also enables an internal pull-up resistor in this pin to allow Flexible I/O selection of SATA Port x or PCIe Port x to be assigned based on the type of card installed and based on the SATAXPCIEx mux selector with the polarity for SATA or PCIe (When SPSGPx = 0, PCIe will be selected if the sampled value is "0" and SATA will be selected if the sampled value is "1"; When SPSGPx = 1, SATA will be selected if the sampled value is "0" and PCIe* will be selected if the sampled value is "1"). SATA/PCIe Select GPIO polarity for SATA Port x (SPSGPx) and Soft straps are handled through FITC and described in the Intel® C620 Series Chipset PCH SPI Flash Programming Guide (RDC document number 559021).
2. Simulation data shows that these resistor values can range from 14 kΩ – 26 kΩ.
3. Integrated pull-ups and pull-downs will not be valid until all the voltages have reached a valid level.

24.6 I/O Signal Planes and States

Signal Name	Power Plane	During Reset	Immediately After Reset	S3/S4/S5	Deep Sx
SATA[7:0]_TXP/N, **SATA[7:0]_RXP/N**[3]	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	Off
SSATA[5:0]_TXP/N, **SSATA[5:0]_RXP/N**[3]	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	Off
SATALED#/GPP_E8[1]	Primary	Undriven	Undriven	Undriven	Off
GPP_F14_SSATA_LED#	Primary	Undriven	Undriven	Undriven	Off
DEVSLP[2:0]/GPP_E[6:4][1]	Primary	Undriven	Undriven	Undriven	Off
DEVSLP[7:3]/GPP_F[9:5][1, 3]	Primary	Undriven	Undriven	Undriven	Off
SSATA_DEVSLP[2:0]	Primary	Undriven	Undriven	Undriven	Off
SSATA_DEVSLP[5:3]	Primary	Undriven	Undriven	Undriven	Off
SATAGP[2:0]/GPP_E[2:0][2, 3]	Primary	Undriven	Undriven	Undriven	Off
SATAGP[7:3]/GPP_F[4:0][2]	Primary	Undriven	Undriven	Undriven	Off
SSATAGP[5:0]	Primary	Undriven	Undriven	Undriven	Off
SATAXPCIE[7:0] [3]	Primary	Internal Pull-up	Internal Pull-u	Undriven	Off
SSATAXPCIE[5:0]	Primary	Internal Pull-up	Internal Pull-u	Undriven	Off
SATA_SCLOCK/GPP_F10[1]	Primary	Undriven	Undriven	Undriven	Off
SATA_SLOAD/GPP_F11[1]	Primary	Undriven	Undriven	Undriven	Off
SATA_SDATAOUT[1:0]	Primary	Undriven	Undriven	Undriven	Off
SSATA_CLOCK[1]	Primary	Undriven	Undriven	Undriven	Off
SSATA_SLOAD	Primary	Undriven	Undriven	Undriven	Off
SSATA_SDATA[1:0]	Primary	Undriven	Undriven	Undriven	Off

Note:
1. Pin defaults to GPIO mode. The pin state during and immediately after reset follows default GPIO mode pin state. The pin state for S0 to Deep Sx reflects assumption that GPIO Use Select register was programmed to native mode functionality. If GPIO Use Select register is programmed to GPIO mode, refer to Multiplexed GPIO (Defaults to GPIO Mode) section for the respective pin states in S0 to Deep Sx.
2. Pin defaults to Native mode as SATAXPCIEx depends on soft-strap.

24.7 Functional Description

This chapter describes the SATA functions of both the SATA and SSATA controllers. For the rest of the chapter, when "SATA" is used, it'll be referring to both SATA and SSATA, unless explicitly calling out them individually.

24.7.1 SATA Host Controller (D23:F0) SSATA Host Controller (D17:F5)

The SATA function supports AHCI or RAID mode.

The SATA controllers do not support legacy IDE mode or combination mode.

The SATA controller features14 ports split across the SATA (8) and SSATA (6) controllers that can be independently enabled or disabled (they cannot be tri-stated or driven low). Each interface is supported by an independent DMA controller.

The PCH SATA controller interacts with an attached mass storage device through a register interface that is compatible with an SATA AHCI/RAID host adapter. The host software follows existing standards and conventions when accessing the register interface and follows standard command protocol conventions.

24.7.2 SATA 6 Gb/s Support

The PCH SATA controller is SATA 6 Gb/s capable and supports 6 Gb/s transfers with all capable SATA devices. The PCH SATA controller also supports SATA 3 Gb/s and 1.5 Gb/s transfer capabilities.

24.7.3 SATA Feature Support

The PCH SATA controller is capable of supporting all AHCI 1.3 and AHCI 1.3.1 expanded capabilities for DEVSLP.

Also, refer to the Intel web site on Advanced Host Controller Interface Specification for current specification status: http://www.intel.com/content/www/us/en/io/serial-ata/ahci.html.

For capability details, refer to D23:F0:Offset 00h CAP, and AHCI BAR PxCMD Offset 18h for the SATA controller, and D17:F5:00fset 00h CAP and AHCI BAR PxCMD off18h for the SSATA controller.

The PCH SATA controller does not support:

- Port Multiplier
- FIS Based Switching
- Command Based Switching
- IDE mode or combination mode
- Cold Presence Detect
- Function Level Reset (FLR)

24.7.4 Hot-Plug Operation

The PCH SATA controller supports Hot-Plug Surprise removal and Insertion Notification. An internal SATA port with a Mechanical Presence Switch can support PARTIAL and SLUMBER with Hot-Plug Enabled. Software can take advantage of power savings in the low power states while enabling Hot-Plug operation. Refer to Chapter 7 of the *AHCI Specification* for details.

24.7.5 Power Management Operation

Power management of the PCH SATA controller and ports will cover operations of the host controller and the SATA link.

24.7.5.1 Power State Mappings

The D0 PCI Power Management (PM) state for device is supported by the PCH SATA controller.

SATA devices may also have multiple power states. SATA adopted three main power states from parallel ATA. The three device states are supported through ACPI. They are:

- **D0** – Device is working and instantly available.
- **D1** – Device enters when it receives a STANDBY IMMEDIATE command. Exit latency from this state is in seconds
- **D3** – From the SATA device's perspective, no different than a D1 state, in that it is entered using the STANDBY IMMEDIATE command. However, an ACPI method is also called which will reset the device and then cut its power.

Each of these device states are subsets of the host controller's D0 state.

Finally, the SATA specification defines three PHY layer power states, which have no equivalent mappings to parallel ATA. They are:

- **PHY READY** – PHY logic and PLL are both on and in active state
- **Partial** – PHY logic is powered up, and in a reduced power state. The link PM exit latency to active state maximum is 10 ns.
- **Slumber** – PHY logic is powered up, and in a reduced power state. The link PM exit latency to active state maximum is 10 ms.
- **Devslp** – PHY logic is powered down. The link PM exit latency from this state to active state maximum is 20 ms, unless otherwise specified by DETO in Identify Device Data Log page 08h (see 13.7.9.4 of the *SATA Rev3.2 Gold Specification*).

Since these states have much lower exit latency than the ACPI D1 and D3 states, the SATA controller specification defines these states as sub-states of the device D0 state.

24.7.5.2 Power State Transitions

24.7.5.2.1 Partial and Slumber State Entry/Exit

The partial and slumber states save interface power when the interface is idle. It would be most analogous to CLKRUN# (in power savings, not in mechanism), where the interface can have power saved while no commands are pending. The SATA controller defines PHY layer power management (as performed using primitives) as a driver

operation from the host side, and a device proprietary mechanism on the device side. The SATA controller accepts device transition types, but does not issue any transitions as a host. All received requests from a SATA device will be ACKed.

When an operation is performed to the SATA controller such that it needs to use the SATA cable, the controller must check whether the link is in the Partial or Slumber states, and if so, must issue a COMWAKE to bring the link back online. Similarly, the SATA device must perform the same COMWAKE action.

24.7.5.2.2 Devslp State Entry/Exit

Device Sleep (DEVSLP) is a host-controlled SATA interface power state. To support a hardware autonomous approach that is software agnostic Intel is recommending that BIOS configure the AHCI controller and the device to enable Device Sleep. This allows the AHCI controller and associated device to automatically enter and exit Device Sleep without the involvement of OS software.

To enter Device Sleep the link must first be in Slumber. By enabling HIPM (with Slumber) or DIPM on a Slumber capable device, the device/host link may enter the DevSleep Interface Power state.

The device must be DevSleep capable. Device Sleep is only entered when the link is in slumber, therefore when exiting the Device Sleep state, the device must resume with the COMWAKE out-of-band signal (and not the COMINIT out-of-band signal). Assuming Device Sleep was asserted when the link was in slumber, the device is expected to exit DEVSLP to the DR_Slumber state. Devices that do not support this feature will not be able to take advantage of the hardware automated entry to Device Sleep that is part of the AHCI 1.3.1 specification and supported by Intel platforms.

24.7.5.2.3 Device D1 and D3 States

These states are entered after some period of time when software has determined that no commands will be sent to this device for some time. The mechanism for putting a device in these states does not involve any work on the host controller, other then sending commands over the interface to the device. The command most likely to be used in ATA/ATAPI is the “STANDBY IMMEDIATE” command.

24.7.5.2.4 Host Controller $D3_{HOT}$ State

After the interface and device have been put into a low power state, the SATA host controller may be put into a low power state. This is performed using the PCI power management registers in configuration space. There are two very important aspects to Note when using PCI power management.

1. When the power state is D3, only accesses to configuration space are allowed. Any attempt to access the memory or I/O spaces will result in master abort.
2. When the power state is D3, no interrupts may be generated, even if they are enabled. If an interrupt status bit is pending when the controller transitions to D0, an interrupt may be generated.

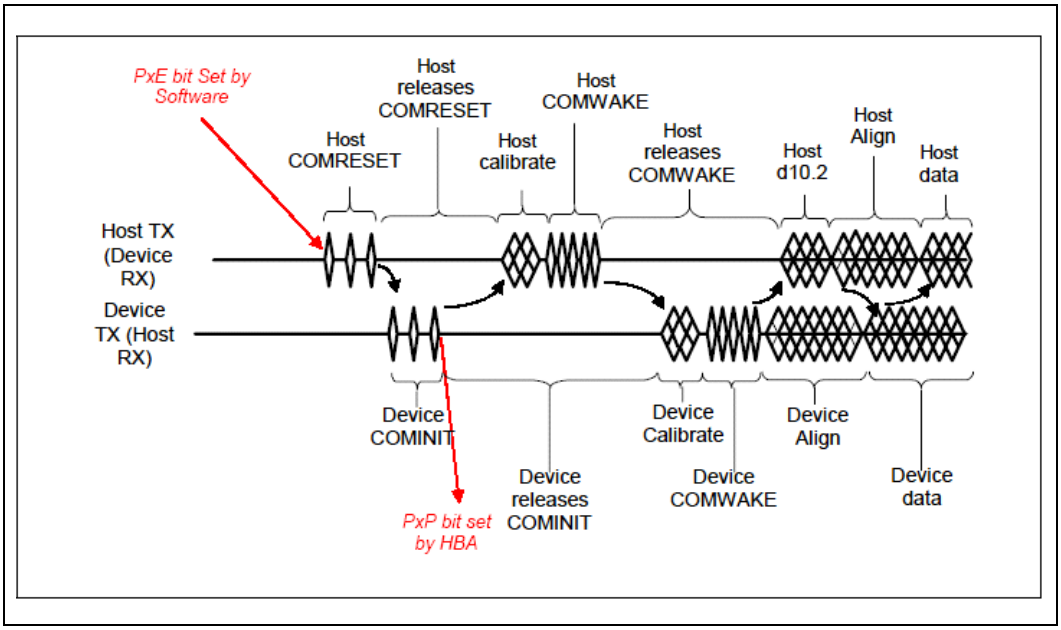
When the controller is put into D3, it is assumed that software has properly shut down the device and disabled the ports. Therefore, there is no need to sustain any values on the port wires. The interface will be treated as if no device is present on the cable, and power will be minimized.

When returning from a D3 state, an internal reset will not be performed.

24.7.6 SATA Device Presence

The flow used to indicate SATA device presence is shown in Figure 24-1. The 'PxE' bit refers to PCS.P[3:0]E bits, depending on the port being checked and the 'PxP' bits refer to the PCS.P[3:0]P bits, depending on the port being checked. If the PCS/PxP bit is set a device is present, if the bit is cleared a device is not present. If a port is disabled, software can check to see if a new device is connected by periodically re-enabling the port and observing if a device is present, if a device is not present it can disable the port and check again later. If a port remains enabled, software can periodically poll PCS.PxP to see if a new device is connected.

Figure 24-1. Flow for Port Enable/Device Present Bits



24.7.7 SATA LED

The SATALED# output is driven whenever the BSY bit is set in any SATA port. The SATALED# is an active-low open-drain output. When SATALED# is low, the LED should be active. When SATALED# is high, the LED should be inactive.

24.7.8 Advanced Host Controller Interface (AHCI) Operation

The PCH SATA controller provides hardware support for Advanced Host Controller Interface (AHCI), a standardized programming interface for SATA host controllers developed through a joint industry effort. Platforms supporting AHCI may take advantage of performance features such as port independent DMA Engines—each device is treated as a master—and hardware-assisted native command queuing.

AHCI defines transactions between the SATA controller and software and enables advanced performance and usability with SATA. Platforms supporting AHCI may take advantage of performance features such as no master/slave designation for SATA devices—each device is treated as a master—and hardware assisted native command queuing. AHCI also provides usability enhancements such as hot-plug and advanced power management. AHCI requires appropriate software support (such as, an AHCI

driver) and for some features, hardware support in the SATA device or additional platform hardware. Visit the Intel web site for current information on the AHCI specification.

The PCH SATA controller supports all of the mandatory features of the *Serial ATA Advanced Host Controller Interface Specification*, Revision 1.3 and many optional features, such as hardware assisted native command queuing, aggressive power management, LED indicator support, and hot-plug through the use of interlock switch support (additional platform hardware and software may be required depending upon the implementation).

Note: For reliable device removal notification while in AHCI operation without the use of interlock switches (surprise removal), interface power management should be disabled for the associated port. See Section 7.3.1 of the *AHCI Specification* for more information.

24.7.9 External SATA

The PCH SATA controller supports external SATA. External SATA utilizes the SATA interface outside of the system box. The usage model for this feature must comply with the *Serial ATA II (SATA Gen2, 3 Gb/s) Cables and Connectors Volume 2 Gold Specification* at: www.sata-io.org. Intel validates one configuration: the back-panel solution involves running a trace to the I/O back panel and connecting a device using an external SATA connector on the board.

24.7.10 Enclosure Management (SGPIO Signals)

Enclosure management is a mechanism by which the storage driver can monitor and control auxiliary service in a drive enclosure. This feature is only valid in AHCI/RAID mode.

The SGPIO signals are used in the enclosure management protocol (refer to the *SFF-8485 Specification*) and supports multiple-activity LEDs to shows the per drive status information.

The enclosure management HW is assuming that these signals will only be used if a device is either connected to a port or could be connected to a port without the system being reset, meaning hot plug is enabled. If hot plug is disabled and there are no devices attached to the SATA/SSATA ports, then the SGPIO pins will not function correctly if any commands are sent to the controller to be drive on the SGPIO pins. As there are two SGPIO controllers, one for SATA and one for SSATA, the rules apply to both. If you have a device attached to SATA, but SSATA has no devices or hot plug enabled, then the SGPIO signals will not function on the SSATA SGPIO pins, but will work on the SATA SGPIO pins.

Note: These signals are not related to SATALED#.

The SGPIO group interfaces with an external controller chip that fetches and serializes the data for driving across the SGPIO bus. The output signals then control the LEDs within the enclosure. The PCH SATA controller only supports LED messages transmission and has three SGPIO protocol signals implemented, that is SCLOCK, SDATAOUT and SLOAD.

Note: Intel does not validate all possible usage cases of this feature. Customers should validate their specific design implementation on their own platforms.

24.7.10.1 Mechanism

The enclosure management for SATA Controller involves sending messages that control LEDs in the enclosure. The messages for this function are stored after the normal registers in the AHCI BAR, at Offset 580h bytes for the PCH from the beginning of the AHCI BAR as specified by the EM_LOC global register.

Software creates messages for transmission in the enclosure management message buffer. The data in the message buffer should not be changed if CTL.TM bit is set by software to transmit an update message. Software should only update the message buffer when CTL.TM bit is cleared by hardware otherwise the message transmitted will be indeterminate. Software then writes a register to cause hardware to transmit the message or take appropriate action based on the message content. The software should only create message types supported by the controller, which is LED messages for the PCH. If the software creates other non LED message types (such as, SAF-TE, SES-2), the SGPIO interface may hang and the result is indeterminate.

During reset, all SGPIO pins will be in the tri-state state. The interface will continue staying in the tri-state state after reset until the first transmission occurs, when software programs the message buffer and sets the transmit bit CTL.TM. The SATA host controller will initiate the transmission by driving SCLOCK and at the same time driving the SLOAD to "0" prior to the actual bit stream transmission. The Host will drive SLOAD low for at least 5 SCLOCK then only start the bit stream by driving the SLOAD to high. SLOAD will be driven high for 1 SCLOCK, followed by vendor-specific pattern that is default to "0000" if software is yet to program the value. A total of 24-bit streams from 8 ports (Port 0, Port 1, Port 2, Port 3, Port 4, Port 5, Port 6, Port 7) of 3-bit per port LED message will be transmitted on SDATAOUT0 pin after the SLOAD is driven high for 1 SCLOCK. For 8 SATA port configuration, only 4 ports (port 4, port 5, port 6 and port 7) of 12 bit total LED message follow by 12 bits of tri-state value will be transmitted out on SDATAOUT1 pin. For 6 SATA port configuration, only 2 ports (port 4 and port 5) of 6 bit total LED message follow by 18 bits of tri-state value will be transmitted out on SDATAOUT1 pin. For 4 SATA port configuration, SDATAOUT1 pin is not required hence can be tri-state always.

All the default LED message values will be high prior to software setting them, except the Activity LED message that is configured to be hardware driven that will be generated based on the activity from the respective port. All the LED message values will be driven to '1' for the port that is unimplemented as indicated in the Port Implemented register regardless of the software programmed value through the message buffer.

There are 2 different ways of resetting the PCH's SGPIO interface, asynchronous reset and synchronous reset. Asynchronous reset is caused by platform reset to cause the SGPIO interface to be tri-state asynchronously. Synchronous reset is caused by setting the CTL.RESET bit, or HBA reset, where Host Controller will complete the existing full bit stream transmission then only tri-state all the SGPIO pins. After the reset, both synchronous reset and asynchronous reset, the SGPIO pins will stay tri-stated.

Note: The PCH Host Controller does not ensure that it will cause the target SGPIO device or controller to be reset. Software is responsible to keep the PCH SGPIO interface in tri-state for 2 second to cause a reset on the target of the SGPIO interface.

24.7.10.2 Message Format

Messages shall be constructed with a one DWord header that describes the message to be sent followed by the actual message contents. The first DWord shall be constructed as shown in Enclosure Management Message Format (EM_MF) register.

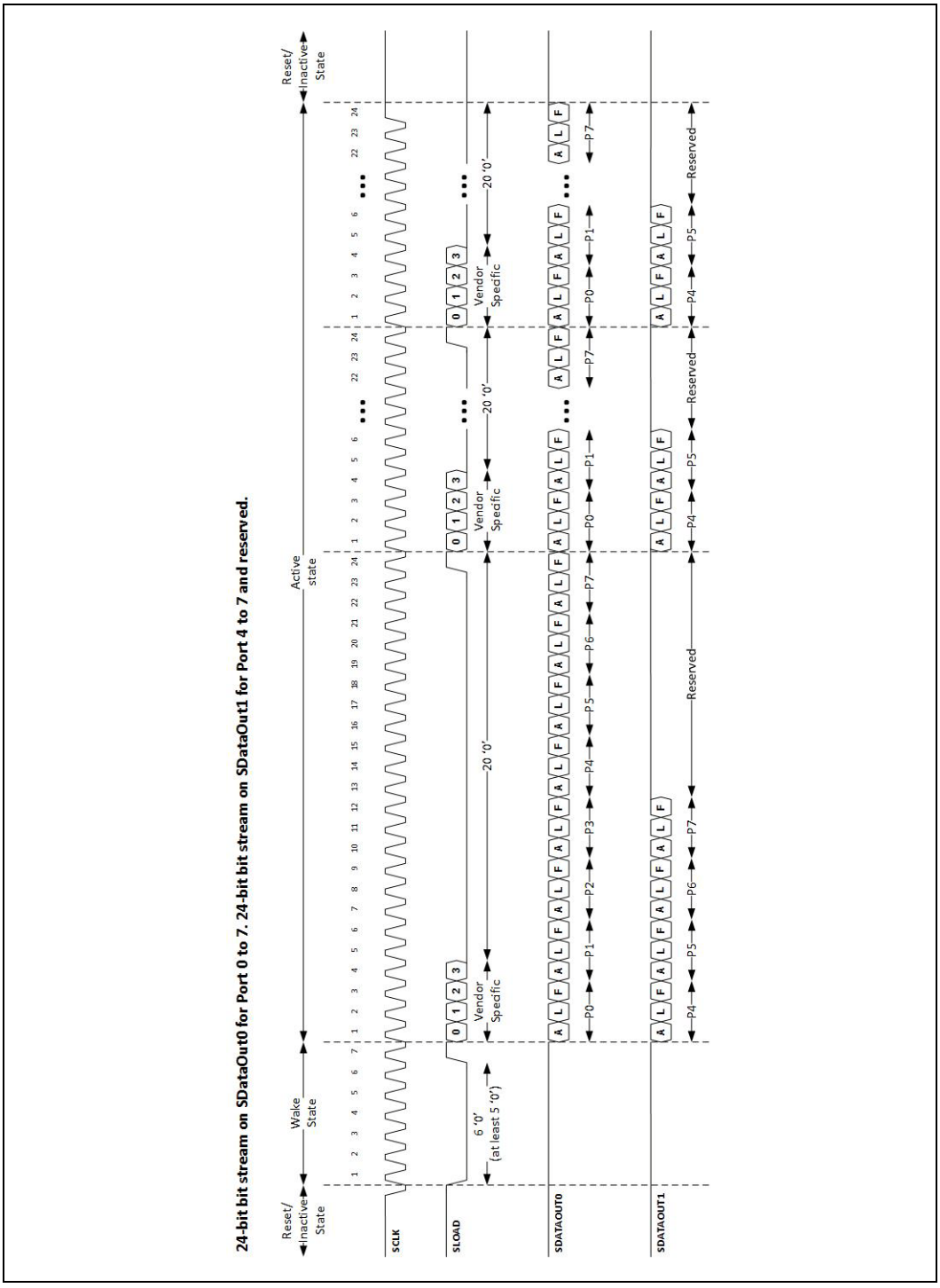
The SAF-TE, SES-2, and SGPIO message formats are defined in the corresponding specifications, respectively. The LED message type is defined in the Enclosure Management LED (EM_LED) register. It is the responsibility of software to ensure the content of the message format is correct. If the message type is not programmed as 'LED' for this controller, the controller shall not take any action to update its LEDs. For LED message type, the message size always consists of 4 bytes.

24.7.10.3 LED Message Type

The LED message type specifies the status of up to three LEDs. Typically, the usage for these LEDs is activity, fault, and locate. Not all implementations necessarily contain all LEDs (for example, some implementations may not have a locate LED). The message identifies the HBA port number and the Port Multiplier port number that the slot status applies to. If a Port Multiplier is not in use with a particular device, the Port Multiplier port number shall be ‘0’. The format of the LED message type is defined in the Enclosure Management LED (EM_LED) register The LEDs shall retain their values until there is a following update for that particular slot.

24.7.10.4 SGPIO Waveform

Figure 24-2. Serial Data Transmitted Over the SGPIO Interface



24.8 SATA Port Registers

24.8.1 SATA Configuration Registers Summary

Table 24-1. Summary of SATA Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	Identifiers (ID)—Offset 0h	1C028086h
4h	5h	Command (CMD)—Offset 4h	0h
6h	7h	Device Status (STS)—Offset 6h	210h
8h	8h	Revision ID (RID)—Offset 8h	0h
9h	9h	Programming Interface (PI)—Offset 9h	1h
Ah	Bh	Class Code (CC)—Offset Ah	106h
Ch	Ch	Cache Line Size (CLS)—Offset Ch	0h
Dh	Dh	Master Latency Timer (MLT)—Offset Dh	0h
Eh	Eh	Header Type (HTYPE)—Offset Eh	0h
10h	13h	MSI-X Table Base Address (MXTBA)—Offset 10h	0h
14h	17h	MXP Base Address (MXPBA)—Offset 14h	0h
20h	23h	AHCI Index Data Pair Base Address (AIDPBA)—Offset 20h	1h
24h	27h	AHCI Base Address (ABAR)—Offset 24h	0h
2Ch	2Fh	Sub System Identifiers (SS)—Offset 2Ch	0h
34h	34h	Capabilities Pointer (CAP)—Offset 34h	80h
3Ch	3Dh	Interrupt Information (INTR)—Offset 3Ch	100h
70h	71h	PCI Power Management Capability ID (PID)—Offset 70h	A801h
72h	73h	PCI Power Management Capabilities (PC)—Offset 72h	4003h
74h	75h	PCI Power Management Control and Status (PMCS)—Offset 74h	8h
80h	81h	Message Signaled Interrupt Identifier (MID)—Offset 80h	7005h
82h	83h	Message Signaled Interrupt Message Control (MC)—Offset 82h	0h
84h	87h	Message Signaled Interrupt Message Address (MA)—Offset 84h	0h
88h	89h	Message Signaled Interrupt Message Data (MD)—Offset 88h	0h
90h	93h	Port Mapping Register (MAP)—Offset 90h	0h
94h	97h	Port Control and Status (PCS)—Offset 94h	0h
9Ch	9Fh	SATA General Configuration (SATAGC)—Offset 9Ch	0h
A0h	A0h	SATA Initialization Register Index (SIRI)—Offset A0h	0h
A4h	A7h	SATA Initialization Register Data (SIRD)—Offset A4h	0h
A8h	ABh	Serial ATA Capability Register 0 (SATACR0)—Offset A8h	100012h
ACh	AFh	Serial ATA Capability Register 1 (SATACR1)—Offset ACh	48h
C0h	C3h	Scratch Pad (SP)—Offset C0h	0h
D0h	D1h	MSI-X Identifiers (MXID)—Offset D0h	11h
D2h	D3h	MSI-X Message Control (MXC)—Offset D2h	0h
D4h	D7h	MSI-X Table Offset/Table BIR (MXT)—Offset D4h	0h
D8h	DBh	MSI-X PBA Offset / PBA BIR (MXP)—Offset D8h	0h

Table 24-1. Summary of SATA Configuration Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
E0h	E3h	BIST FIS Control/Status (BFCS)—Offset E0h	0h
E4h	E7h	BIST FIS Transmit Data 1 (BFTD1)—Offset E4h	0h
E8h	EBh	BIST FIS Transmit Data 2 (BFTD2)—Offset E8h	0h

24.8.1.1 Identifiers (ID)—Offset 0h

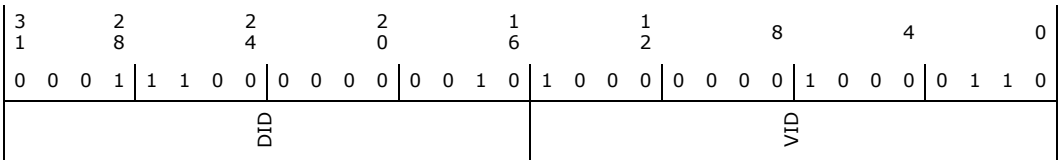
Identifiers

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 1C028086h



Bit Range	Default and Access	Field Name (ID): Description
31:16	1C02h RO	**Device ID (DID):** Indicates the Device ID of the SATA controller.
15:0	8086h RO	**Vendor ID (VID):** 16-bit field which indicates the company vendor as Intel.

24.8.1.2 Command (CMD)—Offset 4h

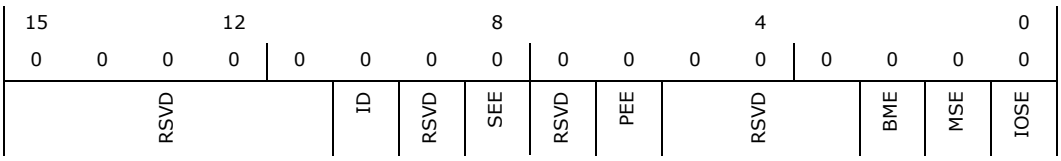
Command

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:11	0h RO	Reserved.
10	0h RW	**Interrupt Disable (ID):** This disables pin-based INTx# interrupts. This bit has no effect on MSI operation. When set, internal INTx# will not be generated. When cleared, internal INTx# are generated if there is an interrupt and MSI is not enabled.
9	0h RO	Reserved.
8	0h RW	**SERR# Enable (SEE):** When set to 1, the HBA is allowed to generate SERR# on DPD or SATAGC.URD event that is enabled for SERR# generation. When cleared to 0, it is not.
7	0h RO	Reserved.
6	0h RW	**Parity Error Response Enable (PEE):** When set, the SATA Controller will corrupt the outbound DATA FIS CRC if a forwarded data parity error is indicated.
5:3	0h RO	Reserved.
2	0h RW	**Bus Master Enable (BME):** Controls the SATA Controller's ability to act as a master for data transfers. This bit does not impact the generation of completions for split transaction commands.
1	0h RW	**Memory Space Enable (MSE):** Controls access to the SATA Controller's target memory space (for AHCI).
0	0h RW	**I/O Space Enable (IOSE):** Controls access to the SATA Controller's target I/O space.

24.8.1.3 Device Status (STS)—Offset 6h

Device Status

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 23
Function: 0

Default: 210h

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0
DPE	SSE	RMA	RTA	STA	DEVT		DPD	RSVD			CL	IS	RSVD		

Bit Range	Default and Access	Field Name (ID): Description
15	0h RW/1C	**Detected Parity Error (DPE):** Set when the SATA Controller detects a parity error on its interface.
14	0h RW/1C	**Signaled System Error (SSE):** Set when SATA Controller generates an SERR#.
13	0h RW/1C	**Received Master-Abort Status (RMA):** Set when the SATA Controller receives a master abort to a cycle it generated.
12	0h RW/1C	**Received Target-Abort Status (RTA):** Set when the SATA Controller receives a target abort to a cycle it generated.

Bit Range	Default and Access	Field Name (ID): Description
11	0h RW/1C	**Signaled Target-Abort Status (STA):** This bit must be set by a target device whenever it terminates a transaction with Target-Abort. Devices that will never signal Target-Abort do not need to implement this bit.
10:9	1h RO	**DEVSEL# Timing Status (DEVT):** Controls the device select time for the SATA Controller's PCI interface.
8	0h RW/1C	**Master Data Parity Error Detected (DPD):** Set when the SATA Controller, as a master, either detects a parity error or sees the parity error line asserted, and the parity error response bit (bit 6 of the command register) is set. This bit can only be set on read completions received from the backbone where there is a parity error.
7:5	0h RO	Reserved.
4	1h RO	**Capabilities List (CL):** Indicates the presence of a capabilities list. The minimum requirement for the capabilities list must be PCI power management for the SATA Controller.
3	0h RO	**Interrupt Status (IS):** Reflects the state of INTx# messages, IRQ14 or IRQ15. This bit is set when the interrupt is to be asserted. This bit is a 0 after the interrupt is cleared
2:0	0h RO	Reserved.

24.8.1.4 Revision ID (RID)—Offset 8h

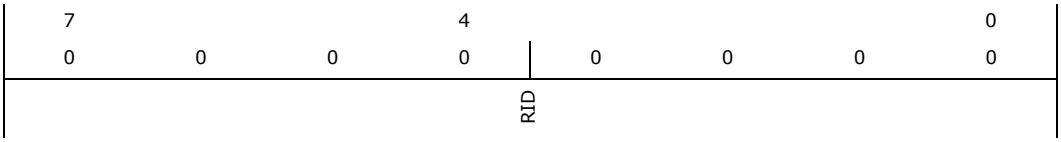
Revision ID

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO	**Revision ID (RID):** Indicates stepping of the host controller hardware.

24.8.1.5 Programming Interface (PI)—Offset 9h

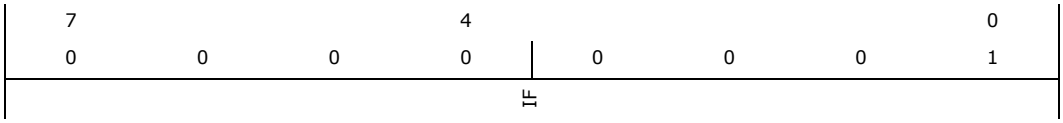
Programming Interface

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 23
Function: 0

Default: 1h



Bit Range	Default and Access	Field Name (ID): Description
7:0	1h RO	**Interface (IF):** If CC.SCC=06h(AHCI mode), it indicates that this is an AHCI HBA that has a major revision of 1 (as specified in the AHCI Version register). If CC.SCC=04h(RAID mode), it indicates that there is no programming interface(IF=00h). Internally, under this condition, the SATA controller is in native mode and its I/O spaces are only accessible through the I/O BARs.

24.8.1.6 Class Code (CC)—Offset Ah

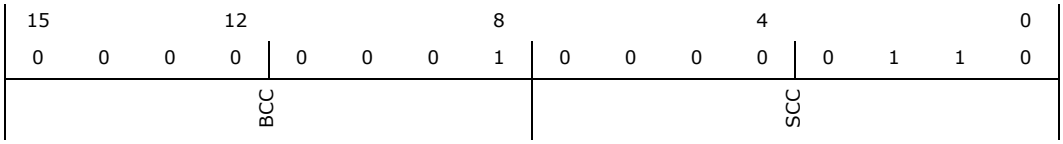
Class Code

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 23
Function: 0

Default: 106h



Bit Range	Default and Access	Field Name (ID): Description
15:8	1h RO	**Base Class Code (BCC):** Indicates that this is a mass storage device.
7:0	6h RO	**Sub Class Code (SCC):** Indicates the sub class code.

24.8.1.7 Cache Line Size (CLS)—Offset Ch

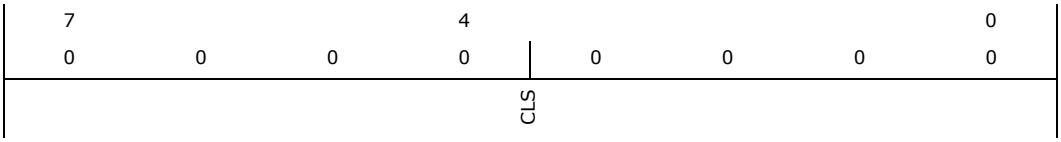
Cache Line Size

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO	**Cache Line Size (CLS):** This register has no meaning for the SATA controller.

24.8.1.8 Master Latency Timer (MLT)—Offset Dh

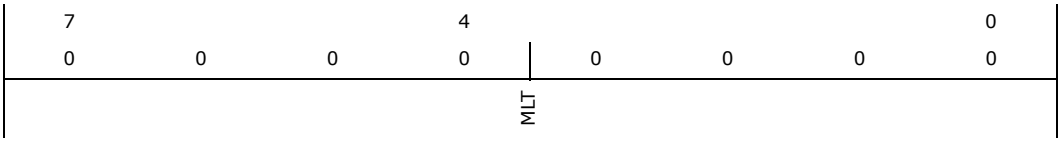
Master Latency Timer

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO	**Master Latency Timer (MLT):** This register has no meaning for the SATA controller.

24.8.1.9 Header Type (HTYPE)—Offset Eh

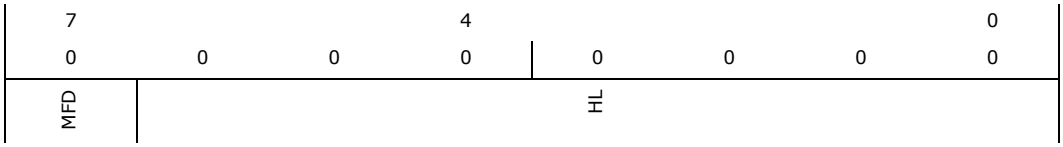
Header Type

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7	0h RO	**Multi-function Device (MFD):** Indicates this controller is not part of a multi-function device.
6:0	0h RO	**Header Layout (HL):** Indicates that the controller uses a target device layout.

24.8.1.10 MSI-X Table Base Address (MXTBA)—Offset 10h

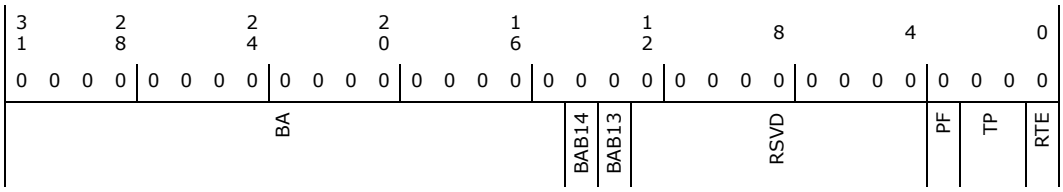
MSI-X Table Base Address. This BAR is used to allocate 32K, 16K or 8K Memory space for the MSI-X Table. The Memory space size is determined by the BIOS by making bit-14 and bit-13 Read-Only '1' or Read-Write '0' based on SATAGC.MSS[1:0].

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:15	0h RW	**Base Address (BA):** Base address of memory space.
14	0h RW	**Base Address Bit 14 (BAB14):** When SATAGC.MSS[1:0]=00, this bit is Read Only '0' else it's Read Write '0'.
13	0h RW	**Base Address Bit 13 (BAB13):** When SATAGC.MSS[1:0]=00 or 01, this bit is Read Only '0' else it's Read Write '0'.

Bit Range	Default and Access	Field Name (ID): Description
12:4	0h RO	Reserved.
3	0h RO	**Prefetchable (PF):** Indicates that this range is not pre-fetchable.
2:1	0h RO	**Type (TP):** Indicates that this range can be mapped anywhere in 32-bit address space.
0	0h RO	**Resource Type Indicator (RTE):** Indicates a request for Memory space.

24.8.1.11 MXP Base Address (MXPBA)—Offset 14h

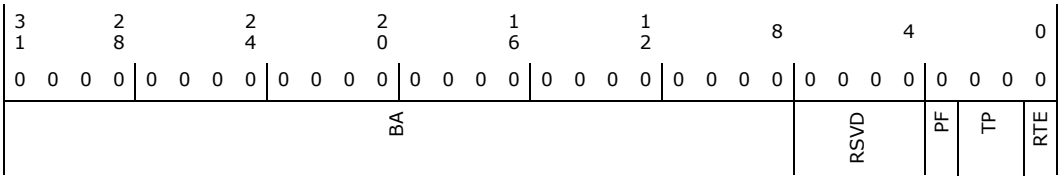
This BAR is used to allocate 256-byte Memory space for the MSI-X PBA.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:8	0h RW	**Base Address (BA):** Base address of memory space (aligned to 256B).
7:4	0h RO	Reserved.
3	0h RO	**Prefetchable (PF):** Indicates that this range is not pre-fetchable.
2:1	0h RO	**Type (TP):** Indicates that this range can be mapped anywhere in 32-bit address space.
0	0h RO	**Resource Type Indicator (RTE):** Indicates a request for Memory space.

24.8.1.12 AHCI Index Data Pair Base Address (AIDPBA)—Offset 20h

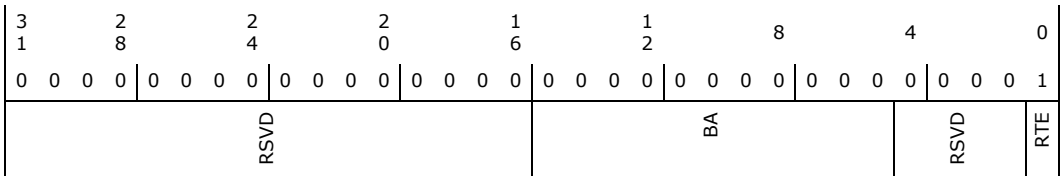
AHCI Index Data Pair Base Address. This BAR is used to allocate I/O space for the AHCI index/data pair mechanism. Note that hardware does not clear the BA bits (including BA4) when switching from IDE mode to non-IDE mode or vice versa. The BIOS is responsible for clearing those bits to 0 since the number of writable bits changes after mode switching.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 1h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:5	0h RW	**Base Address (BA):** Base address of the I/O space.
4:1	0h RO	Reserved.
0	1h RO	**Resource Type Indicator (RTE):** Indicates a request for I/O space.

24.8.1.13 AHCI Base Address (ABAR)—Offset 24h

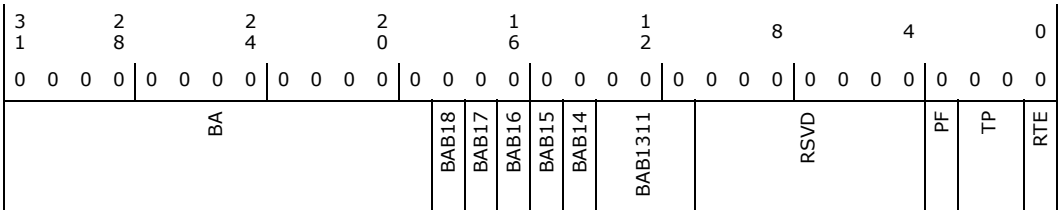
ABAR - AHCI Base Address. This register represents a memory BAR allocating space for the AHCI memory registers. Note that bit [31:16] of this register must be programmed to a value greater than 0 to ensure the memory is mapped to an address of 1 MB and greater (i.e., ABAR must be 00010000h or greater). Otherwise, memory cycle targeting the ABAR range may not be accepted. The Memory space size is determined by BIOS by making bit 15:11 Read-Only '1' or Read-Write '0' based on SATAGC.ASSEL[1:0].

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:19	0h RW	**Base Address (BA):** Base address of register memory space.
18	0h RW	**Base Address Bit 18 (BAB18):** When SATAGC.ASSEL[2:0] selects an ABAR size bigger than 256K, this bit is Read Only '0' else it's Read Write '0'.
17	0h RW	**Base Address Bit 17 (BAB17):** When SATAGC.ASSEL[2:0] selects an ABAR size bigger than 128K, this bit is Read Only '0' else it's Read Write '0'.
16	0h RW	**Base Address Bit 16 (BAB16):** When SATAGC.ASSEL[2:0] selects an ABAR size bigger than 64K, this bit is Read Only '0' else it's Read Write '0'.
15	0h RW	**Base Address Bit 15 (BAB15):** When SATAGC.ASSEL[2:0] selects an ABAR size bigger than 32K, this bit is Read Only '0' else it's Read Write '0'.
14	0h RW	**Base Address Bit 14 (BAB14):** When SATAGC.ASSEL[2:0] selects an ABAR size bigger than 16K, this bit is Read Only '0' else it's Read Write '0'.
13:11	0h RW	**Base Address Bit 13-11 (BAB1311):** When SATAGC.ASSEL[2:0] selects an ABAR size bigger than 2K, this bit is Read Only '0' else it's Read Write '0'.
10:4	0h RO	Reserved.
3	0h RO	**Prefetchable (PF):** Indicates that this range is not pre-fetchable.
2:1	0h RO	**Type (TP):** Indicates that this range can be mapped anywhere in 32-bit address space.
0	0h RO	**Resource Type Indicator (RTE):** Indicates a request for register memory space.

24.8.1.14 Sub System Identifiers (SS)—Offset 2Ch

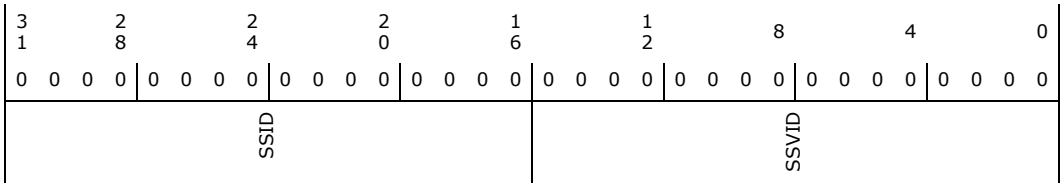
Sub System Identifiers. This register is initialized to logic 0 by the assertion of PLTRST#. This register can be written only once after PLTRST# de-assertion. This register is not reset by FLR.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RW/O	**Subsystem ID (SSID):** This is written by BIOS. No hardware action taken on this value.
15:0	0h RW/O	**Subsystem Vendor ID (SSVID):** This is written by BIOS. No hardware action taken on this value.

24.8.1.15 Capabilities Pointer (CAP)—Offset 34h

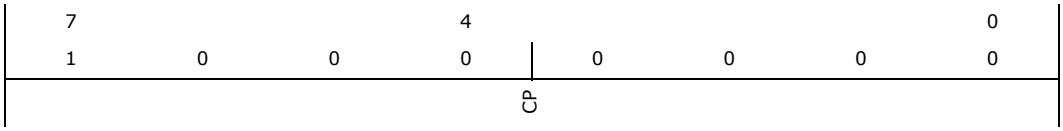
Capabilities Pointer

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 23
Function: 0

Default: 80h



Bit Range	Default and Access	Field Name (ID): Description
7:0	80h RW/L	**Capability Pointer (CP):** Indicates that the first capability pointer offset is offset 80h (the Message Signaled Interrupt capability). The following capability structures are linked by default: CAP.CP -> 80h (MSI) -> D0h (MSI-X)-> 70h (PCI Power) -> A8h (SATA) -> 00h end. BIOS may alter the capability structure list above (by programming a leading capability structure's Next Pointer field) if BIOS wants to bypass any specific capability. The RW/L register attribute allows for flexibility in determining the capability structure available in this PCI function. Refer to SATAGC.REGLOCK description in order to lock the register to become RO. This register is not reset by FLR.

24.8.1.16 Interrupt Information (INTR)—Offset 3Ch

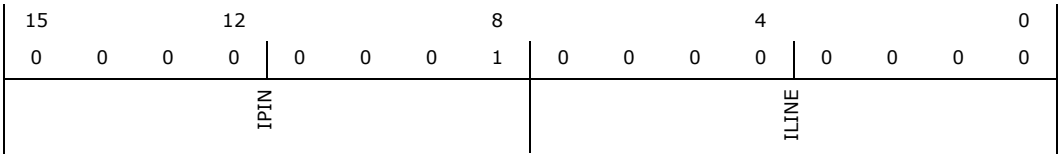
Interrupt Information

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 23
Function: 0

Default: 100h



Bit Range	Default and Access	Field Name (ID): Description
15:8	1h RW/O	**Interrupt Pin (IPIN):** This register tells which interrupt pin the device function uses. A value of 1 corresponds to INTA#. A value of 2 corresponds to INTB#. A value of 3 corresponds to INTC#. A value of 4 corresponds to INTD#. This register is not reset by FLR.
7:0	0h RW	**Interrupt Line (ILINE):** Software written value to indicate which interrupt line (vector) the interrupt is connected to. No hardware action is taken on this register. Interrupt Line register is not reset by FLR.

24.8.1.17 PCI Power Management Capability ID (PID)—Offset 70h

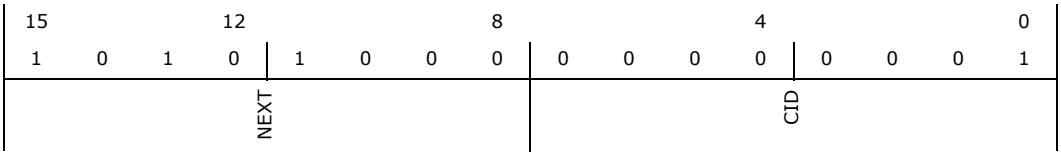
PCI Power Management Capability ID

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 23
Function: 0

Default: A801h



Bit Range	Default and Access	Field Name (ID): Description
15:8	A8h RW/L	**Next Capability (NEXT):** A8h is location of the Serial ATA Capability structure. The RW/L register attribute allows for flexibility in determining the capability structure available in this PCI function. Refer to SATAGC.REGLOCK description in order to lock the register to become RO. This register is not reset by FLR.
7:0	1h RO	**Cap ID (CID):** Indicates that this pointer is a PCI power management capability.

24.8.1.18 PCI Power Management Capabilities (PC)—Offset 72h

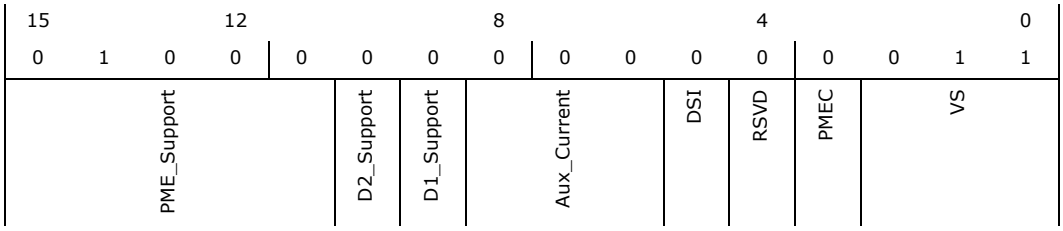
PCI Power Management Capabilities

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 23
Function: 0

Default: 4003h



Bit Range	Default and Access	Field Name (ID): Description
15:11	8h RO	**PME_Support (PME_Support):** The default value is 01000 which indicates PME# can be generated from the $D3_{HOT}$ state in the SATA controller.
10	0h RO	**D2_Support (D2_Support):** The D2 state is not supported.
9	0h RO	**D1_Support (D1_Support):** The D1 state is not supported.
8:6	0h RO	**Aux_Current (Aux_Current):** PME# from $D3_{COLD}$ state is not supported, therefore this field is 000b.
5	0h RO	**Device Specific Initialization (DSI):** Indicates that no device-specific initialization is required.
4	0h RO	Reserved.
3	0h RO	**PME Clock (PMEC):** Indicates that PCI clock is not required to generate PME#.
2:0	3h RO	**Version (VS):** Indicates support for Revision 1.2 of the PCI Power Management Specification.

24.8.1.19 PCI Power Management Control and Status (PMCS)—Offset 74h

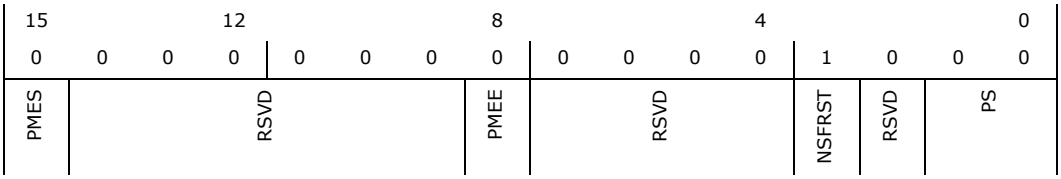
PCI Power Management Control and Status

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 23
Function: 0

Default: 8h



Bit Range	Default and Access	Field Name (ID): Description
15	0h RW/1C	**PME Status (PMES):** This bit is set when a PME event is to be requested, and if this bit is set and PMEE is set, a PME# will be generated. This register field is not reset by FLR.
14:9	0h RO	Reserved.
8	0h RW	**PME Enable (PMEE):** When set, the SATA controller generates PME# from $D3_{HOT}$ on a wake event. ***Note:*** Software is advised to clear PMEE together with PMES prior to changing CC.SCC through SATAGC.SMS. This register field is not reset by FLR.
7:4	0h RO	Reserved.
3	1h RO	**No Soft Reset (NSFRST):** A 1 indicates that devices transitioning from $D3_{HOT}$ to D0 because of PowerState commands do not perform an internal reset. Configuration context is preserved. Upon transition from the $D3_{HOT}$ to the D0 state initialized state, no additional operating system intervention is required to preserve configuration context beyond writing to the PowerState bits. Regardless of this bit, the controller transition from $D3_{HOT}$ to the D0 by a system or bus segment reset will return to the state D0 uninitialized with only PME context preserved if PME is supported and enabled.
2	0h RO	Reserved.
1:0	0h RW	**Power State (PS):** This field is used both to determine the current power state of the SATA Controller and to set a new power state. The values are: 00 = D0 state; 11 = $D3_{HOT}$ state. When in the $D3_{HOT}$ state, the controller's configuration space is available, but the I/O and memory spaces are not. Additionally, interrupts are blocked. If software attempts to write a 10 or 01 to these bits, the write will be ignored. Refer to *PCI PM Specification* Section 8.2.2. on software requirements in ensuring I/O space, memory space and Bus Master are disabled prior to entering D3 state.

24.8.1.20 Message Signaled Interrupt Identifier (MID)—Offset 80h

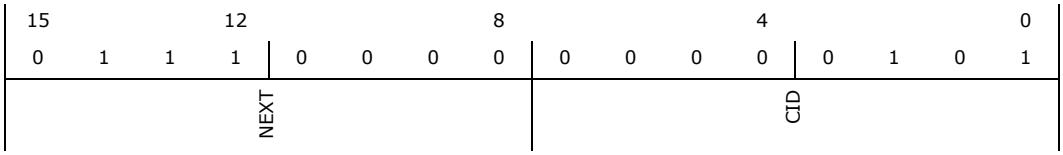
Message Signaled Interrupt Identifier

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 23
Function: 0

Default: 7005h



Bit Range	Default and Access	Field Name (ID): Description
15:8	70h RW/L	**Next Pointer (NEXT):** Indicates the next item in the list is the PCI power management pointer. This is the recommended value. BIOS may program this field to A8h indicating that the next item is Serial ATA Capability structure. The RW/L register attribute allows for flexibility in determining the capability structure available in this PCI function. Refer to SATAGC.REGLOCK description in order to lock the register to become RO. This register is not reset by FLR.
7:0	5h RO	**Capability ID (CID):** Capabilities ID indicates MSI.

24.8.1.21 Message Signaled Interrupt Message Control (MC)—Offset 82h

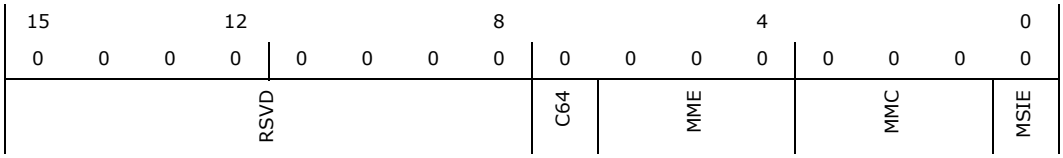
Message Signaled Interrupt Message Control

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:8	0h RO	Reserved.
7	0h RO	**64 Bit Address Capable (C64):** Capable of generating a 32-bit message only.
6:4	0h RO	**Multiple Message Enable (MME):** When this field is cleared to 000 (and MSIE is set), only a single MSI message will be generated for all SATA ports, and bits [15:0] of the message vector will be driven from MD[15:0].

Bit Range	Default and Access	Field Name (ID): Description
3:1	0h RO	**Multiple Message Capable (MMC):** Not supported.
0	0h RW	**MSI Enable (MSIE):** If set, MSI is enabled and traditional interrupt pins are not used to generate interrupts. Note that CMD.ID bit has not effect on MSI. Software must clear this bit to 0 to disable MSI first before changing the number of messages allocated in the MMC field.

24.8.1.22 Message Signaled Interrupt Message Address (MA)—Offset 84h

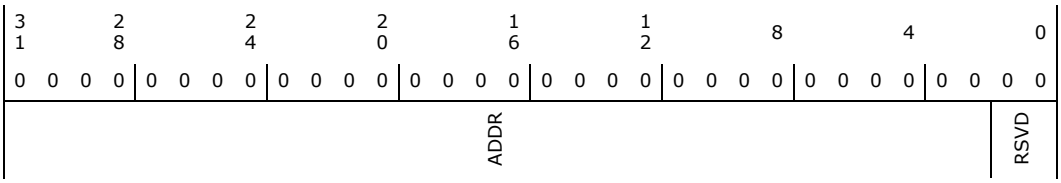
Message Signaled Interrupt Message Address

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:2	0h RW	**Address (ADDR):** Lower 32 bits of the system specified message address, always DWORD aligned.
1:0	0h RO	Reserved.

24.8.1.23 Message Signaled Interrupt Message Data (MD)—Offset 88h

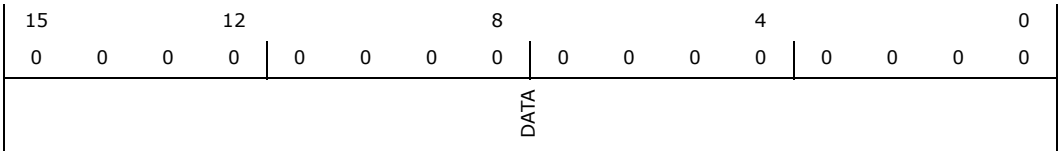
Message Signaled Interrupt Message Data

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:0	0h RW	**Data (DATA):** This 16-bit field is programmed by system software if MSI is enabled. Its content is driven onto the lower word of the data bus of the MSI memory write transaction.

24.8.1.24 Port Mapping Register (MAP)—Offset 90h

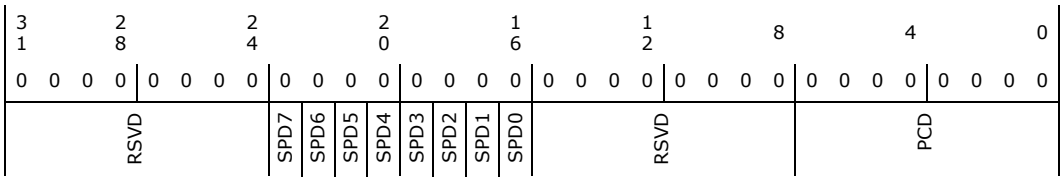
Port Mapping Register

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23	0h RW/O	**SATA Port 7 Disable (SPD7):** Similar to SPD0 but for port 7. This bit is RO-one if ANY of the fuse/strap/GPIO condition below is true else this bit is RW-zero. PCIe/SATA muxing for port 7 based on fuses, soft straps and GPIO does not select SATA. This bit is only applicable to project(s): that has port 7 physically.
22	0h RW/O	**SATA Port 6 Disable (SPD6):** Similar to SPD0 but for port 6. This bit is RO-one if ANY of the fuse/strap/GPIO condition below is true else this bit is RW-zero. PCIe/SATA muxing for port 6 based on fuses, soft straps and GPIO does not select SATA. This bit is only applicable to project(s): that has port 6 physically.
21	0h RW/O	**SATA Port 5 Disable (SPD5):** Similar to SPD0 but for port 5. This bit is RO-one if ANY of the fuse/strap/GPIO condition below is true else this bit is RW-zero. PCIe/SATA muxing for port 5 based on fuses, soft straps and GPIO does not select SATA. This bit is only applicable to project(s): that has port 5 physically.
20	0h RW/O	**SATA Port 4 Disable (SPD4):** Similar to SPD0 but for port 4. This bit is RO-one if ANY of the fuse/strap/GPIO condition below is true else this bit is RW-zero. PCIe/SATA muxing for port 4 based on fuses, soft straps and GPIO does not select SATA. This bit is only applicable to project(s): that has port 4 physically.
19	0h RW/O	**SATA Port 3 Disable (SPD3):** Similar to SPD0 but for port 3. This bit is RO-one if ANY of the fuse/strap/GPIO condition below is true else this bit is RW-zero. 1.Fuse FFSATA7 (disable port 2 and 3). 2.Fuse FFSATA8 (disable port 1 and 3). 3. PCIe/SATA muxing for port 3 based on fuses, soft straps and GPIO does not select SATA. This bit is only applicable to project(s): that has port 3 physically.
18	0h RW/O	**SATA Port 2 Disable (SPD2):** Similar to SPD0 but for port 2. This bit is RO-one if ANY of the fuse/strap/GPIO condition below is true else this bit is RW-zero. 1.Fuse FFSATA7 (disable port 2 and 3). 2.PCIe/SATA muxing for port 3 based on fuses, soft straps and GPIO does not select SATA. This bit is only applicable to project(s): that has port 3 physically.
17	0h RW/O	**SATA Port 1 Disable (SPD1):** Similar to SPD0 but for port 1. This bit is RO-one if ANY of the fuse/strap/GPIO condition below is true else this bit is RW-zero. 1.Fuse FFSATA8 (disable port 1 and 3). 2.PCIe/SATA muxing for port 3 based on fuses, soft straps and GPIO does not select SATA. This bit is only applicable to project(s): that has port 1 physically.

Bit Range	Default and Access	Field Name (ID): Description
16	0h RW/O	**SATA Port 0 Disable (SPD0):** A 1 prevents the SATA port from being enabled via config PCS.PxE. Write of 1 to PCS.PxE has no effect when the corresponding SPD[x] bit is 1.In preventing a port(s) from being enabled, BIOS shall first configure MAP.SPDx. And only then BIOS configures the PCS.PxE. This field is not reset by FLR. This bit is only applicable to project(s): that has port 0 physically.
15:8	0h RO	Reserved.
7:0	0h RW	**Port Clock Disable (PCD):** When any of these bits is set to 1, the backbone clock driven to the associated port logic is gated and will not toggle. When this bit is cleared to 0, all clocks to the associated port logic will operate normally. Assignment of the bits is: Bit 7: Port 7, this bit is only applicable to project(s): that has port 7 physically; Bit 6: Port 6, this bit is only applicable to project(s): that has port 6 physically; Bit 5: Port 5, this bit is only applicable to project(s): that has port 5 physically; Bit 4: Port 4, this bit is only applicable to project(s): that has port 4 physically; Bit 3: Port 3, this bit is only applicable to project(s): that has port 3 physically; Bit 2: Port 2, this bit is only applicable to project(s): that has port 2 physically; Bit 1: Port 1, this bit is only applicable to project(s): that has port 1 physically; Bit 0: Port 0, this bit is only applicable to project(s): That has port 0 physically. If a particular port is not available, software shall set the corresponding bit to 1. Software can also set the corresponding bit(s) to 1 after disabling particular port(s). Software cannot set the PCD[port x]=1 if the corresponding config PCS.PxE=1 or AHCI MMIO GHC.PI[x]=1.

24.8.1.25 Port Control and Status (PCS)—Offset 94h

Port Control and Status. By default, the SATA ports are set (by hardware) to the disabled state (e.g., bits [5:0] == '0') as a result of an initial power on reset. When enabled by software, the ports can transition between the on, partial, and slumber states and can detect devices. When disabled, the port is in the off state and cannot detect any devices. This register is not reset by FLR.

AHCI specific notes:

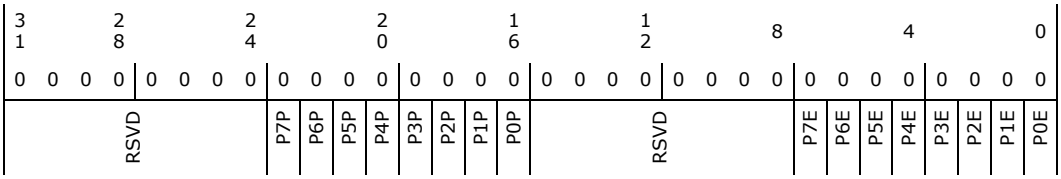
If an AHCI-aware or RAID enabled operating system is being booted then system BIOS shall insure that all supported SATA ports are enabled prior to passing control to the OS. Once the AHCI aware OS is booted it becomes the enabling/disabling policy owner for the individual SATA ports. This is accomplished by manipulating a port's PxSCTL.DET and PxCMD.SUD fields. Because an AHCI or RAID aware OS will typically not have knowledge of the PxE bits and because the PxE bits act as master on/off switches for the ports, pre-boot software must insure that these bits are set to '1' prior to booting the OS, regardless as to whether or not a device is currently on the port.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23	0h RO	**Port 7 Present (P7P):** Same as P0P, except for port 7. This bit is only applicable to project(s): That has port 7 physically.
22	0h RO	**Port 6 Present (P6P):** Same as P0P, except for port 6. This bit is only applicable to project(s): That has port 6 physically.
21	0h RO	**Port 5 Present (P5P):** Same as P0P, except for port 5. This bit is only applicable to project(s): That has port 5 physically.
20	0h RO	**Port 4 Present (P4P):** Same as P0P, except for port 4. This bit is only applicable to project(s): That has port 4 physically.
19	0h RO	**Port 3 Present (P3P):** Same as P0P, except for port 3. This bit is only applicable to project(s): That has port 3 physically.
18	0h RO	**Port 2 Present (P2P):** Same as P0P, except for port 2. This bit is only applicable to project(s): That has port 2 physically.
17	0h RO	**Port 1 Present (P1P):** Same as P0P, except for port 1. This bit is only applicable to project(s): That has port 1 physically.
16	0h RO	**Port 0 Present (P0P):** When set, the SATA controller has detected the presence of a device on port 0. It may change at any time. Clearing P0E bit leads to clearing of this bit after implementation delay. ***Note:*** For system software that intends to clear all PCS.PxE bits that are previously 1 to 0 and then to 1 again in two consecutive write cycles, software shall poll on this bit being 0 before setting P0E bit to 1. This bit is only applicable to project(s): That has port 0 physically.
15:8	0h RO	Reserved.
7	0h RW	**Port 7 Enabled (P7E):** When MAP.SPD[7] is 1, this bit is reserved and is read-only 0. Otherwise if none of the above is true, this field is RW and the definition of this bit is the same as P0E, except for port 7. This bit takes precedence over P7CMD.SUD. This bit is only applicable to project(s): That has port 7 physically.
6	0h RW	**Port 6 Enabled (P6E):** When MAP.SPD[6] is 1, this bit is reserved and is read-only 0. Otherwise if none of the above is true, this field is RW and the definition of this bit is the same as P0E, except for port 6. This bit takes precedence over P6CMD.SUD. This bit is only applicable to project(s): That has port 6 physically.
5	0h RW	**Port 5 Enabled (P5E):** When MAP.SPD[5] is 1, this bit is reserved and is read-only 0. Otherwise if none of the above is true, this field is RW and the definition of this bit is the same as P0E, except for port 5. This bit takes precedence over P5CMD.SUD. This bit is only applicable to project(s): That has port 5 physically.
4	0h RW	**Port 4 Enabled (P4E):** When MAP.SPD[4] is 1, this bit is reserved and is read-only 0. Otherwise if none of the above is true, this bit is RW and same as P0E but for port 4 and takes precedence over P4CMD.SUD. This bit is only applicable to project(s): That has port 4 physically.
3	0h RW	**Port 3 Enabled (P3E):** When MAP.SPD[3] is 1, this bit is reserved and is read-only 0. Otherwise if none of the above is true, this bit is RW and same as P0E but for port 3 and takes precedence over P3CMD.SUD. This bit is only applicable to project(s): That has port 3 physically.
2	0h RW	**Port 2 Enabled (P2E):** When MAP.SPD[2] is 1, this bit is reserved and is read-only 0. Otherwise if none of the above is true, this bit is RW and same as P0E but for port 2 and takes precedence over P2CMD.SUD. This bit is only applicable to project(s): That has port 2 physically.
1	0h RW	**Port 1 Enabled (P1E):** When MAP.SPD[1] is 1, this bit is reserved and read-only 0. Otherwise if none of the above is true, this bit is RW and same as P0E but for port 1 and takes precedence over P1CMD.SUD. This bit is only applicable to project(s): That has port 1 physically.

Bit Range	Default and Access	Field Name (ID): Description
0	0h RW	**Port 0 Enabled (P0E):** When MAP.SPD[0] is 1, this bit is reserved and read-only 0. When set, the port is enabled. When cleared, the port is disabled. When enabled, the port can transition between the on, partial, and slumber states and can detect devices. When disabled, the port is in the off state and cannot detect any devices. This bit takes precedence over P0CMD.SUD. ***Note:*** The recommendation for software code that intends to clear all PCS.PxE bits that are previously 1 to 0 and then to 1 again immediately shall refer to the polling requirement as described in P0P register bit. At any time that BIOS or software is clearing PCS.PxE from 1 to 0, due to time needed for port staggering hardware process (up to 6 ports) to complete, BIOS and software shall delay the write to set the MAP.PCD register by 1.4us. This bit is only applicable to project(s): That has port 0 physically.

24.8.1.26 SATA General Configuration (SATAGC)—Offset 9Ch

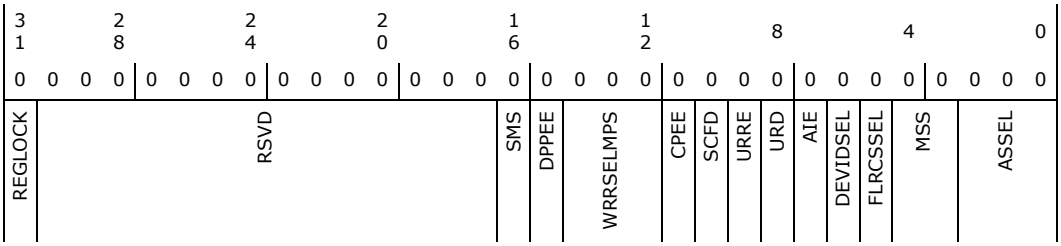
SATA General Configuration

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RW/O	**Register Lock (REGLOCK):** BIOS can set this bit to 1 to lock the following registers with RW/L attribute: CAP,CP, MID.NEXT, PIE.NEXT, SATACR0.NEXT. Once locked the register attribute of above list changes from RW/L to RO holding the existing value. BIOS is required to program this field to 1 prior to handle off to OS. If BIOS needs the SATA host controller to change operation a few times (i.e., changing CC.SCC mode) and need different capability structures for each specific operation mode, BIOS need not activate the lock until BIOS is ready to hand off to OS. BIOS may need to separate write access to this byte offset (x9Fh) from write to the lower 3-byte of the dword (x9C-9Eh) if there is a need to program the lower 3-byte dword location early during boot process. This field is not reset by FLR.
30:17	0h RO	Reserved.
16	0h RW	**SATA Mode Select (SMS):** Software (SW) programs these bits to control the mode in which the SATA HBA should operate: 0b = AHCI mode 1b = RAID mode ***Notes:*** SW shall not manipulate SATAGC.SMS during runtime operation i.e., the OS will not do this. The BIOS may choose to switch from one mode to another during POST; AHCI mode may be selected when RAID feature is enabled by fuse; RAID mode may only be selected when FFSATA5 and FFSATA3 (concatenated value not indicating No RAID); This register field is not reset by FLR.
15	0h RW	**Data Phase Parity Error Enable (DPPEE):** When 1, IOSF data phase parity error handling is enabled. When 0, the data phase parity error handling is disabled.

Bit Range	Default and Access	Field Name (ID): Description
14:12	0h RW	**Write Request_Size Select/Max_Payload_Size (WRRSELMPS):** These two bits select the max write request size that SATA host controller will initiate for DMA write to memory. SATA host controller will internally break up larger write request based on these bits. The request is address-aligned to the selected size. Defined encodings for this field are: 000b = 128 address aligned bytes max payload size; 111b = 64 address aligned bytes max payload size. All other values are reserved for SATA host controller. This field is not reset by FLR.
11	0h RW	**Command Parity Error Enable (CPEE):** When 1, command parity error handing is enable. When 0 the command parity error handling is disable.
10	0h RW	**SATA Controller Function Disable (SCFD):** BIOS program this bit to 1 to disable the SATA Controller function. When 0, SATA Controller function is enabled. When disable, SATA Host Controller will not claimed the register access targeting its Configuration Space. In IOSF primary Fabric Decode scheme, it's expected BIOS also program the corresponding bit used by the Fabric Decoder accordingly hence both SATA SIP and Fabric Decoder are in sync, and BIOS need to program this bit before programming the one in Fabric Decoder. Once this bit is set, BIOS is not able to revert it back to Function Enable until next round of platform reset.
9	0h RW	**Unsupported Request Reporting Enable (URRE):** If set to 1 by software, it allows reporting of an Unsupported Request as a system error. If both URRE and PCI configuration SERR# Enable registers are set to a 1, then the agent must set the Signaled System Error bit in the PCI Status register and send a DO_SERR message in IOSF-SB interface.
8	0h RW/1C	**Unsupported Request Detected (URD):** Set to 1 by hardware upon detecting an Unsupported Request on IOSF Primary interface that is not considered Advisory Non-Fatal. Cleared to 0 by SW.
7	0h RW/O	**Alternate ID Enable (AIE):** When programmed to 0, HW will report the following device id's: 2822h for desktop or 282Ah for mobile. When programmed to a 1, HW will not report these device IDs. ***Note:*** Programming this bit to a 1 will prevent the Windows in-box version of the Intel AHCI driver from loading on the platform - will require that the user perform an 'F6' install of the Intel driver that is appropriate for the reported DID. This field is applicable when the AHCI is configured for RAID mode of operation. It has no impact for AHCI and IDE modes of operation. ***Note:*** The BIOS is recommended to program this bit prior to programming the MAP.SMS field to reflect RAID. This field is reset by PLTRST# and BIOS is required to reprogram the value (either 0 or 1) after resuming from S3, S4 or S5.
6	0h RW/O	**AIE0 DevID Selection (DEVIDSEL):** This register allows BIOS to select Device ID when AIE=0 and Server Feature (SATA AIE DEVIDSEL) Disable Fuse =0. This bit only has effect in Desktop SKU. In Mobile SKU this bit has no effect at all. Refer to config register offset 09h PI for usage. ***Note:*** WBG BIOS is required to program this field to 1 together with the write to the AIE bit in a single configuration write cycle. LPT BIOS is required to program this bit to 0 together with the write to the AIE bit in a single configuration write cycle. When Server Feature (SATA AIE DEVIDSEL) Disable Fuse is programmed to 1, this disables the writeability of this DEVIDSEL register bit, and becomes RO with a value of 0, which only allows a choice of 2822h. This field is not reset by FLR.
5	0h RW/O	**FLR Capability Selection (FLRCSSEL):** This allows the FLR Capability to be bypassed. Refer to config offset B0h. BIOS is required to program this bit to 1 and config offset A8h SATACR0.NEXT to 00h. This field is not reset by FLR.
4:3	0h RW/O	**MXTBA Size Select (MSS):** These 2 bits select the size of the Memory space for the MSI-X Table defined in BAR 0 (Configuration space offset 10h). MSI-X Table Memory space size is 32k when MSS[1:0]=00, 16k when MSS[1:0]=01, 8k when MSS[1:0]=10. This field is not reset by FLR.
2:0	0h RW/O	**ABAR Size Select (ASSEL):** These 3 bits select the size of the Memory space for the ABAR in BAR 5 (Configuration space offset 24h). ABAR Memory space size is 2k when ASSEL[2:0]=000, 16k when ASSEL[2:0]=001, 32k when ASSEL[2:0]=010, 64k when ASSEL[2:0]=011, 128k when ASSEL[2:0]=100, 256k when ASSEL[2:0]=101, 512k when ASSEL[2:0]=110. This field is not reset by FLR.

24.8.1.27 SATA Initialization Register Index (SIRI)—Offset A0h

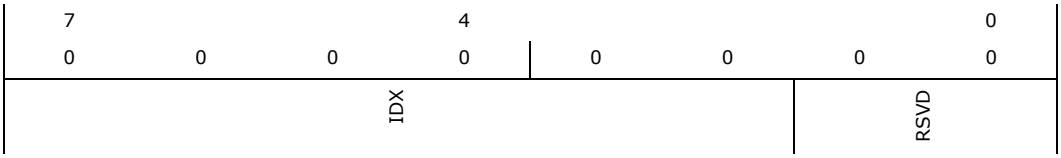
SATA Initialization Register Index

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:2	0h RW	**Index (IDX):** 6-bit index pointer into the 256-byte space. Data is written into the SIRD register and read from the SIRD register. This point to a DWord register. The byte enables on the SIRD register affect what will be written. Refer to SATA Initialization Register section for detail of the register space.
1:0	0h RO	Reserved.

24.8.1.28 SATA Initialization Register Data (SIRD)—Offset A4h

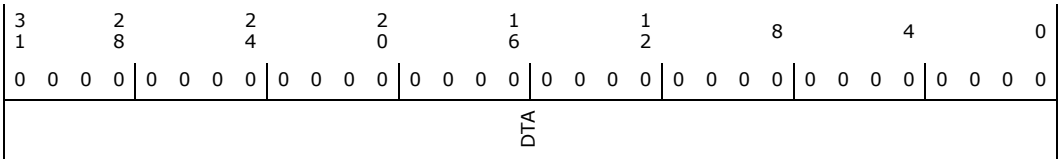
SATA Initialization Register Data

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Data (DTA):** 32-bit data value that is written to the register pointed to by SIRI, or read from the register pointed to by SIRI.

24.8.1.29 Serial ATA Capability Register 0 (SATACR0)—Offset A8h

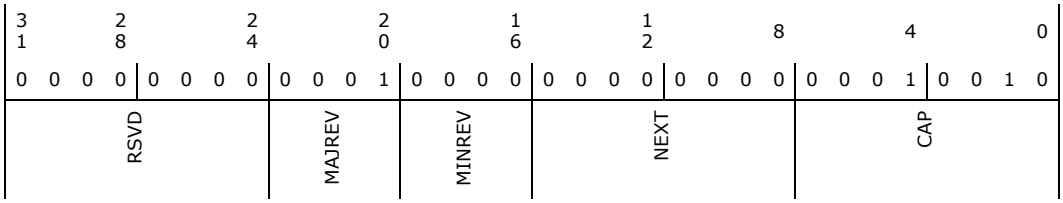
Note that the SATACR0.NEXT is not changed from RO to become RWO because there is an existing method (SATAGC.FLRCSSEL bit) to bypass the FLR Capability structure. And FLR Capability ID.NEXT is already indicating end of capability structure, it does not need change to be RWO.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 100012h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23:20	1h RO	**Major Revision (MAJREV):** Major revision number of the SATA Capability Pointer implemented.
19:16	0h RO	**Minor Revision (MINREV):** Minor revision number of the SATA Capability Pointer implemented.
15:8	0h RW/L	**Next Capability Pointer (NEXT):** 00h indicating the final item in the Capability List. The RW/L register attribute allows for flexibility in determining the capability structure available in this PCI function. Refer to SATAGC.REGLOCK description in order to lock the register to become RO. This register is not reset by FLR.
7:0	12h RO	**Capability ID (CAP):** The value of 12h has been assigned by the PCI SIG to designate the SATA Capability pointer.

24.8.1.30 Serial ATA Capability Register 1 (SATACR1)—Offset ACh

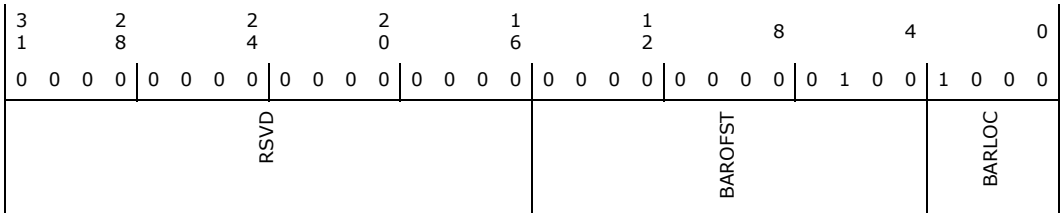
Serial ATA Capability Register 1

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 48h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:4	4h RO	**BAR Offset (BAROFST):** Indicates the offset into the BAR where the AHCI Index/Data pair are located (in Dword granularity). The Index and Data I/O registers are located at offset 10h within the I/O space defined by LBAR (BAR4). A value of 004h indicates offset 10h. 000h = 0h offset; 001h = 4h offset; 002h = 8h offset; 003h = Ch offset; 004h = 10h offset;...; FFFh = 3FFFh offset (max 16 KB)
3:0	8h RO	**BAR Location (BARLOC):** Indicates the absolute PCI Configuration Register address of the BAR containing the Index/Data pair (in Dword granularity). The Index and Data I/O registers reside within the space defined by LBAR (BAR4) in the SATA controller. A value of 8h indicates offset 20h, which is LBAR (BAR4). 0000 - 0011b = reserved; 0100b = 10h => BAR0; 0101b = 14h => BAR1; 0110b = 18h => BAR2; 0111b = 1Ch => BAR3; 1000b = 20h => AIDPBA; 1001b = 24h => BAR5; 1010 - 1110b = reserved; 1111b = Index/Data pair in PCI Configuration space which is not supported.

24.8.1.31 Scratch Pad (SP)—Offset C0h

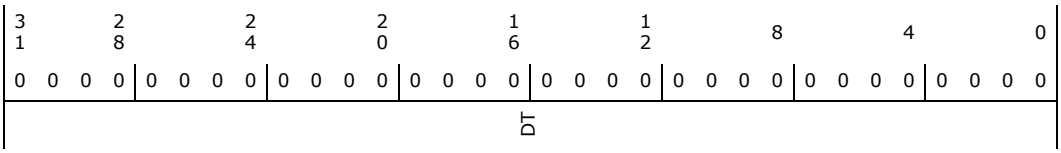
Scratch Pad

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Data (DT):** This is a read/write register that is available for software to use. No hardware action is taken on this register.

24.8.1.32 MSI-X Identifiers (MXID)—Offset D0h

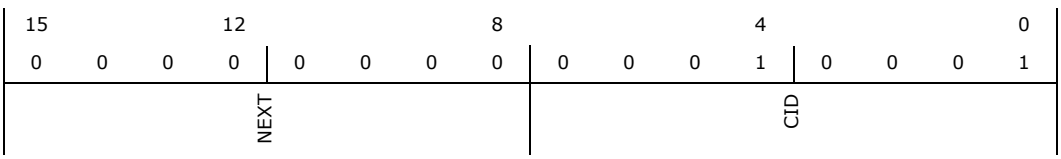
MSI-X Identifiers

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 23
Function: 0

Default: 11h



Bit Range	Default and Access	Field Name (ID): Description
15:8	0h RW/L	**Next Pointer (NEXT):** Indicates the next item in the list. This may be other capability pointers or it may be the last item in the list.The RW/L register attribute allows for flexibility in determining the capability structure available in this PCI function. Refer to SATAGC.REGLOCK description in order to lock the register to become RO. This register is not reset by FLR.
7:0	11h RO	**Capability ID (CID):** Capabilities ID indicates this is an MSI-X capability.

24.8.1.33 MSI-X Message Control (MXC)—Offset D2h

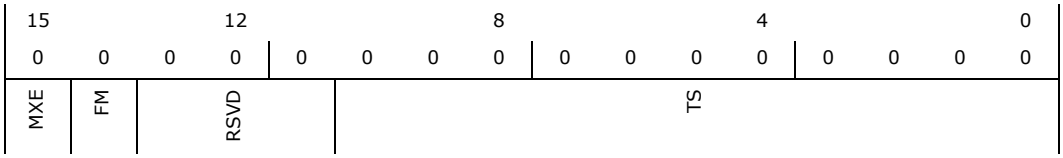
MSI-X Message Control

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15	0h RW	**MSI-X Enable (MXE):** If set to '1' and the MSI Enable bit in the MSI Message Control register is cleared to '0', the function is permitted to use MSI-X to request service and is prohibited from using its INTx# pin (if implemented). If cleared to '0', the function is prohibited from using MSI-X to request service.
14	0h RW	**Function Mask (FM):** If set to '1', all of the vectors associated with the function are masked, regardless of their per vector Mask bit states. If cleared to '0', each vector's Mask bit determines whether the vector is masked or not. Setting or clearing the MSI-X Function Mask bit has no effect on the state of the per vector Mask bits.
13:11	0h RO	Reserved.
10:0	0h RO	**Table Size (TS):** This value indicates the size of the MSI-X Table as the value n, which is encoded as n - 1. For example, a returned value of 3h corresponds to a table size of 4.

24.8.1.34 MSI-X Table Offset/Table BIR (MXT)—Offset D4h

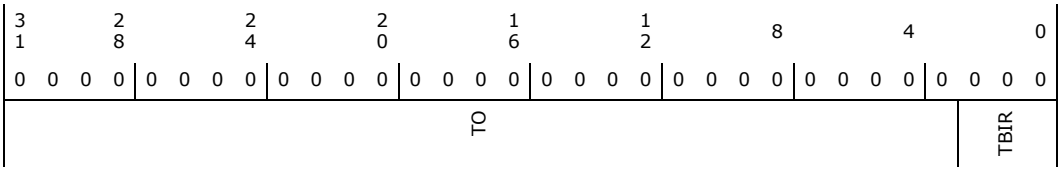
MSI-X Table Offset/Table BIR

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:3	0h RO	**Table Offset (TO):** Used as an offset from the address contained by one of the function's Base Address registers to point to the base of the MSI-X Table. The lower three Table BIR bits are masked off (cleared to 000b) by system software to form a 32-bit Qword-aligned offset.
2:0	0h RO	**Table BIR (TBIR):** This field indicates which one of a function's Base Address registers, located beginning at 10h in Configuration Space, is used to map the function's MSI-X Table into system memory. A read-only value of '0' means 10h.

24.8.1.35 MSI-X PBA Offset / PBA BIR (MXP)—Offset D8h

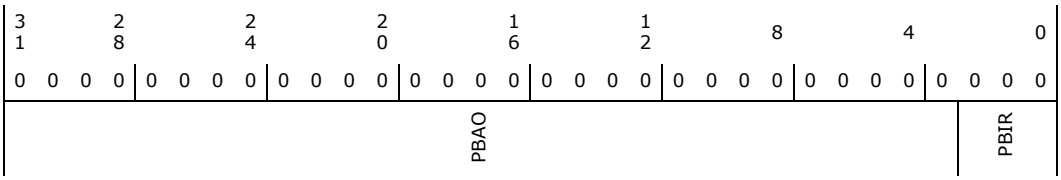
MSI-X PBA Offset / PBA BIR

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:3	0h RO	**PBA Offset (PBAO):** Used as an offset from the address contained by one of the function's Base Address registers to point to the base of the MSI-X PBA. The lower three PBA BIR bits are masked off (cleared to 000b) by software to form a 32-bit Qword-aligned offset.
2:0	0h RO	**PBA BIR (PBIR):** This field indicates which one of a function's Base Address registers, located beginning at 10h in Configuration Space, is used to map the function's MSI-X PBA into system memory. A read-only value of '1' means 14h.

24.8.1.36 BIST FIS Control/Status (BFCS)—Offset E0h

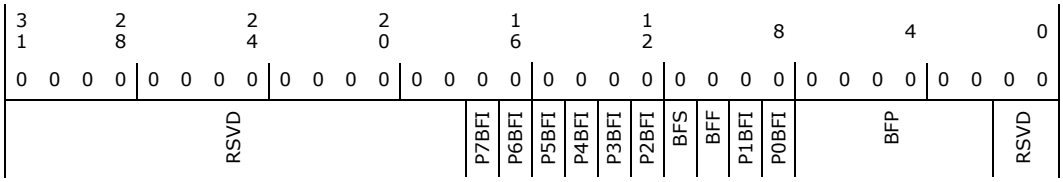
BIST FIS Control/Status

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:18	0h RO	Reserved.
17	0h RW	**Port 7 BIST FIS Initiate (P7BFI):** When a rising edge is detected on this bit, the SATA controller will initiate a BIST FIS to the device on port 7, using the parameters specified in this register and BFTD1 and BFTD2. The BIST FIS will only be initiated if a device is present and not in the partial or slumber states. After a BIST FIS is successfully completed, software must disable and re-enable the PCS.P7E prior to attempting additional BIST FISes or to return the SATA controller to a normal operational mode. If the BIST FIS fails, as indicated by BFF in this register, software can clear and then set this bit to initiate another BIST FIS. This bit is only applicable to project(s): That has port 7 physically.
16	0h RW	**Port 6 BIST FIS Initiate (P6BFI):** When a rising edge is detected on this bit, the SATA controller will initiate a BIST FIS to the device on port 6, using the parameters specified in this register and BFTD1 and BFTD2. The BIST FIS will only be initiated if a device is present and not in the partial or slumber states. After a BIST FIS is successfully completed, software must disable and re-enable the PCS.P6E prior to attempting additional BIST FISes or to return the SATA controller to a normal operational mode. If the BIST FIS fails, as indicated by BFF in this register, software can clear and then set this bit to initiate another BIST FIS. This bit is only applicable to project(s): That has port 6 physically.
15	0h RW	**Port 5 BIST FIS Initiate (P5BFI):** When a rising edge is detected on this bit, the SATA controller will initiate a BIST FIS to the device on port 5, using the parameters specified in this register and BFTD1 and BFTD2. The BIST FIS will only be initiated if a device is present and not in the partial or slumber states. After a BIST FIS is successfully completed, software must disable and re-enable the PCS.P5E prior to attempting additional BIST FISes or to return the SATA controller to a normal operational mode. If the BIST FIS fails, as indicated by BFF in this register, software can clear and then set this bit to initiate another BIST FIS. This bit is only applicable to project(s): That has port 5 physically.
14	0h RW	**Port 4 BIST FIS Initiate (P4BFI):** When a rising edge is detected on this bit, the SATA controller will initiate a BIST FIS to the device on port 4, using the parameters specified in this register and BFTD1 and BFTD2. The BIST FIS will only be initiated if a device is present and not in the partial or slumber states. After a BIST FIS is successfully completed, software must disable and re-enable PCS.P4E prior to attempting additional BIST FISes or to return the SATA controller to a normal operational mode. If the BIST FIS fails, as indicated by BFF in this register, software can clear and then set this bit to initiate another BIST FIS. This bit is only applicable to project(s): That has port 4 physically.
13	0h RW	**Port 3 BIST FIS Initiate (P3BFI):** When a rising edge is detected on this bit, the SATA controller will initiate a BIST FIS to the device on port 3, using the parameters specified in this register and BFTD1 and BFTD2. The BIST FIS will only be initiated if a device is present and not in the partial or slumber states. After a BIST FIS is successfully completed, software must disable and re-enable the PCS.P3E prior to attempting additional BIST FISes or to return the SATA controller to a normal operational mode. If the BIST FIS fails, as indicated by BFF in this register, software can clear and then set this bit to initiate another BIST FIS. This bit is only applicable to project(s): That has port 3 physically.

Bit Range	Default and Access	Field Name (ID): Description
12	0h RW	**Port 2 BIST FIS Initiate (P2BFI):** When a rising edge is detected on this bit, the SATA controller will initiate a BIST FIS to the device on port 2, using the parameters specified in this register and BFTD1 and BFTD2. The BIST FIS will only be initiated if a device is present and not in the partial or slumber states. After a BIST FIS is successfully completed, software must disable and re-enable PCS.P2E prior to attempting additional BIST FISes or to return the SATA controller to a normal operational mode. If the BIST FIS fails, as indicated by BFF in this register, software can clear and then set this bit to initiate another BIST FIS. This bit is only applicable to project(s): That has port 2 physically.
11	0h RW/1C	**BIST FIS Successful (BFS):** This bit is set any time a BIST FIS transmitted by the SATA controller receives an R_OK completion status from the device.
10	0h RW/1C	**BIST FIS Failed (BFF):** This bit is set any time that a BIST FIS transmitted by the SATA controller receives an R_ERR completion status from the device.
9	0h RW	**Port 1 BIST FIS Initiate (P1BFI):** When a rising edge is detected on this bit, the SATA controller will initiate a BIST FIS to the device on port 1, using the parameters specified in this register and BFTD1 and BFTD2. The BIST FIS will only be initiated if a device is present and not in the partial or slumber states. After a BIST FIS is successfully completed, software must disable and re-enable PCS.P1E prior to attempting additional BIST FISes or to return the SATA controller to a normal operational mode. If the BIST FIS fails, as indicated by BFF in this register, software can clear and then set this bit to initiate another BIST FIS. This bit is only applicable to project(s): That has port 1 physically.
8	0h RW	**Port 0 BIST FIS Initiate (P0BFI):** When a rising edge is detected on this bit, the SATA controller will initiate a BIST FIS to the device on port 0, using the parameters specified in this register and BFTD1 and BFTD2. The BIST FIS will only be initiated if a device is present and not in the partial or slumber states. After a BIST FIS is successfully completed, software must disable and re-enable PCS.P0E prior to attempting additional BIST FISes or to return the SATA controller to a normal operational mode. If the BIST FIS fails, as indicated by BFF in this register, software can clear and then set this bit to initiate another BIST FIS. This bit is only applicable to project(s): That has port 0 physically.
7:2	0h RW	**BIST FIS Parameters (BFP):** These bits form the contents of the upper 6 bits of the BIST FIS Pattern Definition in the BIST FIS transmitted by the SATA controller. This field is not port specific - its contents will be used for any BIST FIS initiated on the SATA controller. The specific bit definitions are: Bit 7 (T) Far End Transmit mode; bit 6 (A) Align Bypass mode; bit 5 (S) Bypass Scrambling; bit 4 (L) Far End Retimed Loopback; bit 3 (F) Far End Analog Loopback; bit 2 (P) Primitive bit for use with Transmit mode.
1:0	0h RO	Reserved.

24.8.1.37 BIST FIS Transmit Data 1 (BFTD1)—Offset E4h

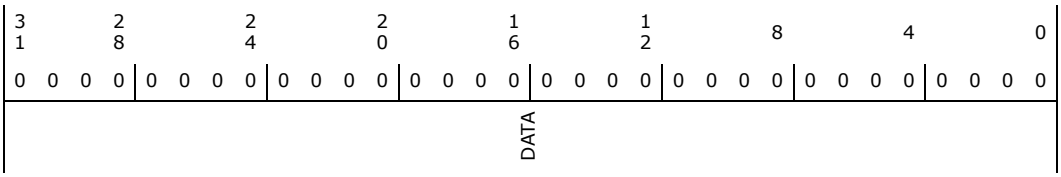
BIST FIS Transmit Data 1

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Data (DATA):** The data programmed into this register will form the contents of the second DW of any BIST FIS initiated by the SATA controller. This register is not port specific - its contents will be used for BIST FIS initiated on any port. Although the 2nd and 3rd DWs of the BIST FIS are only meaningful when the T bit of the BIST FIS is set to indicate Far-End Transmit mode, this register's contents will be transmitted as the BIST FIS 2nd DW regardless of whether or not the T bit is indicated in the BFCS register.

24.8.1.38 BIST FIS Transmit Data 2 (BFTD2)—Offset E8h

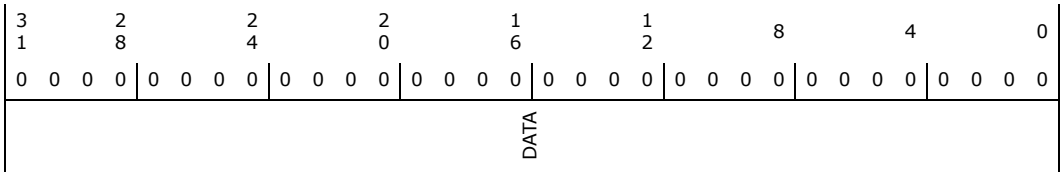
BIST FIS Transmit Data 2

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Data (DATA):** The data programmed into this register will form the contents of the third DW of any BIST FIS initiated by the SATA controller. This register is not port specific - its contents will be used for BIST FIS initiated on any port. Although the 2nd and 3rd DWs of the BIST FIS are only meaningful when the T bit of the BIST FIS is set to indicate Far-End Transmit mode, this register's contents will be transmitted as the BIST FIS 3rd DW regardless of whether or not the T bit is set in the BFCS register.

24.8.2 SATA ABAR Registers Summary

Table 24-2. Summary of SATA ABAR Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	HBA Capabilities (GHC_CAP)—Offset 0h	FF36FF07h
4h	7h	Global HBA Control (GHC)—Offset 4h	80000000h
8h	Bh	Interrupt Status Register (IS)—Offset 8h	0h
Ch	Fh	Ports Implemented (GHC_PI)—Offset Ch	0h
10h	13h	AHCI Version (VS)—Offset 10h	10300h
1Ch	1Fh	Enclosure Management Location (EM_LOC)—Offset 1Ch	1600002h
20h	23h	Enclosure Management Control (EM_CTL)—Offset 20h	7010000h
24h	27h	HBA Capabilities Extended (GHC_CAP2)—Offset 24h	3Ch
A0h	A3h	Vendor Specific (VSP)—Offset A0h	48h
A4h	A7h	Vendor Specific Capabilities Register (VS_CAP)—Offset A4h	1002DEh
C0h	C3h	RAID Platform ID (RPID)—Offset C0h	311C02h

Table 24-2. Summary of SATA ABAR Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
C4h	C5h	Premium Feature Block (PFB)—Offset C4h	0h
C8h	C9h	SW Feature Mask (SFM)—Offset C8h	3Fh
100h	103h	Port 0 Command List Base Address (P0CLB)—Offset 100h	0h
104h	107h	Port 0 Command List Base Address Upper 32-bits (P0CLBU)—Offset 104h	0h
108h	10Bh	Port 0 FIS Base Address (P0FB)—Offset 108h	0h
10Ch	10Fh	Port 0 FIS Base Address Upper 32-Bits (P0FBU)—Offset 10Ch	0h
110h	113h	Port 0 Interrupt Status (P0IS)—Offset 110h	0h
114h	117h	Port 0 Interrupt Enable (P0IE)—Offset 114h	0h
118h	11Bh	Port 0 Command (P0CMD)—Offset 118h	4h
120h	123h	Port 0 Task File Data (P0TFD)—Offset 120h	9h
124h	127h	Port 0 Signature (P0SIG)—Offset 124h	FFFFFFFFh
128h	12Bh	Port 0 Serial ATA Status (P0SSTS)—Offset 128h	0h
12Ch	12Fh	Port 0 Serial ATA Control (P0SCTL)—Offset 12Ch	0h
130h	133h	Port 0 Serial ATA Error (P0SERR)—Offset 130h	0h
134h	137h	Port 0 Serial ATA Active (P0SACT)—Offset 134h	0h
138h	13Bh	Port 0 Command Issue (P0CI)—Offset 138h	0h
13Ch	13Fh	Port 0 SNotification (P0SNTF)—Offset 13Ch	0h
144h	147h	Port 0 Device Sleep (P0DEVSLP)—Offset 144h	1E022852h
180h	183h	Port 1 Command List Base Address (P1CLB)—Offset 180h	0h
184h	187h	Port 1 Command List Base Address Upper 32-bits (P1CLBU)—Offset 184h	0h
188h	18Bh	Port 1 FIS Base Address (P1FB)—Offset 188h	0h
18Ch	18Fh	Port 1 FIS Base Address Upper 32-bits (P1FBU)—Offset 18Ch	0h
190h	193h	Port 1 Interrupt Status (P1IS)—Offset 190h	0h
194h	197h	Port 1 Interrupt Enable (P1IE)—Offset 194h	0h
198h	19Bh	Port 1 Command (P1CMD)—Offset 198h	0h
1A0h	1A3h	Port 1 Task File Data (P1TFD)—Offset 1A0h	0h
1A4h	1A7h	Port 1 Signature (P1SIG)—Offset 1A4h	0h
1A8h	1ABh	Port 1 Serial ATA Status (P1SSTS)—Offset 1A8h	0h
1ACh	1AFh	Port 1 Serial ATA Control (P1SCTL)—Offset 1ACh	0h
1B0h	1B3h	Port 1 Serial ATA Error (P1SERR)—Offset 1B0h	0h
1B4h	1B7h	Port 1 Serial ATA Active (P1SACT)—Offset 1B4h	0h
1B8h	1BBh	Port 1 Command Issue (P1CI)—Offset 1B8h	0h
1BCh	1BFh	Port 1 SNotification (P1SNTF)—Offset 1BCh	0h
1C4h	1C7h	Port 1 Device Sleep (P1DEVSLP)—Offset 1C4h	0h
200h	203h	Port 2 Command List Base Address (P2CLB)—Offset 200h	0h
204h	207h	Port 2 Command List Base Address Upper 32-bits (P2CLBU)—Offset 204h	0h
208h	20Bh	Port 2 FIS Base Address (P2FB)—Offset 208h	0h
20Ch	20Fh	Port 2 FIS Base Address Upper 32-bits (P2FBU)—Offset 20Ch	0h
210h	213h	Port 2 Interrupt Status (P2IS)—Offset 210h	0h
214h	217h	Port 2 Interrupt Enable (P2IE)—Offset 214h	0h
218h	21Bh	Port 2 Command (P2CMD)—Offset 218h	0h

Table 24-2. Summary of SATA ABAR Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
220h	223h	Port 2 Task File Data (P2TFD)—Offset 220h	0h
224h	227h	Port 2 Signature (P2SIG)—Offset 224h	0h
228h	22Bh	Port 2 Serial ATA Status (P2SSTS)—Offset 228h	0h
22Ch	22Fh	Port 2 Serial ATA Control (P2SCTL)—Offset 22Ch	0h
230h	233h	Port 2 Serial ATA Error (P2SERR)—Offset 230h	0h
234h	237h	Port 2 Serial ATA Active (P2SACT)—Offset 234h	0h
238h	23Bh	Port 2 Command Issue (P2CI)—Offset 238h	0h
23Ch	23Fh	Port 2 SNotification (P2SNTF)—Offset 23Ch	0h
244h	247h	Port 2 Device Sleep (P2DEVSLP)—Offset 244h	0h
280h	283h	Port 3 Command List Base Address (P3CLB)—Offset 280h	0h
284h	287h	Port 3 Command List Base Address Upper 32-bits (P3CLBU)—Offset 284h	0h
288h	28Bh	Port 3 FIS Base Address (P3FB)—Offset 288h	0h
28Ch	28Fh	Port 3 FIS Base Address Upper 32-bits (P3FBU)—Offset 28Ch	0h
290h	293h	Port 3 Interrupt Status (P3IS)—Offset 290h	0h
294h	297h	Port 3 Interrupt Enable (P3IE)—Offset 294h	0h
298h	29Bh	Port 3 Command (P3CMD)—Offset 298h	0h
2A0h	2A3h	Port 3 Task File Data (P3TFD)—Offset 2A0h	0h
2A4h	2A7h	Port 3 Signature (P3SIG)—Offset 2A4h	0h
2A8h	2ABh	Port 3 Serial ATA Status (P3SSTS)—Offset 2A8h	0h
2ACh	2AFh	Port 3 Serial ATA Control (P3SCTL)—Offset 2ACh	0h
2B0h	2B3h	Port 3 Serial ATA Error (P3SERR)—Offset 2B0h	0h
2B4h	2B7h	Port 3 Serial ATA Active (P3SACT)—Offset 2B4h	0h
2B8h	2BBh	Port 3 Commands Issued (P3CI)—Offset 2B8h	0h
2BCh	2BFh	Port 3 SNotification (P3SNTF)—Offset 2BCh	0h
2C4h	2C7h	Port 3 Device Sleep (P3DEVSLP)—Offset 2C4h	0h
300h	303h	Port 4 Command List Base Address (P4CLB)—Offset 300h	0h
304h	307h	Port 4 Command List Base Address Upper 32-bits (P4CLBU)—Offset 304h	0h
308h	30Bh	Port 4 FIS Base Address (P4FB)—Offset 308h	0h
30Ch	30Fh	Port 4 FIS Base Address Upper 32-bits (P4FBU)—Offset 30Ch	0h
310h	313h	Port 4 Interrupt Status (P4IS)—Offset 310h	0h
314h	317h	Port 4 Interrupt Enable (P4IE)—Offset 314h	0h
318h	31Bh	Port 4 Command (P4CMD)—Offset 318h	0h
320h	323h	Port 4 Task File Data (P4TFD)—Offset 320h	0h
324h	327h	Port 4 Signature (P4SIG)—Offset 324h	0h
328h	32Bh	Port 4 Serial ATA Status (P4SSTS)—Offset 328h	0h
32Ch	32Fh	Port 4 Serial ATA Control (P4SCTL)—Offset 32Ch	0h
330h	333h	Port 4 Serial ATA Error (P4SERR)—Offset 330h	0h
334h	337h	Port 4 Serial ATA Active (P4SACT)—Offset 334h	0h
338h	33Bh	Port 4 Commands Issued (P4CI)—Offset 338h	0h
33Ch	33Fh	Port 4 SNotification (P4SNTF)—Offset 33Ch	0h
344h	347h	Port 4 Device Sleep (P4DEVSLP)—Offset 344h	0h

Table 24-2. Summary of SATA ABAR Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
380h	383h	Port 5 Command List Base Address (P5CLB)—Offset 380h	0h
384h	387h	Port 5 Command List Base Address Upper 32-bits (P5CLBU)—Offset 384h	0h
388h	38Bh	Port 5 FIS Base Address (P5FB)—Offset 388h	0h
38Ch	38Fh	Port 5 FIS Base Address Upper 32-bits (P5FBU)—Offset 38Ch	0h
390h	393h	Port 5 Interrupt Status (P5IS)—Offset 390h	0h
394h	397h	Port 5 Interrupt Enable (P5IE)—Offset 394h	0h
398h	39Bh	Port 5 Command (P5CMD)—Offset 398h	0h
3A0h	3A3h	Port 5 Task File Data (P5TFD)—Offset 3A0h	0h
3A4h	3A7h	Port 5 Signature (P5SIG)—Offset 3A4h	0h
3A8h	3ABh	Port 5 Serial ATA Status (P5SSTS)—Offset 3A8h	0h
3ACh	3AFh	Port 5 Serial ATA Control (P5SCTL)—Offset 3ACh	0h
3B0h	3B3h	Port 5 Serial ATA Error (P5SERR)—Offset 3B0h	0h
3B4h	3B7h	Port 5 Serial ATA Active (P5SACT)—Offset 3B4h	0h
3B8h	3BBh	Port 5 Commands Issued (P5CI)—Offset 3B8h	0h
3BCh	3BFh	Port 5 SNotification (P5SNTF)—Offset 3BCh	0h
3C4h	3C7h	Port 5 Device Sleep (P5DEVSLP)—Offset 3C4h	0h
400h	403h	Port 6 Command List Base Address (P6CLB)—Offset 400h	0h
404h	407h	Port 6 Command List Base Address Upper 32-bits (P6CLBU)—Offset 404h	0h
408h	40Bh	Port 6 FIS Base Address (P6FB)—Offset 408h	0h
40Ch	40Fh	Port 6 FIS Base Address Upper 32-bits (P6FBU)—Offset 40Ch	0h
410h	413h	Port 6 Interrupt Status (P6IS)—Offset 410h	0h
414h	417h	Port 6 Interrupt Enable (P6IE)—Offset 414h	0h
418h	41Bh	Port 6 Command (P6CMD)—Offset 418h	0h
420h	423h	Port 6 Task File Data (P6TFD)—Offset 420h	0h
424h	427h	Port 6 Signature (P6SIG)—Offset 424h	0h
428h	42Bh	Port 6 Serial ATA Status (P6SSTS)—Offset 428h	0h
42Ch	42Fh	Port 6 Serial ATA Control (P6SCTL)—Offset 42Ch	0h
430h	433h	Port 6 Serial ATA Error (P6SERR)—Offset 430h	0h
434h	437h	Port 6 Serial ATA Active (P6SACT)—Offset 434h	0h
438h	43Bh	Port 6 Commands Issued (P6CI)—Offset 438h	0h
43Ch	43Fh	Port 6 SNotification (P6SNTF)—Offset 43Ch	0h
444h	447h	Port 6 Device Sleep (P6DEVSLP)—Offset 444h	0h
480h	483h	Port 7 Command List Base Address (P7CLB)—Offset 480h	0h
484h	487h	Port 7 Command List Base Address Upper 32-bits (P7CLBU)—Offset 484h	0h
488h	48Bh	Port 7 FIS Base Address (P7FB)—Offset 488h	0h
48Ch	48Fh	Port 7 FIS Base Address Upper 32-bits (P7FBU)—Offset 48Ch	0h
490h	493h	Port 7 Interrupt Status (P7IS)—Offset 490h	0h
494h	497h	Port 7 Interrupt Enable (P7IE)—Offset 494h	0h
498h	49Bh	Port 7 Command (P7CMD)—Offset 498h	0h
4A0h	4A3h	Port 7 Task File Data (P7TFD)—Offset 4A0h	0h
4A4h	4A7h	Port 7 Signature (P7SIG)—Offset 4A4h	0h

Table 24-2. Summary of SATA ABAR Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
4A8h	4ABh	Port 7 Serial ATA Status (P7SSTS)—Offset 4A8h	0h
4ACh	4AFh	Port 7 Serial ATA Control (P7SCTL)—Offset 4ACh	0h
4B0h	4B3h	Port 7 Serial ATA Error (P7SERR)—Offset 4B0h	0h
4B4h	4B7h	Port 7 Serial ATA Active (P7SACT)—Offset 4B4h	0h
4B8h	4BBh	Port 7 Commands Issued (P7CI)—Offset 4B8h	0h
4BCh	4BFh	Port 7 SNotification (P7SNTF)—Offset 4BCh	0h
4C4h	4C7h	Port 7 Device Sleep (P7DEVSLP)—Offset 4C4h	0h
580h	583h	Enclosure Management Message Format (EM_MF)—Offset 580h	0h
584h	587h	Enclosure Management LED (EM_LED)—Offset 584h	0h

24.8.2.1 HBA Capabilities (GHC_CAP)—Offset 0h

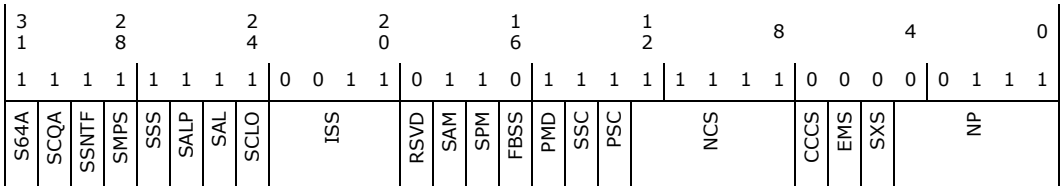
HBA Capabilities. This register indicates basic capabilities of the HBA to driver software. The RWO bits in this register are only cleared upon PLTRST#. This register is not reset by FLR.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: FF36FF07h



Bit Range	Default and Access	Field Name (ID): Description
31	1h RW/O	**Supports 64-bit Addressing (S64A):** Indicates the S-ATA controller can access 64-bit data structures. The 32-bit upper bits of the port DMA Descriptor, the PRD Base, and each PRD entry are read/write.
30	1h RW/O	**Supports Native Command Queuing Acceleration (SCQA):** Indicates the SATA controller supports Serial-ATA Native Command Queueing. The HBA will handle DMA Setup FISes in hardware, including support for auto-activate optimization through the FIS.
29	1h RW/O	**Supports SNotification Register (SSNTF):** When set to 1, indicates that the HBA supports the PxSNTF (SNotification) register and its associated functionality. When cleared to 0., the HBA does not support the PxSNTF (SNotification) register and its associated functionality.
28	1h RW/O	**Supports Mechanical Presence Switch (SMPS):** When set to 1, the HBA supports mechanical presence switches on its ports for use in hot plug operations. When cleared to 0, this function is not supported. This value is loaded by the BIOS prior to OS initialization.
27	1h RW/O	**Supports Staggered Spin-up (SSS):** Indicates whether the S-ATA controller supports staggered spin-up on its ports, for use in balancing power spikes. This value is loaded by platform BIOS prior to OS initialization.

Bit Range	Default and Access	Field Name (ID): Description
26	1h RW/O	**Supports Aggressive Link Power Management (SALP):** Indicates the S-ATA controller supports auto-generating link requests to the partial or slumber states when there are no commands to process. When cleared to 0, software shall treat the PxCMD.ALPE and PxCMD.ASP bits as reserved.
25	1h RW/O	**Supports Activity LED (SAL):** Indicates the S-ATA controller supports a single output pin (SATALED#) which indicates activity.
24	1h RW/O	**Supports Command List Override (SCLO):** When set to 1, indicates that the HBA supports the PxCMD.CLO bit and it's associated function. When cleared to 0., The HBA is not capable of clearing the BSY and DRQ bits in the Status register in order to issue a software reset if these bits are still set from a previous operation.
23:20	3h RW/O	**Interface Speed Support (ISS):** Indicates the maximum speed the S-ATA controller can support on its ports. These encodings match the system software programmable PxSCTL.DET.SPD field. 0000 = Reserved 0001 = Gen 1 (1.5 Gbps) 0010 = Gen 2 (3 Gbps) 0011 = Gen 3 (6 Gbps) 0100 - 1111 = Reserved. ***Note:*** If (FFSATA0p0, FFSATA0p1, FFSATA0p2, FFSATA0p3, FFSATA0p4 and FFSATA0p5) is 1, this field is RWO defaulting to 0010 and ignores software write value of 0011. If either FFSATA0p0, FFSATA0p1, FFSATA0p2, FFSATA0p3, FFSATA0p4 or FFSATA0p5 is 0, this field is RWO defaulting to 0011.
19	0h RO	Reserved.
18	1h RO	**Supports AHCI mode only (SAM):** The SATA controller may optionally support AHCI access mechanism only. A value of 0 indicates that in addition to the native AHCI mechanism (via ABAR), the SATA controller implements a legacy, task-file based register interface such as SFF-8038i. A value of 1 indicates that the SATA controller does not implement a legacy, task-file based register interface.
17	1h RO	**Supports Port Multiplier (SPM):** The SATA controller may optionally support command-based switching Port Multipliers. BIOS must clear this bit if Port Multipliers are not supported.
16	0h RO	**FIS-based Switching Supported (FBSS):** Not supported.
15	1h RO	**PIO Multiple DRQ Block (PMD):** If set to 1, the HBA supports multiple DRQ block data transfers for the PIO command protocol.
14	1h RW/O	**Slumber State Capable (SSC):** The SATA controller supports the slumber state.
13	1h RW/O	**Partial State Capable (PSC):** The SATA controller supports the partial state.
12:8	1Fh RO	**Number of Command Slots (NCS):** 1Fh indicating support for 32 slots.
7	0h RO	**Command Completion Coalescing Supported (CCCS):** When set to 1, indicates that the HBA supports command completion coalescing. When command completion coalescing is supported, the HBA has implemented the CCC_CTL and the CCC_PORTS global HBA registers. When cleared to 0, indicates that the HBA does not support command completion coalescing and the CCC_CTL and CCC_PORTS global HBA registers are not implemented.

Bit Range	Default and Access	Field Name (ID): Description
6	0h RO	**Enclosure Management Supported (EMS):** When set to 1, indicates that the HBA supports enclosure management. When enclosure management is supported, the HBA has implemented the EM_LOC and EM_CTL global HBA registers. When cleared to 0, indicates that the HBA does not support enclosure management and the EM_LOC and EM_CTL global HBA registers are not implemented.
5	0h RW/O	**Supports External SATA (SXS):** When set to 1, indicates that the HBA has one or more Serial ATA ports that has a signal only connector that is externally accessible. If this bit is set, software may refer to the PxCMD.ESP bit to determine whether a specific port has its signal connector externally accessible as a signal only connector (i.e., power is not part of that connector). When the bit is cleared to 0, indicates that the HBA has no Serial ATA ports that have a signal only connector externally accessible.
4:0	7h RO	**Number of Ports (NP):** 0's based value indicating the maximum number of ports supported. Note that the number of ports indicated in this field may be more than the number of ports indicated in the PI register. Number of ports shall be dependent on MAP.SC, fuses (FFSATA7, FFSATA8) and PCIe/SATA muxing configuration where if ANY of these parameter disable a particular port then that port is disabled and not counted. The maximum number of ports supported by SIP is 8 and the least is 0 (i.e., Function Disable). In the case of 0 port configuration, the value of NP is a don't care (while implementation has it fixed as 07h). Any combination in between is supported by SATA host controller. Indicates the number of supported ports.

24.8.2.2 Global HBA Control (GHC)—Offset 4h

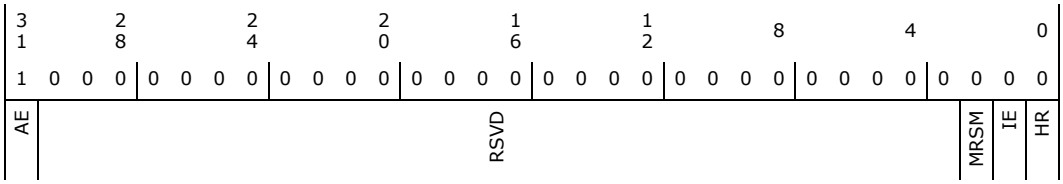
Global HBA Control. This register controls various global actions of the HBA.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 80000000h



Bit Range	Default and Access	Field Name (ID): Description
31	1h RO	**AHCI Enable (AE):** When set, indicates that an AHCI driver is loaded and communication to the HBA shall be via AHCI mechanisms. This can be used by an HBA that supports both legacy mechanisms (such as SFF-8038i) and AHCI to know when the HBA is running under an AHCI driver. When set, software shall only talk to the HBA using AHCI. The HBA will not have to allow command processing via both AHCI and legacy mechanisms. When cleared, software will only communicate with the HBA using legacy mechanisms. Software shall set this bit to 1 before accessing other AHCI registers. ***Note:*** The implementation of this bit is dependent upon the value of the CAP.SAM bit. If CAP.SAM is 0, then GHC.AE should be RW and shall have a reset value of 0. If CAP.SAM is 1, then AE shall be read only and shall have a reset value of 1.
30:3	0h RO	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
2	0h RO	**MSI Revert to Single Message (MRSM):** When set to 1 by hardware, indicates that the HBA requested more than one MSI vector but has reverted to using the first vector only. When this bit is cleared to 0, the HBA has not reverted to single MSI mode (i.e., hardware is already in single MSI mode, software has allocated the number of messages requested, or hardware is sharing interrupt vectors if MC.MME < MC.MMC). The HBA may revert to single MSI mode when the number of vectors allocated by the host is less than the number requested. This bit shall only be set to 1 when the following conditions hold: MC.MSIE = 1 (MSI is enabled); MC.MMC > 0 (multiple messages requested); MC.MME > 0 (more than one message allocated); MC.MME != MC.MMC (messages allocated not equal to number requested). When this bit is set to 1, single MSI mode operation is in use and software is responsible for clearing bits in the IS register to clear interrupts. This bit shall be cleared to 0 by hardware when any of the four conditions stated is false. This bit is also cleared to 0 when MC.MSIE = 1 and MC.MME = 0h. In this case, the hardware has been programmed to use single MSI mode, and is not reverting to that mode. The HBA shall always revert to single MSI mode when the number of vectors allocated by the host is less than the number requested. Value of MRSM is a don't care when GHC.HR=1.
1	0h RW	**Interrupt Enable (IE):** This global bit enables interrupts from the HBA. When cleared (reset default), all interrupt sources from all ports are disabled. When set, interrupts are enabled.
0	0h RW/1S	**HBA Reset (HR):** When set by SW, this bit causes an internal reset of the HBA. All state machines that relate to data transfers and native command queuing will return to an idle condition, and all ports will be re-initialized via COMRESET. When the HBA has performed the reset action, it will reset this bit to 0. A software write of 0 will have no effect. For a description on which bits are reset when this bit is set, see the *AHCI Specification*, Section 10.3.3.

24.8.2.3 Interrupt Status Register (IS)—Offset 8h

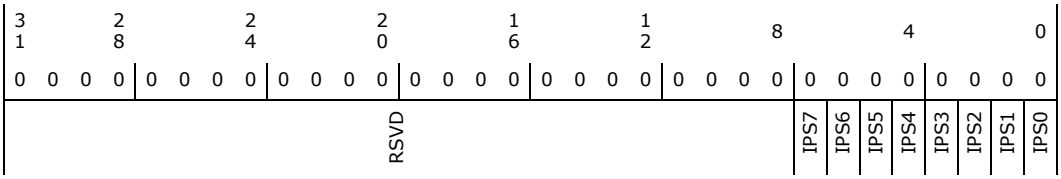
Interrupt Status Register. This register indicates which of the ports within the controller have an interrupt pending and require service.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:8	0h RO	Reserved.
7	0h RW/1C	**Interrupt Pending Status Port 7 (IPS7):** If set, indicates that port 7 has an interrupt pending. Software can use this information to determine which ports require service after an interrupt. This bit is only applicable to project(s): That has port 7 physically.
6	0h RW/1C	**Interrupt Pending Status Port 6 (IPS6):** If set, indicates that port 6 has an interrupt pending. Software can use this information to determine which ports require service after an interrupt. This bit is only applicable to project(s): That has port 6 physically.

Bit Range	Default and Access	Field Name (ID): Description
5	0h RW/1C	**Interrupt Pending Status Port 5 (IPS5):** If set, indicates that port 5 has an interrupt pending. Software can use this information to determine which ports require service after an interrupt. This bit is only applicable to project(s): That has port 5 physically.
4	0h RW/1C	**Interrupt Pending Status Port 4 (IPS4):** If set, indicates that port 4 has an interrupt pending. Software can use this information to determine which ports require service after an interrupt. This bit is only applicable to project(s): That has port 4 physically.
3	0h RW/1C	**Interrupt Pending Status Port 3 (IPS3):** If set, indicates that port 3 has an interrupt pending. Software can use this information to determine which ports require service after an interrupt. This bit is only applicable to project(s): That has port 3 physically.
2	0h RW/1C	**Interrupt Pending Status Port 2 (IPS2):** If set, indicates that port 2 has an interrupt pending. Software can use this information to determine which ports require service after an interrupt. This bit is only applicable to project(s): That has port 2 physically.
1	0h RW/1C	**Interrupt Pending Status Port 1 (IPS1):** If set, indicates that port 1 has an interrupt pending. Software can use this information to determine which ports require service after an interrupt. This bit is only applicable to project(s): That has port 1 physically.
0	0h RW/1C	**Interrupt Pending Status Port 0 (IPS0):** If set, indicates that port 0 has an interrupt pending. Software can use this information to determine which ports require service after an interrupt. This bit is only applicable to project(s): That has port 0 physically.

24.8.2.4 Ports Implemented (GHC_PI)—Offset Ch

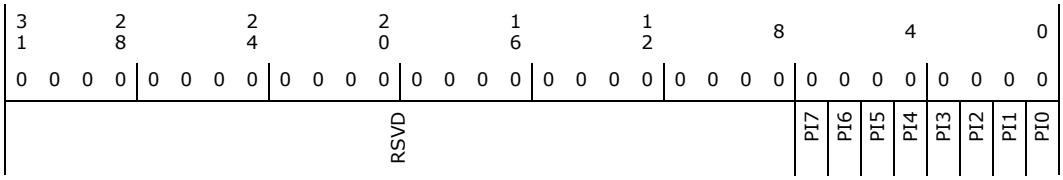
Ports Implemented. This register indicates which ports are exposed to the HBA. It is loaded by platform BIOS. It indicates which ports that the device supports are available for software to use. Any available port may not be implemented. This register is not reset by FLR.

Access Method

Type: MEM Register (Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:8	0h RO	Reserved.
7	0h RW/O	**Port 7 Implemented (PI7):** If set, then port 7 is available for use. If cleared, the port is not available for use. This bit is reserved and is read-only 0 if below condition is true else it's RWO-zero: Refer to CAP.NP where port 7 is not available.
6	0h RW/O	**Port 6 Implemented (PI6):** If set, then port 6 is available for use. If cleared, the port is not available for use. This bit is reserved and is read-only 0 if below condition is true else it's RWO-zero: Refer to CAP.NP where port 6 is not available.

Bit Range	Default and Access	Field Name (ID): Description
5	0h RW/O	**Port 5 Implemented (PI5):** If set, then port 5 is available for use. If cleared, the port is not available for use. This bit is reserved and is read-only 0 if below condition is true else it's RWO-zero: Refer to CAP.NP where port 5 is not available.
4	0h RW/O	**Port 4 Implemented (PI4):** If set, then port 4 is available for use. If cleared, the port is not available for use. This bit is reserved and is read-only 0 if below condition is true else it's RWO-zero: Refer to CAP.NP where port 4 is not available.
3	0h RW/O	**Port 3 Implemented (PI3):** If set, then port 3 is available for use. If cleared, the port is not available for use. This bit is reserved and is read-only 0 if below condition is true else it's RWO-zero: Refer to CAP.NP where port 3 is not available.
2	0h RW/O	**Port 2 Implemented (PI2):** If set, then port 2 is available for use. If cleared, the port is not available for use. This bit is reserved and is read-only 0 if below condition is true else it's RWO-zero: Refer to CAP.NP where port 2 is not available.
1	0h RW/O	**Port 1 Implemented (PI1):** If set, then port 1 is available for use. If cleared, the port is not available for use. This bit is reserved and is read-only 0 if below condition is true else it's RWO-zero: Refer to CAP.NP where port 1 is not available.
0	0h RW/O	**Port 0 Implemented (PI0):** If set, then port 0 is available for use. If cleared, the port is not available for use. This bit is reserved and is read-only 0 if below condition is true else it's RWO-zero: Refer to CAP.NP where port 0 is not available.

24.8.2.5 AHCI Version (VS)—Offset 10h

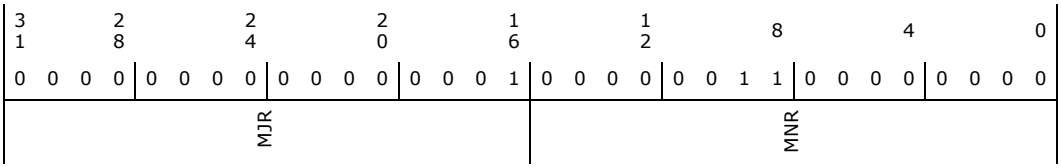
AHCI Version. This register indicates the major and minor version of the AHCI specification. It is BCD encoded. The upper two bytes represent the major version number, and the lower two bytes represent the minor version number. Example: Version 3.12 would be represented as 00030102h.

Access Method

Type: MEM Register (Size: 32 bits)

Device: 23
Function: 0

Default: 10300h



Bit Range	Default and Access	Field Name (ID): Description
31:16	1h RO	**Major Version Number (MJR):** Indicates the major version is 1
15:0	300h RO	**Minor Version Number (MNR):** Indicates the minor version is 30

24.8.2.6 Enclosure Management Location (EM_LOC)—Offset 1Ch

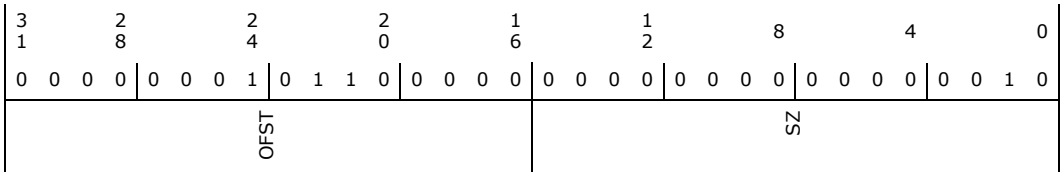
Enclosure Management Location. The enclosure management location register identifies the location and size of the enclosure management message buffer. This register is not implemented if enclosure management is not supported (i.e., CAP.EMS = 0).

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 1600002h



Bit Range	Default and Access	Field Name (ID): Description
31:16	160h RO	**Offset (OFST):** The offset of the message buffer in Dwords from the beginning of the ABAR.
15:0	2h RO	**Buffer Size (SZ):** Specifies the size of the transmit message buffer area in Dwords. If both transmit and receive buffers are supported, then the transmit buffer begins at ABAR[EM_LOC.OFST*4] and the receive buffer directly follows it. If both transmit and receive buffers are supported, both buffers are of the size indicated in the Buffer Size field. A value of 0 is invalid. Note that SATA controller only supports transmit buffer.

24.8.2.7 Enclosure Management Control (EM_CTL)—Offset 20h

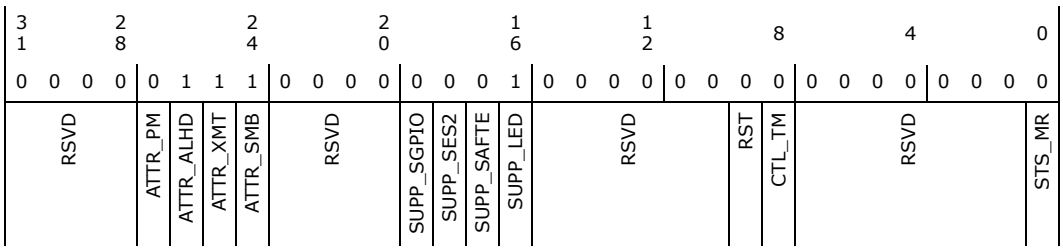
Enclosure Management Control. This register is used to control and obtain status for the enclosure management interface. The register includes information on the attributes of the implementation, enclosure management messages supported, the status of the interface, whether any messages are pending, and is used to initiate sending messages. This register is not implemented if enclosure management is not supported (i.e., CAP.EMS = 0).

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 7010000h



Bit Range	Default and Access	Field Name (ID): Description
31:28	0h RO	Reserved.
27	0h RO	**Port Multiplier Support (ATTR_PM):** The HBA does not support enclosure management messages for devices attached via a Port Multiplier. Software should use the Serial ATA enclosure management bridge that is built into many Port Multipliers for enclosure services with these devices. For more information on Serial ATA enclosure management bridges, refer to the Serial ATA II: Extensions to Serial ATA 1.0a revision 1.2 specification.
26	1h RW/O	**Activity LED Hardware Driven (ATTR_ALHD):** If set to 1, the HBA drives the activity LED for the LED message type in hardware and does not utilize software settings for this LED. The HBA does not begin transmitting the hardware based activity signal until after software has written CTL.TM=1 after a reset condition.
25	1h RO	**Transmit Only (ATTR_XMT):** If set to 1, the HBA only supports transmitting messages and does not support receiving messages. If cleared to 0, the HBA supports transmitting and receiving messages.
24	1h RO	**Single Message Buffer (ATTR_SMB):** If set to 1, the HBA has one message buffer that is shared for messages to transmit and messages received. In this case, unsolicited receive messages are not supported and it is software's responsibility to manage access to this buffer. If cleared to 0, there are separate receive and transmit buffers such that unsolicited messages could be supported.
23:20	0h RO	Reserved.
19	0h RO	**SGPIO Enclosure Management Messages (SUPP_SGPIO):** If set to 1, the HBA supports the SGPIO register interface message type.
18	0h RO	**SES-2 Enclosure Management Messages (SUPP_SES2):** If set to 1, the HBA supports the SES-2 message type.
17	0h RO	**SAF-TE Enclosure Management Messages (SUPP_SAFTE):** If set to 1, the HBA supports the SAF-TE message type.
16	1h RO	**LED Message Types (SUPP_LED):** If set to 1, the HBA supports the LED message type defined in LED Message Type.
15:10	0h RO	Reserved.
9	0h RW/1S	**Reset (RST):** When set to 1 by software, the HBA shall reset all enclosure management message logic and take all appropriate reset actions to ensure messages can be transmitted/received after the reset. After the HBA completes the reset operation, the HBA shall set the value to 0. A write of 0 by software to this field shall have no effect.
8	0h RW/1S	**Transmit Message (CTL_TM):** When set to 1 by software, the HBA shall transmit the message contained in the message buffer. When the message is completely sent, the HBA shall clear this bit to 0. A write of 0 to this bit by software shall have no effect. Software shall not change the contents of the message buffer while CTL.TM is set to 1.
7:1	0h RO	Reserved.
0	0h RO	**Message Received (STS_MR):** Message received is not supported.

24.8.2.8 HBA Capabilities Extended (GHC_CAP2)—Offset 24h

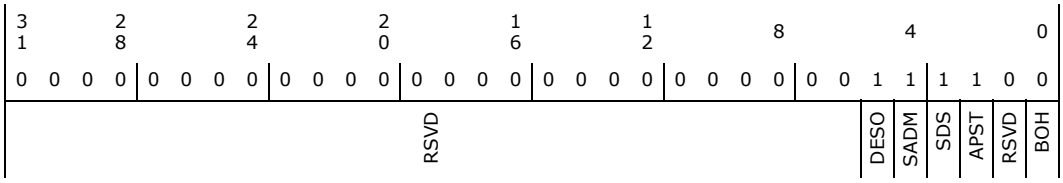
HBA Capabilities Extended. This register indicates basic capabilities of the HBA to driver software. The RWO bits in this register are only cleared upon PLTRST#. This register is not reset by FLR.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 3Ch



Bit Range	Default and Access	Field Name (ID): Description
31:6	0h RO	Reserved.
5	1h RW/O	**DEVSLP Entrance from Slumber Only (DESO):** This field specifies that the HBA shall only assert DEVSLP if the interface is in Slumber. When this bit is set to 1, the HBA shall ignore software directed entrance to DEVSLP via PxCMD.ICC unless PxSSTS.IPM = 6h. When this bit is cleared to 0, the HBA may enter DEVSLP from any link state (active, Partial, or Slumber). BIOS is required to program this field to 1.
4	1h RW/O	**Supports Aggressive DEVSLP Management (SADM):** When set to 1, the HBA supports hardware assertion of the DEVSLP signal after the idle timeout expires. When cleared to 0, this function is not supported and software shall treat the PxDEVSLP.ADSE field as reserved. ***Note:*** If PHY I/O PM Disable Fuse is 1, this register will read 0. Else this register will read 1 with RWO attribute.
3	1h RW/O	**Supports DEVSLP (SDS):** When set to 1, the HBA supports the DEVSLP feature. When cleared to 0, DEVSLP is not supported. ***Note:*** If PHY I/O PM Disable Fuse is 1, this register will read 0. Else this register will read 1 with RWO attribute.
2	1h RW/O	**Automatic Partial to Slumber Transitions (APST):** When set to 1, the HBA supports Automatic Partial to Slumber Transitions. When cleared to 0, Automatic Partial to Slumber Transition is not supported. ***Note:*** If SATA PHY PM Disable Fuse is 1, this register will read only 0. Else this register will read 1 with RWO attribute.
1	0h RO	Reserved.
0	0h RO	**BIOS/OS Handoff (BOH):** Not supported.

24.8.2.9 Vendor Specific (VSP)—Offset A0h

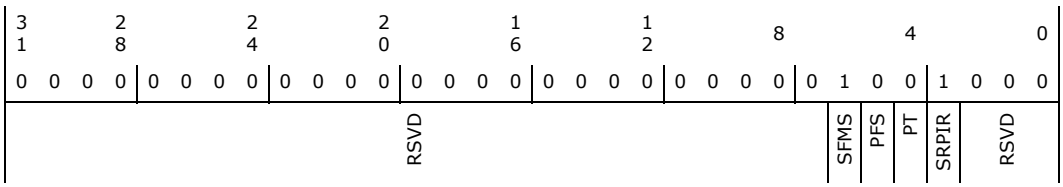
Vendor Specific

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 48h



Bit Range	Default and Access	Field Name (ID): Description
31:7	0h RO	Reserved.
6	1h RO	**Software Feature Mask Supported (SFMS):** Set to 1 if the platform is enabled for premium storage features mask (SFM) as described in VS MMIO space at offset ABAR[RPID.(OFST*4)+4].
5	0h RO	**Premium Features Supported (PFS):** Set to 1 if the platform is enabled for premium storage features (PFB) as described in VS MMIO space at offset ABAR[RPID.OFST*4]. Set to 0 if the platform is not enabled for premium storage features.
4	0h RO	**Platform Type (PT):** Set to 1 if mobile platform. Clear (0) if desktop.
3	1h RO	**Supports RAID Platform ID Reporting (SRPIR):** If set to 1, then indicates that the RAID Platform ID is reported via ABAR + C0h. Set to 0 if this ID is not reported via MMIO space.
2:0	0h RO	Reserved.

24.8.2.10 Vendor Specific Capabilities Register (VS_CAP)—Offset A4h

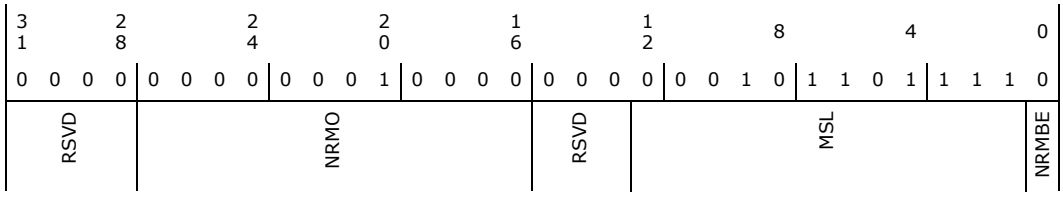
Vendor Specific Capabilities Register

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 1002DEh



Bit Range	Default and Access	Field Name (ID): Description
31:28	0h RO	Reserved.
27:16	10h RW/O	**NVM Remapped Register Offset (NRMO):** Specifies the offset (in 128B unit) within ABAR as to where the PCIe NAND memory BAR register space is remapped. For example, NRMO=1 means ABAR + 128B. This allows the remapped offset to shift between ABAR + 0B, and ABAR + 512KK - 128B, with 128B step. The remapped offset into the AHCI memory space must not overlap with the memory space used for SATA. The remapped offset into the AHCI memory space and the remapped size must not exceed the allocated AHCI memory space of the integrated AHCI controller. Only valid when NRMBE = 1. The reset default of this field is 10h, this places the start of the PCIe NAND memory BAR register space at ABAR + 2K. This field is not reset by FLR.

Bit Range	Default and Access	Field Name (ID): Description
15:13	0h RO	Reserved.
12:1	16Fh RW/O	**Memory Space Limit. (MSL):** This field specifies the size (in 128B unit) of the remapped memory space for the PCIe NAND device. It is a 0-based field. For example, MSL=1 means 256B. This allows the remapped size to shift between 128B and 512K with the step of 128B.Memory BAR offset from 0 to MSL of the PCIe NAND device are remapped under the integrated AHCI controller memory space.The remapped offset into the AHCI memory space and the value programmed in this field must not exceed the allocated AHCI memory space of the integrated AHCI controller. Only valid when NRMBE = 1. The reset default of this field is 16Fh which specifies the size of the remapped memory space as 34k.This field is not reset by FLR.
0	0h RW/O	**PCIe NAND Memory BAR Remapped Enable (NRMBE):** Set to 1 if a PCIe NAND device is present and remapping of its memory BAR register space is enabled. Cleared to 0 if there is no PCIe NAND device present or the remapping of its memory BAR register space is disabled. This bit is not reset by FLR.

24.8.2.11 RAID Platform ID (RPID)—Offset C0h

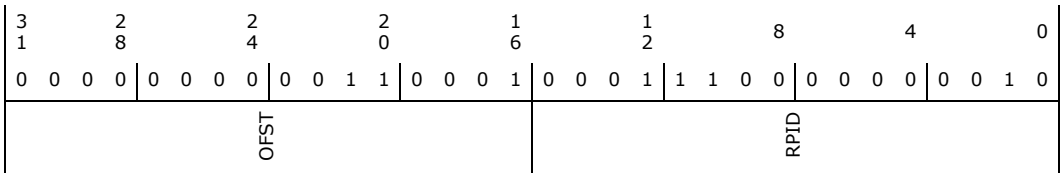
RAID Platform ID. This register is used by the Intel Matrix Storage Manager OROM to match the features supported by the OROM with the platform on which the OROM is executing. This prevents the use of an OROM designed for newer chipsets from being use on older chipsets as this could reduce up-sell potential.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: A206h



Bit Range	Default and Access	Field Name (ID): Description
31:16	31h RO	**Offset (OFST):** The offset of the Premium Feature Block (PFB) in DWords from the beginning of the ABAR. SFM follows directly after PFB.
15:0	1C02h RO	**RAID Platform ID (RPID):** Specifies the DID value that has been assigned to the platform. This is the same DID that is reported by the SATA controller when SATAGC.AIE is set to 1 except that the DID is always reported through this register, regardless if the programming of SATAGC.AIE.

24.8.2.12 Premium Feature Block (PFB)—Offset C4h

Premium Feature Block.

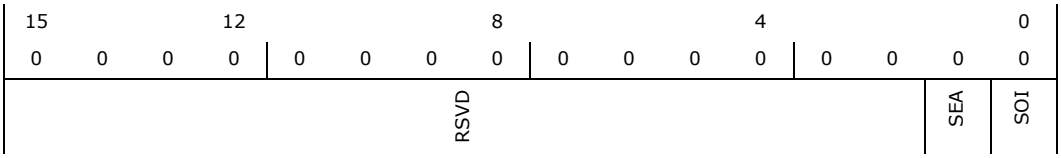
Note: Bits 4-0 are not bit-mapped to individual fuses and/or soft SKU settings; rather a single fuse FFSATA5 and FFSATA 3 /soft sku is used to indicate support for all of these features (refer to VSP.PFS). These registers indicate to the Intel Rapid Storage Technology AHCI driver that those premium RAID features that can be supported on the platform.

Access Method

Type: MEM Register
(Size: 16 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:2	0h RO	Reserved.
1	0h RO	**Supports Email Alert (SEA):** Read value is the same as VSP.PFS. If set to 1, then this feature is supported as part of the premium feature set.
0	0h RO	**Supports OEM IOCTL (SOI):** Read value is the same as VSP.PFS. If set to 1, then this feature is supported as part of the premium feature set.

24.8.2.13 SW Feature Mask (SFM)—Offset C8h

SW Feature Mask. The following will be programmed by the BIOS when VS_CAP.SFMS == 1. The feature mask is used by SW to determine which non-premium features shall be supported by SW. These register bits are not reset by FLR since they are programmed by BIOS.

Access Method

Type: MEM Register
(Size: 16 bits)

Device: 23
Function: 0

Default: 3Fh

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
RSVD				OROM_UI_Normal_Delay		Smart_Response_Technology	IRRT_Only_on_ESATA	LED_Locate	HDDUNLOCK	OROM_UI_and_BANNER	IRRT	R5	R10	R1	R0

Bit Range	Default and Access	Field Name (ID): Description
15:12	0h RO	Reserved.
11:10	0h RW/O	**OROM UI Normal Delay. (OROM_UI_Normal_Delay):** Values of these bits specify the delay of the OROM UI Splash Screen in a normal status. 00 = 2 secs (default and previous value); 01 = 4 secs; 10 = 6 secs; 11 = 8 secs. If bit 5 == 0, then these values are disregarded.
9	0h RW/O	**Smart Response Technology. (Smart_Response_Technology):** If set to '1', then Smart Response Technology is enabled. If cleared to '0', the feature is disabled.
8	0h RW/O	**RRT Only on ESATA (IRRT_Only_on_ESATA):** If set to 1, then only RRT volumes can span internal and external SATA ports (e.g., eSATA). If cleared to 0, then any RAID volume can span internal and external SATA ports (e.g., eSATA).
7	0h RW/O	**LED Locate (LED_Locate):** If set to 1, then LED/SGPIO hardware is attached and the ping to locate feature is enabled in the OS.
6	0h RW/O	**HDDUNLOCK (HDDUNLOCK):** If set to 1, then HDD password unlock is enabled in the OS.
5	1h RW/O	**OROM UI and BANNER (OROM_UI_and_BANNER):** If set to 1, then the OROM UI is displayed. When cleared to 0, the OROM UI and BANNER are not displayed if all disks and volumes have a normal status.
4	1h RW/O	**RRT (IRRT):** If set to 1, then Rapid Recovery Technology is enabled.
3	1h RW/O	**R5 (R5):** If set to 1, then RAID5 is enabled.
2	1h RW/O	**R10 (R10):** If set to 1, then RAID10 is enabled.
1	1h RW/O	**R1 (R1):** If set to 1, then RAID1 is enabled.
0	1h RW/O	**R0 (R0):** If set to 1, then RAID0 is enabled.

24.8.2.14 Port 0 Command List Base Address (P0CLB)—Offset 100h

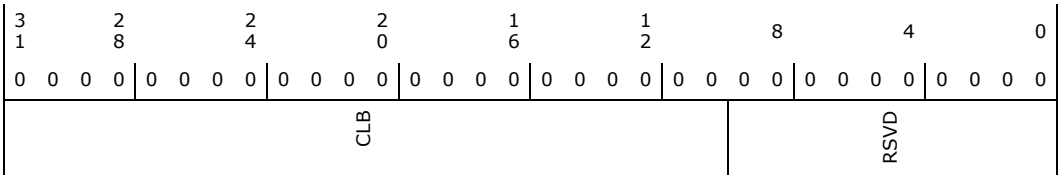
Port 0 Command List Base Address

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:10	0h RW	**Command List Base Address (CLB):** Indicates the 32-bit base for the command list for this port. This base is used when fetching commands to execute. This address must be 1K aligned as indicated by bits 31:10 being read/write. Note that these bits are not reset on a HBA reset.
9:0	0h RO	Reserved.

24.8.2.15 Port 0 Command List Base Address Upper 32-bits (P0CLBU)—Offset 104h

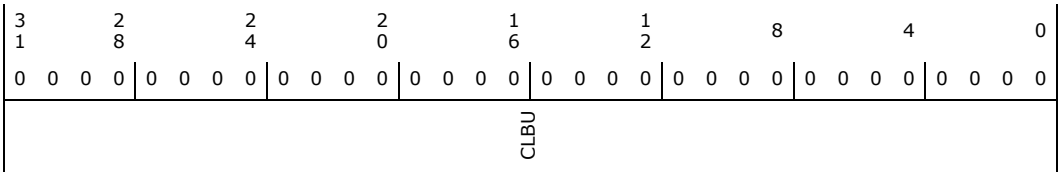
Port 0 Command List Base Address Upper 32-bits

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Command List Base Address Upper (CLBU):** Indicates the upper 32-bits for the command list base address for this port. This base is used when fetching commands to execute. Note that these bits are not reset on a HBA reset.

24.8.2.16 Port 0 FIS Base Address (P0FB)—Offset 108h

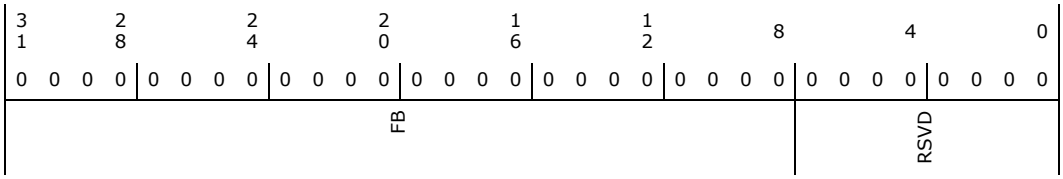
Port 0 FIS Base Address

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:8	0h RW	**FIS Base Address (FB):** Indicates the 32-bit base for received FISes. This address must be 256-byte aligned as indicated by bits 31:08 being read/write. When FIS-based switching is in use, this structure is 4KB in length and the address shall be 4KB aligned. Note that these bits are not reset on a HBA reset.
7:0	0h RO	Reserved.

24.8.2.17 Port 0 FIS Base Address Upper 32-Bits (P0FBU)—Offset 10Ch

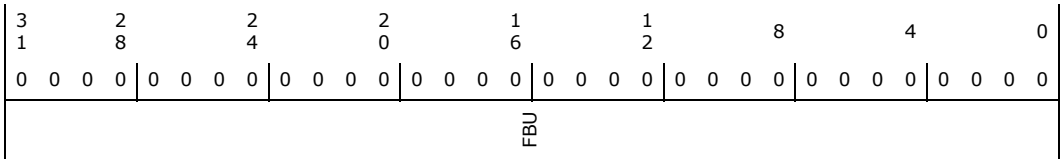
Port 0 FIS Base Address Upper 32-bits

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**FIS Base Address Upper (FBU):** Indicates the upper 32-bits for the received FIS base for this port. Note that these bits are not reset on a HBA reset.

24.8.2.18 Port 0 Interrupt Status (P0IS)—Offset 110h

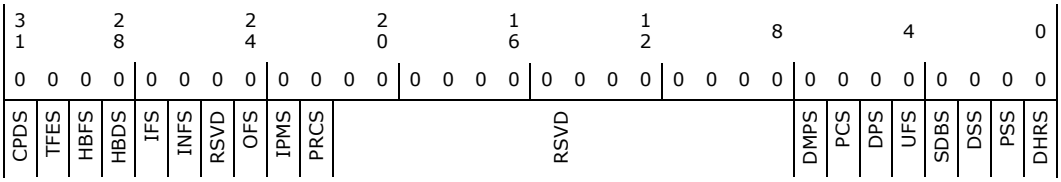
Port 0 Interrupt Status

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RO	**Cold Presence Detect Status (CPDS):** The SATA controller does not support cold presence detect.
30	0h RW/1C	**Task File Error Status (TFES):** This bit is set whenever the status register is updated by the device and the error bit (bit 0 of the Status field in the received FIS) is set.
29	0h RW/1C	**Host Bus Fatal Error Status (HBFS):** Indicates that the HBA encountered a host bus error that it cannot recover from, such as a bad software pointer. In PCI, such an indication would be a target or master abort.
28	0h RW/1C	**Host Bus Data Error Status (HBDS):** Indicates that the HBA encountered a data error (uncorrectable ECC / parity) when reading from or writing to system memory.
27	0h RW/1C	**Interface Fatal Error Status (IFS):** Indicates that the HBA encountered an error on the SATA interface which caused the transfer to stop.
26	0h RW/1C	**Interface Non-fatal Error Status (INFS):** Indicates that the HBA encountered an error on the SATA interface but was able to continue operation.
25	0h RO	Reserved.
24	0h RW/1C	**Overflow Status (OFS):** Indicates that the HBA received more bytes from a device than was specified in the PRD table for the command.
23	0h RW/1C	**Incorrect Port Multiplier Status (IPMS):** Indicates that the HBA received a FIS from a device whose port multiplier field did not match what was expected. The IPMS bit may be set during emuneration process. It is recommended that IPMS only be used after enumeration is complete on the Port Multiplier.
22	0h RO	**PhyRdy Change Status (PRCS):** When set to one indicates the internal PhyRdy signal changed state. This bit reflects the state of PxSERR.DIAG.N. This bit is RO and is only cleared when PxSERR.DIAG.N is cleared.
21:8	0h RO	Reserved.
7	0h RW/1C	**Device Mechanical Presence Status (DMPS):** When set, indicates that a platform mechanical presence switch has been opened or closed, which may lead to a change in the connection state of the device. This bit is only valid in systems that support mechanical presence switch (CAP.SMPS and PxCMD.MPSP are set).
6	0h RO	**Port Connect Change Status (PCS):** 1=Change in Current Connect Status. 0=No change in Current Connect Status. This bit reflects the state of PxSERR.DIAG.X. This bit is only cleared when PxSERR.DIAG.X is cleared.
5	0h RW/1C	**Descriptor Processed (DPS):** A PRD with the I. bit set has transferred all of its data.

Bit Range	Default and Access	Field Name (ID): Description
4	0h RO	**Unknown FIS Interrupt (UFS):** When set to 1 indicates that an unknown FIS was received and has been copied into system memory. This bit is cleared to 0 by software clearing the PxSERR.DIAG.F bit to 0. Note that this bit does not directly reflect the PxSERR.DIAG.F bit. PxSERR.DIAG.F is set immediately when an unknown FIS is detected, whereas this bit is set when the FIS is posted to memory. Software should wait to act on an unknown FIS until this bit is set to 1 or the two bits may become out of sync.
3	0h RW/1C	**Set Device Bits Interrupt (SDBS):** A Set Device Bits FIS has been received with the I. bit set and has been copied into system memory.
2	0h RW/1C	**DMA Setup FIS Interrupt (DSS):** A DMA Setup FIS has been received with the I. bit set and has been copied into system memory.
1	0h RW/1C	**PIO Setup FIS Interrupt (PSS):** A PIO Setup FIS has been received with the I. bit set, it has been copied into system memory, and the data related to that FIS has been transferred. This bit shall be set even if the data transfer resulted in an error.
0	0h RW/1C	**Device to Host Register FIS Interrupt (DHRS):** A D2H register FIS has been received with the I. bit set, and has been copied into system memory.

24.8.2.19 Port 0 Interrupt Enable (P0IE)—Offset 114h

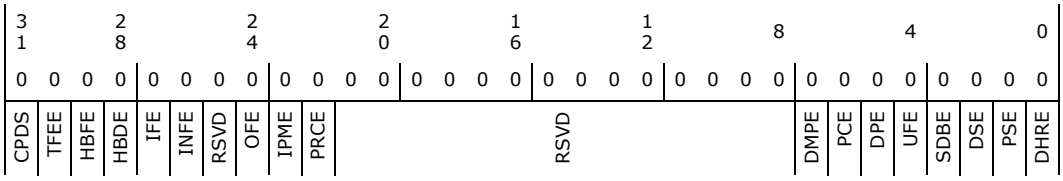
Port 0 Interrupt Enable

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RO	**Cold Presence Detect Enable (CPDS):** The SATA controller does not support cold presence detect.
30	0h RW	**Task File Error Enable (TFEE):** When set, GHC.IE is set, and P0S.TFES is set, the HBA shall generate an interrupt.
29	0h RW	**Host Bus Fatal Error Enable (HBFE):** When set, GHC.IE is set, and P0IS.HBFS is set, the HBA shall generate an interrupt.
28	0h RW	**Host Bus Data Error Enable (HBDE):** When set, GHC.IE is set, and P0IS.HBDS is set, the HBA shall generate an interrupt.
27	0h RW	**Interface Fatal Error Enable (IFE):** When set, GHC.IE is set, and P0IS.IFS is set, the HBA shall generate an interrupt.
26	0h RW	**Interface Non-fatal Error Enable (INFE):** When set, GHC.IE is set, and P0IS.INFS is set, the HBA shall generate an interrupt.
25	0h RO	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
24	0h RW	**Overflow Enable (OFE):** When set, and GHC.IE and P0IS.OFS are set, the HBA shall generate an interrupt.
23	0h RW	**Incorrect Port Multiplier Enable (IPME):** When set, and GHC.IE and P0IS.IPMS are set, the HBA shall generate an interrupt. BIOS is required to program this field to '0'. The same applies to AHCI driver.
22	0h RW	**PhyRdy Change Interrupt Enable (PRCE):** When set, and GHC.IE is set, and PxIS.PRCS is set, the HBA shall generate an interrupt.
21:8	0h RO	Reserved.
7	0h RW	**Device Mechanical Enable (DMPE):** When set, and P0IS.DMPS is set, the HBA shall generate an interrupt.
6	0h RW	**Port Change Interrupt Enable (PCE):** When set, GHC.IE is set, and P0IS.PCS is set, the HBA shall generate an interrupt.
5	0h RW	**Descriptor Processed Interrupt Enable (DPE):** When set, GHC.IE is set, and P0IS.DPS is set, the HBA shall generate an interrupt.
4	0h RW	**Unknown FIS Interrupt Enable (UFE):** When set, GHC.IE is set, and PxIS.UFS is set to 1, the HBA shall generate an interrupt.
3	0h RW	**Set Device Bits FIS Interrupt Enable (SDBE):** When set, GHC.IE is set, and P0IS.SDBS is set, the HBA shall generate an interrupt.
2	0h RW	**DMA Setup FIS Interrupt Enable (DSE):** When set, GHC.IE is set, and P0IS.DSS is set, the HBA shall generate an interrupt.
1	0h RW	**PIO Setup FIS Interrupt Enable (PSE):** When set, GHC.IE is set, and P0IS.PSS is set, the HBA shall generate an interrupt.
0	0h RW	**Device to Host Register FIS Interrupt Enable (DHRE):** When set, GHC.IE is set, and P0IS.DHRS is set, the HBA shall generate an interrupt.

24.8.2.20 Port 0 Command (P0CMD)—Offset 118h

Port 0 Command

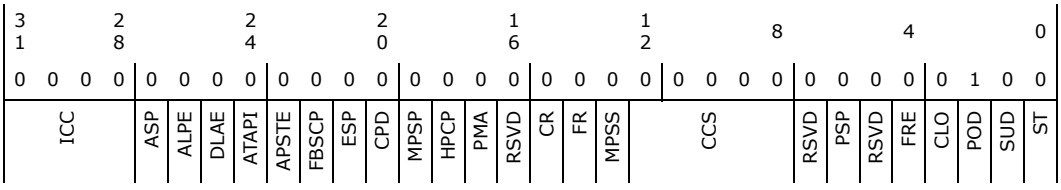
Access Method

Type: MEM Register (Size: 32 bits)

Device: 23

Function: 0

Default: 4h



Bit Range	Default and Access	Field Name (ID): Description
31:28	0h RW	**Interface Communication Control (ICC):** This is a four bit field which can be used to control reset and power states of the interface. If the Link layer is currently in the L_IDLE state, writes to this field shall cause the HBA to initiate a transition to the interface power management state requested. If the Link layer is not currently in the L_IDLE state, writes to this field shall have no effect.

Bit Range	Default and Access	Field Name (ID): Description
27	0h RW	**Aggressive Slumber Partial (ASP):** When set to 1, and the ALPE bit is set, the HBA shall aggressively enter the Slumber state when it clears a bit in the PxCI or PxSACT register and the register values are then PxCI = 0h and PxSACT = 0h. When cleared, and the ALPE bit is set, the HBA will aggressively enter the Partial state when it clears a bit in the PxCI or PxSACT register and the register values are then PxCI = 0h and PxSACT = 0h. If CAP.SALP is cleared to 0., software shall treat this bit as reserved.
26	0h RW	**Aggressive Link Power Management Enable (ALPE):** When set, the HBA will aggressively enter a lower link power state (Partial or Slumber) based upon the setting of the ASP bit. Software shall only set this bit to 1 if CAP.SALP is set to 1. If CAP.SALP is cleared to 0, software shall treat this bit as reserved. BIOS is recommended to program this field to 1.
25	0h RW	**Drive LED on ATAPI Enable (DLAE):** When set, the HBA will drive the LED pin active for commands regardless of the state of PxCMD.ATAPI. When cleared, the HBA will only drive the LED pin active for commands if PxCMD.ATAPI is set to 0. This bit is set by software
24	0h RW	**Device is ATAPI (ATAPI):** When set, the connected device is an ATAPI device. This bit is used by the HBA to control whether or not to generate the desktop LED when commands are active.
23	0h RW	**Automatic Partial to Slumber Transitions Enabled (APSTE):** When set to 1., the HBA may perform Automatic Partial to Slumber Transitions. When cleared to 0. the port shall not perform Automatic Partial to Slumber Transitions. Software shall only set this bit to 1. if CAP2.APST is set to 1.; if CAP2.APST is cleared to 0. software shall treat this bit as reserved.
22	0h RO	**FIS-based Switching Capable Port (FBSCP):** The SATA controller does not support FIS-Based Switching.
21	0h RW/O	**External SATA Port (ESP):** When set to 1, indicates that this port is routed externally and will be used with an external SATA device. When set to 1, CAP.SXS must also be set to 1. When cleared (0), indicates that this port is not routed externally and supports internal SATA devices only. If ESP is set to '1', then the port may experience hot plug events.
20	0h RO	**Cold Presence Detection (CPD):** The SATA controller does not support cold presence detect.
19	0h RW/O	**Mechanical Presence Switch Attached to Port (MPSP):** If set to 1, the platform supports a mechanical presence switch attached to this port. If cleared to 0, the platform does not support a mechanical presence switch attached to this port.
18	0h RW/O	**Hot Plug Capable Port (HPCP):** This indicates whether the this port is connected to a device which can be hot plugged. SATA by definition is hot-pluggable, but not all platforms are constructed to allow the device to be removed (it may be screwed into the chassis, for example). This bit can be used by system software to indicate a feature such as eject device to the end-user.
17	0h RO	**Port Multiplier Attached (PMA):** When set, a Port Multiplier is attached to the HBA for this port. When cleared, a Port Multiplier is not attached to the HBA for this port. This bit is a read only 0. when CAP.PMS = 0., and read/write when CAP.PMS = 1. Note that this bit is set by software; hardware does not auto-detect that a Port Multiplier is attached.
16	0h RO	Reserved.
15	0h RO	**Command List Running (CR):** When this bit is set it indicates that the command list DMA engine for the port is running.
14	0h RO	**FIS Receive Running (FR):** When this bit is set it indicates that the FIS Receive DMA engine for the port is running. Note to software: When FR bit stays set, read the PxIS.PCS and PxSERR.DET.X to service the PCS interrupt accordingly if any
13	0h RO	**Mechanical Presence Switch State (MPSS):** The MPSS bit reports the state of a mechanical presence switch attached to this port. If CAP.SMPS is set to 1 and the mechanical presence switch is closed then this bit is cleared to 0. If CAP.SMPS is set to 1 and the mechanical presence switch is open then this bit is set to 1. If CAP.SMPS is set to 0 then this bit is cleared to 0. Software should only use this bit if both CAP.SMPS and PxCMD.MPSP are set to 1.

Bit Range	Default and Access	Field Name (ID): Description
12:8	0h RO	**Current Command Slot (CCS):** Indicates the current command slot the HBA is processing. This field is valid when the PxCMD.ST bit is set, and is constantly updated by the HBA. This field can be updated as soon as the HBA recognizes an active command slot, or at some point soon after when it begins processing the command. When PxCMD.ST transitions from a 1 to a 0, the HBA will reset this field to 0. After PxCMD.ST transitions from 0 to 1, the highest priority slot to issue from next is command slot 0. After the first command has been issued, the highest priority slot to issue from next is PxCMD.CCS + 1. For example, after the HBA has issued its first command, if PxCMD.CCS = 0h and PxCI is set to 3h, the next command that will be issued is from command slot 1.
7	0h RO	Reserved.
6	0h RO	**PHYSLP Present (PSP):** If set to '1', the platform supports PHYSLP on this port. If cleared to '0', the platform does not support PHYSLP on this port. This bit may only be set to '1' if CAP2.SPS is set to '1'.
5	0h RO	Reserved.
4	0h RW	**FIS Receive Enable (FRE):** When set, the HBA may post received FISes into the FIS receive area pointed to by PxFB and PxFBU. When cleared, received FISes are not accepted by the HBA, except for the first D2H register FIS after the initialization sequence.
3	0h RW/1S	**Command List Override (CLO):** Setting this bit to 1 causes PxTFD.STS.BSY and PxTFD.STS.DRQ to be cleared to 0. This allows a software reset to be transmitted to the device regardless of whether the BSY and DRQ bits are still set in the PxTFD.STS register. The HBA sets this bit to 0 when PxTFD.STS.BSY and PxTFD.STS.DRQ have been cleared to 0. A write to this register with a value of 0 shall have no effect.
2	1h RO	**Power On Device (POD):** The SATA controller does not support cold presence detect.
1	0h RW	**Spin-Up Device (SUD):** This bit is read/write for HBAs that support staggered spin-up via CAP.SSS. This bit is read only 1. for HBAs that do not support staggered spin-up. On an edge detect from 0. to 1, the HBA shall start a COMRESET initialization sequence to the device. Clearing this bit causes no action on the interface. Clearing this bit to0 does not cause any OOB signal to be sent on the interface. When this bit is cleared to 0. and PxSCTL.DET = 0h, the HBA will enter listen mode.
0	0h RW	**Start (ST):** When set, the HBA may process the command list. When cleared, the HBA may not process the command list. Whenever this bit is changed from a 0 to a 1, the HBA starts processing the command list at entry 0. Whenever this bit is changed from a 1 to a 0, the PxCI and PxSACT register is cleared by the HBA upon the HBA putting the controller into an idle state. PxTFD shall be updated also.

24.8.2.21 Port 0 Task File Data (P0TFD)—Offset 120h

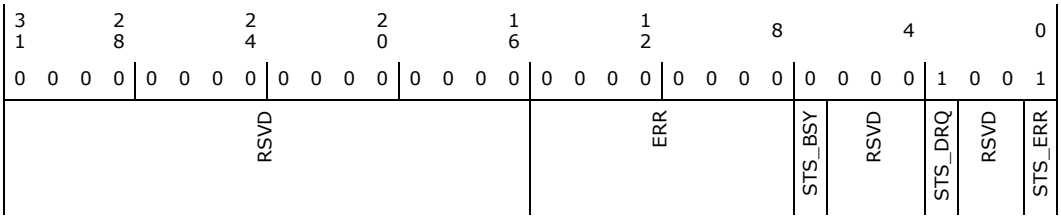
Port 0 Task File Data

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 9h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:8	0h RO	**Error (ERR):** Contains the latest copy of the task file error register.
7	0h RO	**Status Busy (STS_BSY):** Status - Indicates the interface is busy.
6:4	0h RO	Reserved.
3	1h RO	**Status Drq (STS_DRQ):** Status - Indicates a data transfer is requested.
2:1	0h RO	Reserved.
0	1h RO	**Status Err (STS_ERR):** Status - Indicates an error during the transfer.

24.8.2.22 Port 0 Signature (P0SIG)—Offset 124h

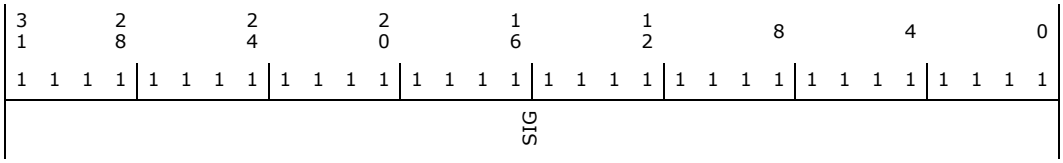
Port 0 Signature

Access Method

Type: MEM Register (Size: 32 bits)

Device: 23
Function: 0

Default: FFFFFFFFh



Bit Range	Default and Access	Field Name (ID): Description
31:0	FFFFFFFFh RO	**Signature (SIG):** Contains the signature received from a device on the first D2H Register FIS.

24.8.2.23 Port 0 Serial ATA Status (P0SSTS)—Offset 128h

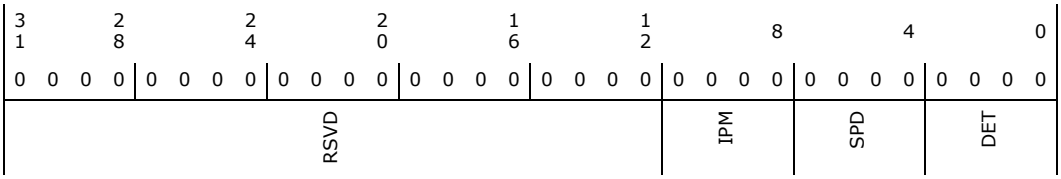
Port 0 Serial ATA Status

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:12	0h RO	Reserved.
11:8	0h RO	**Interface Power Management (IPM):** Indicates the current interface state
7:4	0h RO	**Current Interface Speed (SPD):** Indicates the negotiated interface communication speed.
3:0	0h RO	**Device Detection (DET):** Indicates the interface device detection and PHY state.

24.8.2.24 Port 0 Serial ATA Control (P0SCTL)—Offset 12Ch

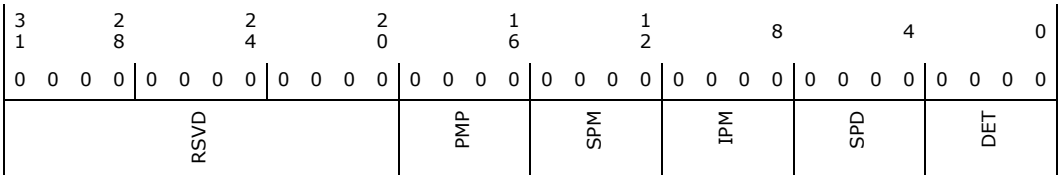
Port 0 Serial ATA Control

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:20	0h RO	Reserved.
19:16	0h RW	**Port Multiplier Port (PMP):** This field is not used by AHCI.
15:12	0h RW	**Select Power Management (SPM):** This field is not used by AHCI.
11:8	0h RW	**Interface Power Management Transitions Allowed (IPM):** Indicates which power states the HBA is allowed to transition to. If an interface power management state is not allowed via this register field, the HBA will not initiate that state and the HBA will PMNAKP any request from the device to enter that state.
7:4	0h RW	**Speed Allowed (SPD):** Indicates the highest allowable speed of the interface.
3:0	0h RW	**Device Detection Initialization (DET):** Controls the HBA.s device detection and interface initialization.

24.8.2.25 Port 0 Serial ATA Error (P0SERR)—Offset 130h

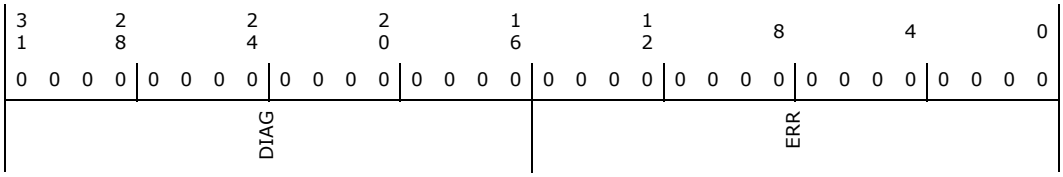
Port 0 Serial ATA Error

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RW/1C	**Diagnostics (DIAG):** Contains diagnostic error information for use by diagnostic software in validating correct operation or isolating failure modes.
15:0	0h RW/1C	**Error (ERR):** The ERR field contains error information for use by host software in determining the appropriate response to the error condition. If one or more of bits 11:8 of this register are set, the controller will stop the current transfer.

24.8.2.26 Port 0 Serial ATA Active (P0SACT)—Offset 134h

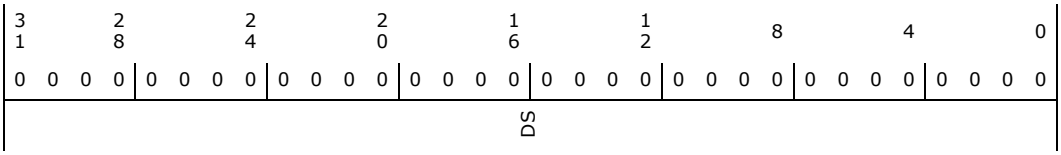
Port 0 Serial ATA Active

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/1S	**Device Status (DS):** System software sets this bit for native command queuing commands prior to setting the PxCI.CI bit in the same command slot entry. This field is cleared via the Set Device Bits FIS.

24.8.2.27 Port 0 Command Issue (P0CI)—Offset 138h

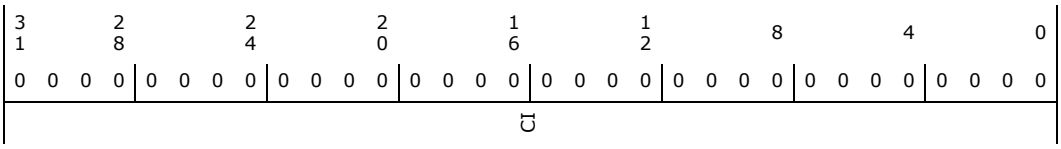
Port 0 Commands Issued

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Commands Issued (CI):** This field is set by software to indicate to the HBA that a command has been built in system memory for a command slot and may be sent to the device. When the HBA receives a FIS which clears the BSY, ERR, and DRQ bits for the command, it clears the corresponding bit in this register for that command slot. Bits in this field shall only be set to 1 by software when PxCMD.ST is set to 1.

24.8.2.28 Port 0 SNotification (P0SNTF)—Offset 13Ch

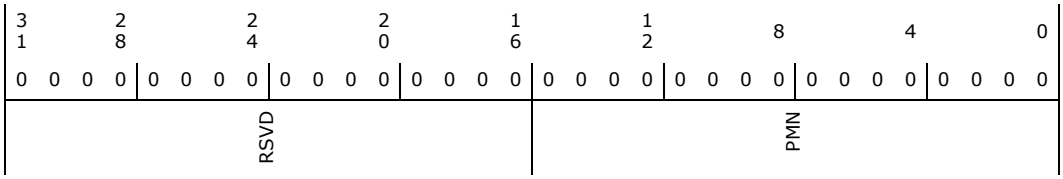
Port 0 SNotification

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:0	0h RW/1C	**PM Notify (PMN):** This field indicates whether a particular device with the corresponding PM Port number issued a Set Device Bits FIS to the host with the Notification bit set. PM Port 0h sets bit 0. PM Port Fh sets bit 15. Individual bits are cleared by software writing 1's to the corresponding bit positions. Note that, while this field is reset to default on a HBA Reset, it is not reset by COMRESET or SRST

24.8.2.29 Port 0 Device Sleep (P0DEVSLP)—Offset 144h

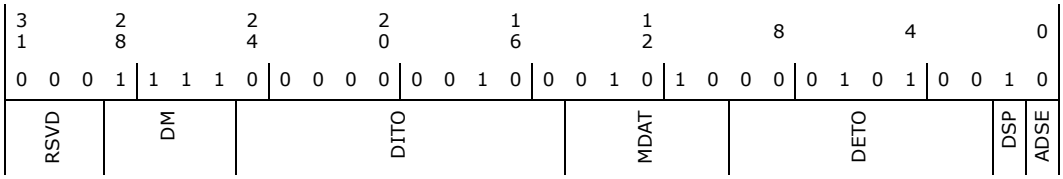
Port 0 Device Sleep

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 1E022852h



Bit Range	Default and Access	Field Name (ID): Description
31:29	0h RO	Reserved.
28:25	Fh RW/O	**DITO Multiplier (DM):** 0's based value that specifies the DITO multiplier that the HBA applies to the specified DITO value, effectively extending the range of DITO from 1ms to 16368ms. A value of 0h indicates a multiplier of 1. A maximum multiplier of 16 may be applied. The HBA computes the total idle timeout as a product of DM and DITO (i.e., DITO actual = DITO * DM).

Bit Range	Default and Access	Field Name (ID): Description
24:15	4h RW	**DEVSLP Idle Timeout (DITO):** This field specifies the amount of the time (in approximate 1ms granularity) that the HBA shall wait before driving the DEVSLP signal.Hardware reloads its port specific DEVSLP timer with this value each time the port transitions out of DEVSLP state. For example: from DEVSLP to active or PxDEVSLP.ADSE transitions from 0 to a 1. If CAP2.SDS or CAP2.SADM or PxDEVSLP.DSP is cleared to 0 then these bits are read-only 0h and software shall treat these bits as reserved. Software shall only set this value when PxCMD.ST is cleared to 0 and PxDEVSLP.ADSE is cleared to 0.
14:10	Ah RW	**DEVSLP Minimum Assertion Time (MDAT):** This field specifies the minimum amount of time (in 1ms granularity) that the HBA must assert the DEVSLP signal before it may be de-asserted. The nominal value is 10ms and the minimum is 1ms depending on device identification information. If CAP2.SDS is cleared to 0 or PxDEVSLP.DSP is cleared to 0 then these bits are read-only 0h and software shall treat these bits as reserved. Software shall only set this value when PxCMD.ST is cleared to 0, PxDEVSLP.ADSE is cleared to 0, and prior to setting PxCMD.ICC to 8h.
9:2	14h RW	**DEVSLP Exit Timeout (DETO):** This field specifies the maximum duration (in approximate 1ms granularity) from DEVSLP de-assertion until the device is ready to accept OOB. The nominal value is 20ms while the max value is 255ms depending on device identification information. If CAP2.SDS is cleared to 0 or PxDEVSLP.DSP is cleared to 0 then these bits are read-only 0h and software shall treat these bits as reserved. Software shall only set this value when PxCMD.ST is cleared to 0, PxDEVSLP.ADSE is cleared to 0, and prior to setting PxCMD.ICC to 8h.
1	1h RW/O	**DEVSLP Present (DSP):** If set to '1', the platform supports DEVSLP on this port. If cleared to '0', the platform does not support DEVSLP on this port. This bit may only be set to '1' if CAP2.SDS is set to '1'.DSP is mutually exclusive with the PxCMD.HPCP bit and PxCMD.ESP bit. Note that these bits are not reset on a HBA reset.
0	0h RW	**Aggressive DEVSLP Enable (ADSE):** This bit is read/write for HBAs that support aggressive DEVSLP management (CAP2. SADM = ?1?). When this bit is set to ?1?, the HBA shall assert the DEVSLP signal after the port has been idle (PxCI = 0h and PxSACT = 0h) for the amount of time specified by the PxDEVSLP.DITO register and the interface is in Slumber (PxSSTS.IPM = 6h). When this bit is cleared to ?0?, the HBA does not enter DEVSLP unless software directed via PxCMD.ICC. This bit shall only be set to ?1? if PxDEVSLP.DSP is set to 1. If this bit is set to 1 and software clears the bit to 0, then the HBA shall de-assert the DEVSLP signal if asserted. Note that these bits are not reset on a HBA reset. BIOS is recommended to program this field to 1 if the platform support the DEVSLP feature. If CAP2.SDS is cleared to 0 or CAP2.SADM is cleared to 0, or if PxDEVSLP.DSP is cleared to 0 then these bits are read-only 0h and software shall treat these bits as reserved.

24.8.2.30 Port 1 Command List Base Address (P1CLB)—Offset 180h

Same bit definition as P0CLB.

24.8.2.31 Port 1 Command List Base Address Upper 32-bits (P1CLBU)—Offset 184h

Same bit definition as P0CLBU.

24.8.2.32 Port 1 FIS Base Address (P1FB)—Offset 188h

Same bit definition as P0FB.

24.8.2.33 Port 1 FIS Base Address Upper 32-bits (P1FBU)—Offset 18Ch

Same bit definition as P0FBU.

24.8.2.34 Port 1 Interrupt Status (P1IS)—Offset 190h

Same bit definition as P0IS.

24.8.3 Port 1 Interrupt Enable (P1IE)—Offset 194h

Same bit definition as P0IE.

24.8.3.1 Port 1 Command (P1CMD)—Offset 198h

Same bit definition as P0CMD.

24.8.3.2 Port 1 Task File Data (P1TFD)—Offset 1A0h

Same bit definition as P0TFD.

24.8.3.3 Port 1 Signature (P1SIG)—Offset 1A4h

Same bit definition as P0SIG.

24.8.3.4 Port 1 Serial ATA Status (P1SSTS)—Offset 1A8h

Same bit definition as P0SSTS.

24.8.3.5 Port 1 Serial ATA Control (P1SCTL)—Offset 1ACh

Same bit definition as P0SCTL.

24.8.3.6 Port 1 Serial ATA Error (P1SERR)—Offset 1B0h

Same bit definition as P0SERR.

24.8.3.7 Port 1 Serial ATA Active (P1SACT)—Offset 1B4h

Same bit definition as P0SACT.

24.8.3.8 Port 1 Command Issue (P1CI)—Offset 1B8h

Same bit definition as P0CI.

24.8.3.9 Port 1 SNotification (P1SNTF)—Offset 1BCh

Same bit definition as P0SNTF.

24.8.3.10 Port 1 Device Sleep (P1DEVSLP)—Offset 1C4h

Same bit definition as P0DEVSLP.

24.8.3.11 Port 2 Command List Base Address (P2CLB)—Offset 200h

Same bit definition as P0CLB.

24.8.3.12 Port 2 Command List Base Address Upper 32-bits (P2CLBU)—Offset 204h

Same bit definition as P0CLBU.

24.8.3.13 Port 2 FIS Base Address (P2FB)—Offset 208h

Same bit definition as P0FB.

24.8.3.14 Port 2 FIS Base Address Upper 32-bits (P2FBU)—Offset 20Ch

Same bit definition as P0FBU.

24.8.3.15 Port 2 Interrupt Status (P2IS)—Offset 210h

Same bit definition as P0IS.

24.8.3.16 Port 2 Interrupt Enable (P2IE)—Offset 214h

Same bit definition as P0IE.

24.8.3.17 Port 2 Command (P2CMD)—Offset 218h

Same bit definition as P0CMD.

24.8.3.18 Port 2 Task File Data (P2TFD)—Offset 220h

Same bit definition as P0TFD.

24.8.3.19 Port 2 Signature (P2SIG)—Offset 224h

Same bit definition as P0SIG.

24.8.3.20 Port 2 Serial ATA Status (P2SSTS)—Offset 228h

Same bit definition as P0SSTS.

24.8.3.21 Port 2 Serial ATA Control (P2SCTL)—Offset 22Ch

Same bit definition as P0SCTL.

24.8.3.22 Port 2 Serial ATA Error (P2SERR)—Offset 230h

Same bit definition as P0SERR.

24.8.3.23 Port 2 Serial ATA Active (P2SACT)—Offset 234h

Same bit definition as P0SACT.

24.8.3.24 Port 2 Command Issue (P2CI)—Offset 238h

Same bit definition as P0CI.

24.8.3.25 Port 2 SNotification (P2SNTF)—Offset 23Ch

Same bit definition as P0SNTF.

24.8.3.26 Port 2 Device Sleep (P2DEVSLP)—Offset 244h

Same bit definition as P0DEVSLP.

24.8.3.27 Port 3 Command List Base Address (P3CLB)—Offset 280h

Same bit definition as P0CLB.

24.8.3.28 Port 3 Command List Base Address Upper 32-bits (P3CLBU)—Offset 284h

Same bit definition as P0CLBU.

24.8.3.29 Port 3 FIS Base Address (P3FB)—Offset 288h

Same bit definition as P0FB.

24.8.3.30 Port 3 FIS Base Address Upper 32-bits (P3FBU)—Offset 28Ch

Same bit definition as P0FBU.

24.8.3.31 Port 3 Interrupt Status (P3IS)—Offset 290h

Same bit definition as P0IS.

24.8.3.32 Port 3 Interrupt Enable (P3IE)—Offset 294h

Same bit definition as P0IE.

24.8.3.33 Port 3 Command (P3CMD)—Offset 298h

Same bit definition as P0CMD.

24.8.3.34 Port 3 Task File Data (P3TFD)—Offset 2A0h

Same bit definition as P0TFD.

24.8.3.35 Port 3 Signature (P3SIG)—Offset 2A4h

Same bit definition as P0SIG.

24.8.3.36 Port 3 Serial ATA Status (P3SSTS)—Offset 2A8h

Same bit definition as P0SSTS.

24.8.3.37 Port 3 Serial ATA Control (P3SCTL)—Offset 2ACh

Same bit definition as P0SCTL.

24.8.3.38 Port 3 Serial ATA Error (P3SERR)—Offset 2B0h

Same bit definition as P0SERR.

24.8.3.39 Port 3 Serial ATA Active (P3SACT)—Offset 2B4h

Same bit definition as P0SACT.

24.8.3.40 Port 3 Commands Issued (P3CI)—Offset 2B8h

Same bit definition as P0CI.

24.8.3.41 Port 3 SNotification (P3SNTF)—Offset 2BCh

Same bit definition as P0SNTF.

24.8.3.42 Port 3 Device Sleep (P3DEVSLP)—Offset 2C4h

Same bit definition as P0DEVSLP.

24.8.3.43 Port 4 Command List Base Address (P4CLB)—Offset 300h

Same bit definition as P0CLB.

24.8.3.44 Port 4 Command List Base Address Upper 32-bits (P4CLBU)—Offset 304h

Same bit definition as P0CLBU.

24.8.3.45 Port 4 FIS Base Address (P4FB)—Offset 308h

Same bit definition as P0FB.

24.8.3.46 Port 4 FIS Base Address Upper 32-bits (P4FBU)—Offset 30Ch

Same bit definition as P0FBU.

24.8.3.47 Port 4 Interrupt Status (P4IS)—Offset 310h

Same bit definition as P0IS.

24.8.3.48 Port 4 Interrupt Enable (P4IE)—Offset 314h

Same bit definition as P0IE.

24.8.3.49 Port 4 Command (P4CMD)—Offset 318h

Same bit definition as P0CMD.

24.8.3.50 Port 4 Task File Data (P4TFD)—Offset 320h

Same bit description as P0TFD.

24.8.3.51 Port 4 Signature (P4SIG)—Offset 324h

Same bit description as P0SIG.

24.8.3.52 Port 4 Serial ATA Status (P4SSTS)—Offset 328h

Same bit description as P0SSTS.

24.8.3.53 Port 4 Serial ATA Control (P4SCTL)—Offset 32Ch

Same bit description as P0SCTL.

24.8.3.54 Port 4 Serial ATA Error (P4SERR)—Offset 330h

Same bit description as P0SERR.

24.8.3.55 Port 4 Serial ATA Active (P4SACT)—Offset 334h

Same bit description as P0SACT.

24.8.3.56 Port 4 Commands Issued (P4CI)—Offset 338h

Same bit description as P0CI.

24.8.3.57 Port 4 SNotification (P4SNTF)—Offset 33Ch

Same bit description as P0SNTF.

24.8.3.58 Port 4 Device Sleep (P4DEVSLP)—Offset 344h

Same bit description as P0DEVSLP.

24.8.3.59 Port 5 Command List Base Address (P5CLB)—Offset 380h

Same bit description as P0CLB.

24.8.3.60 Port 5 Command List Base Address Upper 32-bits (P5CLBU)—Offset 384h

Same bit definition as P0CLBU.

24.8.3.61 Port 5 FIS Base Address (P5FB)—Offset 388h

Same bit definition as P0FB.

24.8.3.62 Port 5 FIS Base Address Upper 32-bits (P5FBU)—Offset 38Ch

Same bit definition as P0FBU.

24.8.3.63 Port 5 Interrupt Status (P5IS)—Offset 390h

Same bit definition as P0IS.

24.8.3.64 Port 5 Interrupt Enable (P5IE)—Offset 394h

Same bit definition as P0IE.

24.8.3.65 Port 5 Command (P5CMD)—Offset 398h

Same bit definition as P0CMD.

24.8.3.66 Port 5 Task File Data (P5TFD)—Offset 3A0h

Same bit definition as P0TFD.

24.8.3.67 Port 5 Signature (P5SIG)—Offset 3A4h

Same bit definition as P0SIG.

24.8.3.68 Port 5 Serial ATA Status (P5SSTS)—Offset 3A8h

Same bit definition as P0SSTS.

24.8.3.69 Port 5 Serial ATA Control (P5SCTL)—Offset 3ACh

Same bit definition as P0SCTL.

24.8.3.70 Port 5 Serial ATA Error (P5SERR)—Offset 3B0h

Same bit definition as P0SERR.

24.8.3.71 Port 5 Serial ATA Active (P5SACT)—Offset 3B4h

Same bit definition as P0SACT.

24.8.3.72 Port 5 Commands Issued (P5CI)—Offset 3B8h

Same bit definition as P0CI.

24.8.3.73 Port 5 SNotification (P5SNTF)—Offset 3BCh

Same bit definition as P0SNTF.

24.8.3.74 Port 5 Device Sleep (P5DEVSLP)—Offset 3C4h

Same bit definition as P0DEVSLP.

24.8.3.75 Port 6 Command List Base Address (P6CLB)—Offset 400h

Same bit definition as P0CLB.

24.8.3.76 Port 6 Command List Base Address Upper 32-bits (P6CLBU)—Offset 404h

Same bit definition as P0CLBU.

24.8.3.77 Port 6 FIS Base Address (P6FB)—Offset 408h

Same bit definition as P0FB.

24.8.3.78 Port 6 FIS Base Address Upper 32-bits (P6FBU)—Offset 40Ch

Same bit definition as P0FBU.

24.8.3.79 Port 6 Interrupt Status (P6IS)—Offset 410h

Same bit definition as P0IS.

24.8.3.80 Port 6 Interrupt Enable (P6IE)—Offset 414h

Same bit definition as P0IE.

24.8.3.81 Port 6 Command (P6CMD)—Offset 418h

Same bit definition as P0CMD.

24.8.3.82 Port 6 Task File Data (P6TFD)—Offset 420h

Same bit definition as P0TFD.

24.8.3.83 Port 6 Signature (P6SIG)—Offset 424h

Same bit definition as P0SIG.

24.8.3.84 Port 6 Serial ATA Status (P6SSTS)—Offset 428h

Same bit definition as P0SSTS.

24.8.3.85 Port 6 Serial ATA Control (P6SCTL)—Offset 42Ch

Same bit definition as P0SCTL.

24.8.3.86 Port 6 Serial ATA Error (P6SERR)—Offset 430h

Same bit definition as P0SERR.

24.8.3.87 Port 6 Serial ATA Active (P6SACT)—Offset 434h

Same bit definition as P0SACT.

24.8.3.88 Port 6 Commands Issued (P6CI)—Offset 438h

Same bit definition as P0CI.

24.8.3.89 Port 6 SNotification (P6SNTF)—Offset 43Ch

Same bit definition as P0SNTF.

24.8.3.90 Port 6 Device Sleep (P6DEVSLP)—Offset 444h

Same bit definition as P0DEVSLP.

24.8.3.91 Port 7 Command List Base Address (P7CLB)—Offset 480h

Same bit definition as P0CLB.

24.8.3.92 Port 7 Command List Base Address Upper 32-bits (P7CLBU)—Offset 484h

Same bit definition as P0CLBU.

24.8.3.93 Port 7 FIS Base Address (P7FB)—Offset 488h

Same bit definition as P0FB.

24.8.3.94 Port 7 FIS Base Address Upper 32-bits (P7FBU)—Offset 48Ch

Same bit definition as P0FBU.

24.8.3.95 Port 7 Interrupt Status (P7IS)—Offset 490h

Same bit definition as P0IS.

24.8.3.96 Port 7 Interrupt Enable (P7IE)—Offset 494h

Same bit definition as P0IE.

24.8.3.97 Port 7 Command (P7CMD)—Offset 498h

Same bit definition as P0CMD.

24.8.3.98 Port 7 Task File Data (P7TFD)—Offset 4A0h

Same bit definition as P0TFD.

24.8.3.99 Port 7 Signature (P7SIG)—Offset 4A4h

Same bit definition as P0SIG.

24.8.3.100 Port 7 Serial ATA Status (P7SSTS)—Offset 4A8h

Same bit definition as P0SSTS.

24.8.3.101 Port 7 Serial ATA Control (P7SCTL)—Offset 4ACh

Same bit definition as P0SCTL.

24.8.3.102 Port 7 Serial ATA Error (P7SERR)—Offset 4B0h

Same bit definition as P0SERR.

24.8.3.103 Port 7 Serial ATA Active (P7SACT)—Offset 4B4h

Same bit definition as P0SACT.

24.8.3.104 Port 7 Commands Issued (P7CI)—Offset 4B8h

Same bit definition as P0CI.

24.8.3.105 Port 7 SNotification (P7SNTF)—Offset 4BCh

Same bit definition as P0SNTF.

24.8.3.106 Port 7 Device Sleep (P7DEVSLP)—Offset 4C4h

Same bit definition as P0DEVSLP.

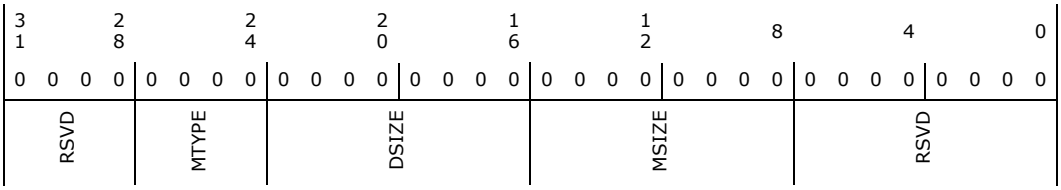
24.8.3.107 Enclosure Management Message Format (EM_MF)—Offset 580h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:28	0h RO	Reserved.
27:24	0h RW	**Message Type (MTYPE):** Specifies the type of the message. The message types are: 0h = LED; 1h = SAF-TE; 2h = SES-2; 3h = SGPIO (register based interface); All other values reserved.

Bit Range	Default and Access	Field Name (ID): Description
23:16	0h RW	**Data Size (DSIZE):** Specifies the data size in bytes. If the message (enclosure services command) has a data buffer that is associated with it that is transferred, the size of that data buffer is specified in this field. If there is no separate data buffer, this field shall have a value of '0'. The data directly follows the message in the message buffer. This value should always be '0'.
15:8	0h RW	**Message Size (MSIZE):** Specifies the size of the message in bytes. The message size does not include the one Dword header. A value of '0'. is invalid. The message size is always 4 bytes.
7:0	0h RO	Reserved.

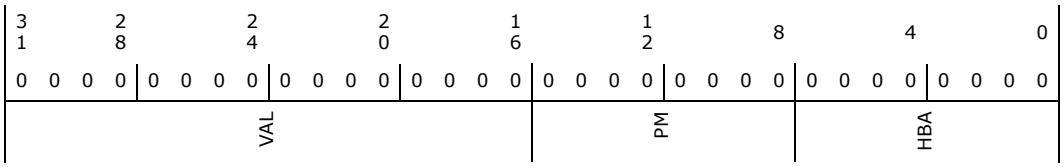
24.8.3.108 Enclosure Management LED (EM_LED)—Offset 584h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RW	**Value (VAL):** This field describes the state of each LED for a particular location. There are three LEDs that may be supported by the HBA. Each LED has 3 bits of control. LED values are: 000b - LED must be off 001b - LED must be solid on as perceived by human eye; All other values reserved. The LED bit locations are: Bits 2:0 - Activity LED (may be driven by hardware) Bits 5:3 - Vendor Specific LED (e.g., locate) Bits 8:6 - Vendor Specific LED (e.g., fault) Bits 15:9 - Reserved. Vendor specific message is: Bit 3:0 - Vendor Specific Pattern Bit 15:4 - Reserved. ***Note:*** If Activity LED Hardware Driven (ATTR.ALHD) bit is set, host will output the hardware LED value sampled internally and will ignore software written activity value on bit [2:0]. Since Enclosure Management does not support port multiplier based LED message, the LED message will be generated independently based on respective port's operation activity. Vendor specific LED values Locate (Bits 5:3) and Fault (Bits 8:6) always are driven by software.
15:8	0h RW	**Port Multiplier Information (PM):** Specifies slot specific information related to Port Multiplier. Bits 3:0 specify the Port Multiplier port number for the slot that requires the status update. If a Port Multiplier is not attached to the device in the affected slot, the Port Multiplier port number shall be '0'. Bits 7:4 are reserved. SATA does not support LED messages for devices behind a Port Multiplier. This byte should be 0.
7:0	0h RW	**HBA Information (HBA):** Specifies slot specific information related to the HBA. Bits 4:0 - HBA port number for the slot that requires the status update. Bit 5 - If set to '1', Value is a vendor specific message that applies to the entire enclosure. If cleared to '0', Value applies to the port specified in bits 4:0. Bits 7:6 - Reserved.

24.8.4 SATA AIDP Registers Summary

Table 24-3. Summary of SATA AIDP Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
10h	13h	AHCI Index Register (INDEX)—Offset 10h	0h
14h	17h	AHCI Data Register (DATA)—Offset 14h	0h

24.8.4.1 AHCI Index Register (INDEX)—Offset 10h

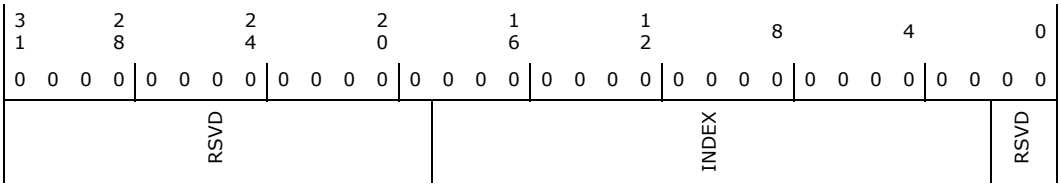
AHCI Index Register. This registers are only available if CC.SCC is not 01h to index into all memory registers defined in Memory Registers and the message buffer used for enclosure management. If CC.SCC is 01h, these AHCI Index Data Pair registers are not accessible and SINDEX/SDATA register pair shall be used to index into a subset of the memory registers defined in (See Memory Registers for more information on which registers could be indexed).

Access Method

Type: I/O Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:19	0h RO	Reserved.
18:2	0h RW	**Index (INDEX):** This Index register is used to select the Dword offset of the Memory Mapped AHCI register to be accessed. A Dword, Word or Byte access is specified by the active byte enables of the I/O access to the Data register.
1:0	0h RO	Reserved.

24.8.4.2 AHCI Data Register (DATA)—Offset 14h

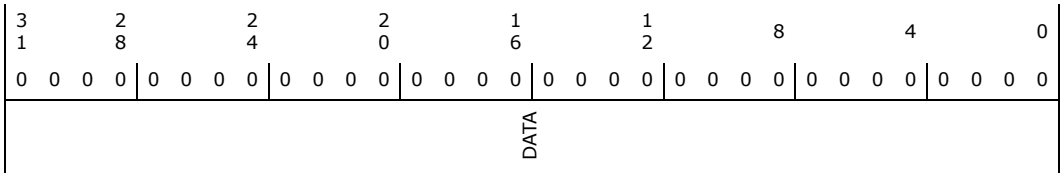
AHCI Data Register. This registers are index into all memory registers defined in Memory Registers and the message buffer used for enclosure management.

Access Method

Type: I/O Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Data (DATA):** This Data register is a window through which data is read or written to the memory mapped register pointed to by the Index register. Note that a physical register is not actually implemented as the data is actually stored in the memory mapped registers. Since this is not a physical register, the default value is the same as the default value of the register pointed to by Index.

24.8.5 SATA MXTBA Registers Summary

Table 24-4. Summary of SATA MXTBA Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	MSI-X Table Entries 0 Message Lower Address (MXTE0MLA)—Offset 0h	0h
4h	7h	MSI-X Table Entries 0 Message Upper Address (MXTE0MUA)—Offset 4h	0h
8h	Bh	MSI-X Table Entries 0 Message Data (MXTE0MD)—Offset 8h	0h
Ch	Fh	MSI-X Table Entries 0 Vector Control (MXTE0VC)—Offset Ch	1h

24.8.5.1 MSI-X Table Entries 0 Message Lower Address (MXTE0MLA)—Offset 0h

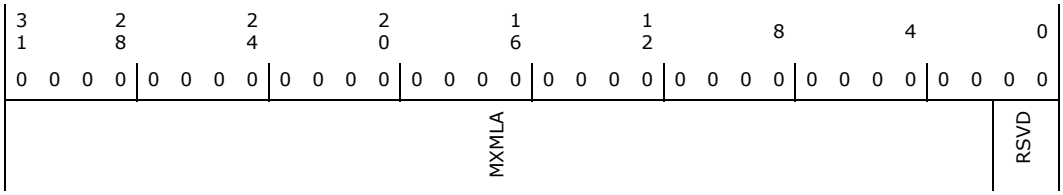
MSI-X Table Entries 0 Message Lower Address

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:2	0h RW	**MSI-X message lower address (MXMLA):** Specifies the lower 32-bit of the DWORD-aligned address (Addr[31:02]) of the MSI-X Message.
1:0	0h RO	Reserved.

24.8.5.2 MSI-X Table Entries 0 Message Upper Address (MXTE0MUA)—Offset 4h

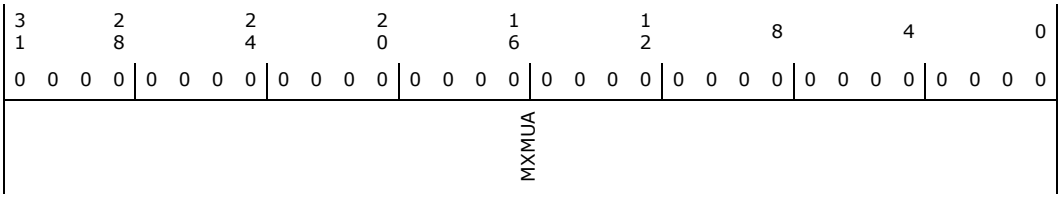
MSI-X Table Entries 0 Message Upper Address

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**MSI-X message upper 32-bit address (MXMUA):** Specifies the lower 32-bit of the DWORD-aligned address (Addr[31:02]) of the MSI-X Message.

24.8.5.3 MSI-X Table Entries 0 Message Data (MXTE0MD)—Offset 8h

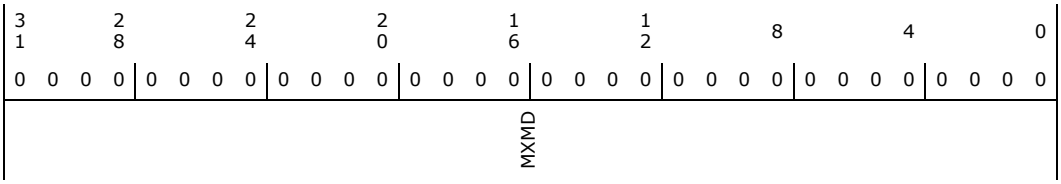
MSI-X Table Entries 0 Message Data

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**MSI-X message Data (MXMD):** Specifies the 32-bit Data of the MSI-X Message.

24.8.5.4 MSI-X Table Entries 0 Vector Control (MXTE0VC)—Offset Ch

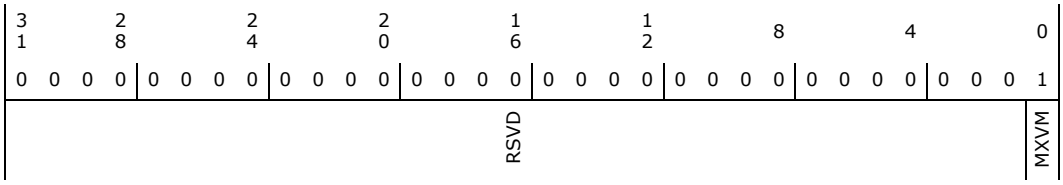
MSI-X Table Entries 0 Vector Control

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 23
Function: 0

Default: 1h



Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	1h RW	**MSI-X vector Mask (MXVM):** When this bit is set, the function is prohibited from sending a message using this MSI-X Table entry. However, any other MSI-X Table entries programmed with the same vector will still be capable of sending an equivalent message unless they are also masked. This bit's state after reset is 1 (entry is masked).

24.8.6 SATA MXPBA Registers Summary

Table 24-5. Summary of SATA MXPBA Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	MSI-X Pending Bit Array QW 0 (MXPQW0_DW0)—Offset 0h	0h

24.8.6.1 MSI-X Pending Bit Array QW 0 (MXPQW0_DW0)—Offset 0h

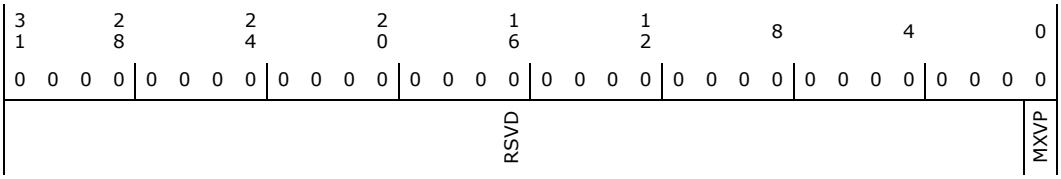
MSI-X Pending Bit Array QW 0

Access Method

Type: MEM Register (Size: 32 bits)

Device: 23
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h RO	**MSI-X vector Pending (MXVP):** For each Pending Bit that is set, the function has a pending message for the associated MSI-X Table entry. Pending bits that have no associated MSI-X Table entry are reserved. After reset, the state of reserved Pending bits must be 0. Software should never write, and should only read Pending Bits. If software writes to Pending Bits, the result is undefined. Each Pending Bit's state after reset is 0 (no message pending).

24.9 SSATA Port Registers

24.9.1 SSATA Configuration Registers Summary

Table 24-6. Summary of SATA Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	Identifiers (ID)—Offset 0h	1C028086h
4h	5h	Command (CMD)—Offset 4h	0h
6h	7h	Device Status (STS)—Offset 6h	210h
8h	8h	Revision ID (RID)—Offset 8h	0h
9h	9h	Programming Interface (PI)—Offset 9h	1h
Ah	Bh	Class Code (CC)—Offset Ah	106h
Ch	Ch	Cache Line Size (CLS)—Offset Ch	0h
Dh	Dh	Master Latency Timer (MLT)—Offset Dh	0h
Eh	Eh	Header Type (HTYPE)—Offset Eh	0h
10h	13h	MSI-X Table Base Address (MXTBA)—Offset 10h	0h
14h	17h	MXP Base Address (MXPBA)—Offset 14h	0h
20h	23h	AHCI Index Data Pair Base Address (AIDPBA)—Offset 20h	1h
24h	27h	AHCI Base Address (ABAR)—Offset 24h	0h
2Ch	2Fh	Sub System Identifiers (SS)—Offset 2Ch	0h

Table 24-6. Summary of SATA Configuration Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
34h	34h	Capabilities Pointer (CAP)—Offset 34h	80h
3Ch	3Dh	Interrupt Information (INTR)—Offset 3Ch	100h
70h	71h	PCI Power Management Capability ID (PID)—Offset 70h	A801h
72h	73h	PCI Power Management Capabilities (PC)—Offset 72h	4003h
74h	75h	PCI Power Management Control and Status (PMCS)—Offset 74h	8h
80h	81h	Message Signaled Interrupt Identifier (MID)—Offset 80h	7005h
82h	83h	Message Signaled Interrupt Message Control (MC)—Offset 82h	0h
84h	87h	Message Signaled Interrupt Message Address (MA)—Offset 84h	0h
88h	89h	Message Signaled Interrupt Message Data (MD)—Offset 88h	0h
90h	93h	Port Mapping Register (MAP)—Offset 90h	0h
94h	97h	Port Control and Status (PCS)—Offset 94h	0h
9Ch	9Fh	SATA General Configuration (SATAGC)—Offset 9Ch	0h
A0h	A0h	SATA Initialization Register Index (SIRI)—Offset A0h	0h
A4h	A7h	SATA Initialization Register Data (SIRD)—Offset A4h	0h
A8h	ABh	Serial ATA Capability Register 0 (SATACR0)—Offset A8h	100012h
ACh	AFh	Serial ATA Capability Register 1 (SATACR1)—Offset ACh	48h
C0h	C3h	Scratch Pad (SP)—Offset C0h	0h
D0h	D1h	MSI-X Identifiers (MXID)—Offset D0h	11h
D2h	D3h	MSI-X Message Control (MXC)—Offset D2h	0h
D4h	D7h	MSI-X Table Offset/Table BIR (MXT)—Offset D4h	0h
D8h	DBh	MSI-X PBA Offset / PBA BIR (MXP)—Offset D8h	0h
E0h	E3h	BIST FIS Control/Status (BFCS)—Offset E0h	0h
E4h	E7h	BIST FIS Transmit Data 1 (BFTD1)—Offset E4h	0h
E8h	EBh	BIST FIS Transmit Data 2 (BFTD2)—Offset E8h	0h

24.9.1.1 Identifiers (ID)—Offset 0h

Identifiers

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 1C028086h

31			28				24				20				16				12				8				4				0
1	0	1	0	0	0	2	0	0	1	0	1	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0
DID																VID															

Bit Range	Default and Access	Field Name (ID): Description
31:16	A1D2/ A252h RO	**Device ID (DID):** Indicates the Device ID of the SSATA controller. For Super SKUs this will be A252. For Production SKUs it will be A1D2.
15:0	8086h RO	**Vendor ID (VID):** 16-bit field which indicates the company vendor as Intel.

24.9.1.2 Command (CMD)—Offset 4h

Command

Access Method

Type: CFG Register (Size: 16 bits)

Device: 17
Function: 5

Default: 0h

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD					ID	RSVD	SEE	RSVD	PEE	RSVD			BME	MSE	IOSE

Bit Range	Default and Access	Field Name (ID): Description
15:11	0h RO	Reserved.
10	0h RW	**Interrupt Disable (ID):** This disables pin-based INTx# interrupts. This bit has no effect on MSI operation. When set, internal INTx# will not be generated. When cleared, internal INTx# are generated if there is an interrupt and MSI is not enabled.
9	0h RO	Reserved.
8	0h RW	**SERR# Enable (SEE):** When set to 1, the HBA is allowed to generate SERR# on DPD or SATAGC.URD event that is enabled for SERR# generation. When cleared to 0, it is not.
7	0h RO	Reserved.
6	0h RW	**Parity Error Response Enable (PEE):** When set, the SATA Controller will corrupt the outbound DATA FIS CRC if a forwarded data parity error is indicated.
5:3	0h RO	Reserved.
2	0h RW	**Bus Master Enable (BME):** Controls the SATA Controller's ability to act as a master for data transfers. This bit does not impact the generation of completions for split transaction commands.
1	0h RW	**Memory Space Enable (MSE):** Controls access to the SATA Controller's target memory space (for AHCI).
0	0h RW	**I/O Space Enable (IOSE):** Controls access to the SATA Controller's target I/O space.

24.9.1.3 Device Status (STS)—Offset 6h

Device Status

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 5

Default: 210h

15			12				8				4				0
0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0
DPE	SSE	RMA	RTA	STA	DEVT		DPD	RSVD			CL	IS	RSVD		

Bit Range	Default and Access	Field Name (ID): Description
15	0h RW/1C	**Detected Parity Error (DPE):** Set when the SATA Controller detects a parity error on its interface.
14	0h RW/1C	**Signaled System Error (SSE):** Set when SATA Controller generates an SERR#.
13	0h RW/1C	**Received Master-Abort Status (RMA):** Set when the SATA Controller receives a master abort to a cycle it generated.
12	0h RW/1C	**Received Target-Abort Status (RTA):** Set when the SATA Controller receives a target abort to a cycle it generated.
11	0h RW/1C	**Signaled Target-Abort Status (STA):** This bit must be set by a target device whenever it terminates a transaction with Target-Abort. Devices that will never signal Target-Abort do not need to implement this bit.
10:9	1h RO	**DEVSEL# Timing Status (DEVT):** Controls the device select time for the SATA Controller's PCI interface.
8	0h RW/1C	**Master Data Parity Error Detected (DPD):** Set when the SATA Controller, as a master, either detects a parity error or sees the parity error line asserted, and the parity error response bit (bit 6 of the command register) is set. This bit can only be set on read completions received from the backbone where there is a parity error.
7:5	0h RO	Reserved.
4	1h RO	**Capabilities List (CL):** Indicates the presence of a capabilities list. The minimum requirement for the capabilities list must be PCI power management for the SATA Controller.
3	0h RO	**Interrupt Status (IS):** Reflects the state of INTx# messages, IRQ14 or IRQ15. This bit is set when the interrupt is to be asserted. This bit is a 0 after the interrupt is cleared.
2:0	0h RO	Reserved.

24.9.1.4 Revision ID (RID)—Offset 8h

Revision ID

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 5

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
RID							

Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO	**Revision ID (RID):** Indicates stepping of the host controller hardware.

24.9.1.5 Programming Interface (PI)—Offset 9h

Programming Interface

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 5

Default: 01h

7			4				0
0	0	0	0	0	0	0	1
IF							

Bit Range	Default and Access	Field Name (ID): Description
7:0	1h RO	**Interface (IF):** If CC.SCC=06h(AHCI mode), it indicates that this is an AHCI HBA that has a major revision of 1 (as specified in the AHCI Version register). If CC.SCC=04h(RAID mode), it indicates that there is no programming interface(IF=00h). Internally, under this condition, the SATA controller is in native mode and its I/O spaces are only accessible through the I/O BARs.

24.9.1.6 Class Code (CC)—Offset Ah

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 5

Default: 0106h

15			12				8				4				0
0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	0
BCC								SCC							

Bit Range	Default and Access	Field Name (ID): Description
15:8	1h RO	**Base Class Code (BCC):** Indicates that this is a mass storage device.
7:0	6h RO	**Sub Class Code (SCC):** Indicates the sub class code.

24.9.1.7 Cache Line Size (CLS)—Offset Ch

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 5

Default: 00h

7			4				0
0	0	0	0	0	0	0	0
CLS							

Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO	**Cache Line Size (CLS):** This register has no meaning for the SATA controller.

24.9.1.8 Master Latency Timer (MLT)—Offset Dh

Master Latency Timer

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 5

Default: 00h

7			4				0
0	0	0	0	0	0	0	0
MLT							

Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO	**Master Latency Timer (MLT):** This register has no meaning for the SATA controller.

24.9.1.9 Header Type (HTYPE)—Offset Eh

Header Type

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 5

Default: 80h

7			4				0
1	0	0	0	0	0	0	0
MFD	HL						

Bit Range	Default and Access	Field Name (ID): Description
7	1b RO	**Multi-function Device (MFD):** Indicates this controller is part of a multi-function device.
6:0	0h RO	**Header Layout (HL):** Indicates that the controller uses a target device layout.

24.9.1.10 MSI-X Table Base Address (MXTBA)—Offset 10h

MSI-X Table Base Address. This BAR is used to allocate 32K, 16K or 8K Memory space for the MSI-X Table. The Memory space size is determined by BIOS by making bit-14 and bit-13 Read-Only '1' or Read-Write '0' based on SATAGC.MSS[1:0].

Type: CFG Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BA																		BAB14	BAB13	RSVD									PF	TP		RTE

Bit Range	Default and Access	Field Name (ID): Description
31:15	0h RW	**Base Address (BA):** Base address of memory space.
14	0h RW	**Base Address Bit 14 (BAB14):** When SATAGC.MSS[1:0]=00, this bit is Read Only '0' else it's Read Write '0'.
13	0h RW	**Base Address Bit 13 (BAB13):** When SATAGC.MSS[1:0]=00 or 01, this bit is Read Only '0' else it's Read Write '0'.
12:4	0h RO	Reserved.
3	0h RO	**Prefetchable (PF):** Indicates that this range is not pre-fetchable.
2:1	0h RO	**Type (TP):** Indicates that this range can be mapped anywhere in 32-bit address space.
0	0h RO	**Resource Type Indicator (RTE):** Indicates a request for Memory space.

24.9.1.11 MXP Base Address (MXPBA)—Offset 14h

This BAR is used to allocate 256-byte Memory space for the MSI-X PBA.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BA																							RSVD					PF	TP		RTE

Bit Range	Default and Access	Field Name (ID): Description
31:8	0h RW	**Base Address (BA):** Base address of memory space (aligned to 256B).
7:4	0h RO	Reserved.
3	0h RO	**Prefetchable (PF):** Indicates that this range is not pre-fetchable.
2:1	0h RO	**Type (TP):** Indicates that this range can be mapped anywhere in 32-bit address space.
0	0h RO	**Resource Type Indicator (RTE):** Indicates a request for Memory space.

24.9.1.12 AHCI Index Data Pair Base Address (AIDPBA)—Offset 20h

AHCI Index Data Pair Base Address. This BAR is used to allocate I/O space for the AHCI index/data pair mechanism. Note that hardware does not clear the BA bits (including BA4) when switching from IDE mode to non-IDE mode or vice versa. The BIOS is responsible for clearing those bits to 0 since the number of writable bits changes after mode switching.

Type: CFG Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 00000001h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
RSVD																BA												RSVD				RTE

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:5	0h RW	**Base Address (BA):** Base address of the I/O space.
4:1	0h RO	Reserved.
0	1h RO	**Resource Type Indicator (RTE):** Indicates a request for I/O space.

24.9.1.13 AHCI Base Address (ABAR)—Offset 24h

ABAR - AHCI Base Address. This register represents a memory BAR allocating space for the AHCI memory registers. Note that bit [31:16] of this register must be programmed to a value greater than 0 to ensure the memory is mapped to an address of 1 MBytes and greater (i.e., ABAR must be 00010000h or greater). Otherwise, memory cycle targeting the ABAR range may not be accepted. The Memory space size is determined by BIOS by making bit 15:11 Read-Only '1' or Read-Write '0' based on SATAGC.ASSEL[1:0].

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BA													BAB18	BAB17	BAB16	BAB15	BAB14	BAB1311			RSVD							PF	TP		RTE	

Bit Range	Default and Access	Field Name (ID): Description
31:19	0h RW	**Base Address (BA):** Base address of register memory space.
18	0h RW	**Base Address Bit 18 (BAB18):** When SATAGC.ASSEL[2:0] selects an ABAR size bigger than 256K, this bit is Read Only '0' else it's Read Write '0'.
17	0h RW	**Base Address Bit 17 (BAB17):** When SATAGC.ASSEL[2:0] selects an ABAR size bigger than 128K, this bit is Read Only '0' else it's Read Write '0'.
16	0h RW	**Base Address Bit 16 (BAB16):** When SATAGC.ASSEL[2:0] selects an ABAR size bigger than 64K, this bit is Read Only '0' else it's Read Write '0'.
15	0h RW	**Base Address Bit 15 (BAB15):** When SATAGC.ASSEL[2:0] selects an ABAR size bigger than 32K, this bit is Read Only '0' else it's Read Write '0'.
14	0h RW	**Base Address Bit 14 (BAB14):** When SATAGC.ASSEL[2:0] selects an ABAR size bigger than 16K, this bit is Read Only '0' else it's Read Write '0'.
13:11	0h RW	**Base Address Bit 13-11 (BAB1311):** When SATAGC.ASSEL[2:0] selects an ABAR size bigger than 2K, this bit is Read Only '0' else it's Read Write '0'.
10:4	0h RO	Reserved.
3	0h RO	**Prefetchable (PF):** Indicates that this range is not pre-fetchable
2:1	0h RO	**Type (TP):** Indicates that this range can be mapped anywhere in 32-bit address space
0	0h RO	**Resource Type Indicator (RTE):** Indicates a request for register memory space.

24.9.1.14 Sub System Identifiers (SS)—Offset 2Ch

Sub System Identifiers. This register is initialized to logic 0 by the assertion of PLTRST#. This register can be written only once after PLTRST# de-assertion. This register is not reset by FLR.

Type: CFG Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SSID																SSVID																

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RW/O	**Subsystem ID (SSID):** This is written by BIOS. No hardware action taken on this value.
15:0	0h RW/O	**Subsystem Vendor ID (SSVID):** This is written by BIOS. No hardware action taken on this value.

24.9.1.15 Capabilities Pointer (CAP)—Offset 34h

Capabilities Pointer

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 0

Default: 80h

7			4				0
1	0	0	0	0	0	0	0
CP							

Bit Range	Default and Access	Field Name (ID): Description
7:0	80h RW/L	**Capability Pointer (CP):** Indicates that the first capability pointer offset is offset 80h (the Message Signaled Interrupt capability). The following capability structures are linked by default: CAP.CP -> 80h (MSI) -> D0h (MSI-X)-> 70h (PCI Power) -> A8h (SATA) -> 00h end. BIOS may alter the capability structure list above (by programming a leading capability structure's Next Pointer field) if BIOS wants to bypass any specific capability. The RW/L register attribute allows for flexibility in determining the capability structure available in this PCI function. Refer to SATAGC.REGLOCK description in order to lock the register to become RO. This register is not reset by FLR.

24.9.1.16 Interrupt Information (INTR)—Offset 3Ch

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 5

Default: 0100h

15			12				8				4				0
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
IPIN								ILINE							

Bit Range	Default and Access	Field Name (ID): Description
15:8	1h RW/O	**Interrupt Pin (IPIN):** This register tells which interrupt pin the device function uses. A value of 1 corresponds to INTA#. A value of 2 corresponds to INTB#. A value of 3 corresponds to INTC#. A value of 4 corresponds to INTD#. This register is not reset by FLR.
7:0	0h RW	**Interrupt Line (ILINE):** Software written value to indicate which interrupt line (vector) the interrupt is connected to. No hardware action is taken on this register. Interrupt Line register is not reset by FLR.

24.9.1.17 PCI Power Management Capability ID (PID)—Offset 70h

PCI Power Management Capability ID

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 5

Default: A801h

15			12				8				4				0
1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	1
NEXT								CID							

Bit Range	Default and Access	Field Name (ID): Description
15:8	A8h RW/L	**Next Capability (NEXT):** A8h is location of the Serial ATA Capability structure. The RW/L register attribute allows for flexibility in determining the capability structure available in this PCI function. Refer to SATAGC.REGLOCK description in order to lock the register to become RO. This register is not reset by FLR.
7:0	1h RO	**Cap ID (CID):** Indicates that this pointer is a PCI power management capability.

24.9.1.18 PCI Power Management Capabilities (PC)—Offset 72h

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 5

Default: 4003h

15			12				8				4				0
0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1
PME_Support					D2_Support	D1_Support	Aux_Current			DSI	RSVD	PMEC	VS		

Bit Range	Default and Access	Field Name (ID): Description
15:11	8h RO	**PME_Support (PME_Support):** The default value is 01000 which indicates PME# can be generated from the $D3_{HOT}$ state in the SATA controller.
10	0h RO	**D2_Support (D2_Support):** The D2 state is not supported.
9	0h RO	**D1_Support (D1_Support):** The D1 state is not supported.
8:6	0h RO	**Aux_Current (Aux_Current):** PME# from $D3_{COLD}$ state is not supported, therefore this field is 000b.
5	0h RO	**Device Specific Initialization (DSI):** Indicates that no device-specific initialization is required.
4	0h RO	**Reserved.**
3	0h RO	**PME Clock (PMEC):** Indicates that PCI clock is not required to generate PME#.
2:0	3h RO	**Version (VS):** Indicates support for Revision 1.2 of the PCI Power Management Specification.

24.9.1.19 PCI Power Management Control and Status (PMCS)—Offset 74h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 5

Default: 8h

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
PMES	RSVD						PMEE	RSVD				NSFRST	RSVD	PS	

Bit Range	Default and Access	Field Name (ID): Description
15	0h RW/1C	**PME Status (PMES):** This bit is set when a PME event is to be requested, and if this bit is set and PMEE is set, a PME# will be generated. This register field is not reset by FLR.
14:9	0h RO	Reserved.
8	0h RW	**PME Enable (PMEE):** When set, the SATA controller generates PME# from $D3_{HOT}$ on a wake event. ***Note:*** Software is advised to clear PMEE together with PMES prior to changing CC.SCC through SATAGC.SMS. This register field is not reset by FLR.
7:4	0h RO	Reserved.
3	1h RO	**No Soft Reset (NSFRST):** A 1 indicates that devices transitioning from $D3_{HOT}$ to D0 because of PowerState commands do not perform an internal reset. Configuration context is preserved. Upon transition from the $D3_{HOT}$ to the D0 state initialized state, no additional operating system intervention is required to preserve configuration context beyond writing to the PowerState bits. Regardless of this bit, the controller transition from $D3_{HOT}$ to the D0 by a system or bus segment reset will return to the state D0 uninitialized with only PME context preserved if PME is supported and enabled.
2	0h RO	Reserved.
1:0	0h RW	**Power State (PS):** This field is used both to determine the current power state of the SATA Controller and to set a new power state. The values are: 00 = D0 state; 11 = $D3_{HOT}$ state. When in the $D3_{HOT}$ state, the controller's configuration space is available, but the I/O and memory spaces are not. Additionally, interrupts are blocked. If software attempts to write a 10 or 01 to these bits, the write will be ignored. Refer to *PCI PM Specification* Section 8.2.2. on software requirements in ensuring I/O space, memory space and Bus Master are disabled prior to entering D3 state.

24.9.1.20 Message Signaled Interrupt Identifier (MID)—Offset 80h

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 5

Default: 7005h

15			12				8				4				0
0	1	1	1	0	0	0	0	0	0	0	0	0	1	0	1
NEXT								CID							

Bit Range	Default and Access	Field Name (ID): Description
15:8	70h RW/L	**Next Pointer (NEXT):** Indicates the next item in the list is the PCI power management pointer. This is the recommended value. BIOS may program this field to A8h indicating that the next item is Serial ATA Capability structure. The RW/L register attribute allows for flexibility in determining the capability structure available in this PCI function. Refer to SATAGC.REGLOCK description in order to lock the register to become RO. This register is not reset by FLR.
7:0	5h RO	**Capability ID (CID):** Capabilities ID indicates MSI.

24.9.1.21 Message Signaled Interrupt Message Control (MC)—Offset 82h

Message Signaled Interrupt Message Control

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 5

Default: 0000h

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD								C64	MME			MMC			MSIE

Bit Range	Default and Access	Field Name (ID): Description
15:8	0h RO	Reserved.
7	0h RO	**64 Bit Address Capable (C64):** Capable of generating a 32-bit message only.
6:4	0h RO	**Multiple Message Enable (MME):** When this field is cleared to 000 (and MSIE is set), only a single MSI message will be generated for all SATA ports, and bits [15:0] of the message vector will be driven from MD[15:0].
3:1	0h RO	**Multiple Message Capable (MMC):** Not supported.
0	0h RW	**MSI Enable (MSIE):** If set, MSI is enabled and traditional interrupt pins are not used to generate interrupts. Note that CMD.ID bit has not effect on MSI. Software must clear this bit to 0 to disable MSI first before changing the number of messages allocated in the MMC field.

24.9.1.22 Message Signaled Interrupt Message Address (MA)—Offset 84h

Message Signaled Interrupt Message Address

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ADDR																														RSVD	

Bit Range	Default and Access	Field Name (ID): Description
31:2	0h RW	**Address (ADDR):** Lower 32 bits of the system specified message address, always DWORD aligned.
1:0	0h RO	Reserved.

24.9.1.23 Message Signaled Interrupt Message Data (MD)—Offset 88h

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 5

Default: 0000h

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DATA															

Bit Range	Default and Access	Field Name (ID): Description
15:0	0h RW	**Data (DATA):** This 16-bit field is programmed by system software if MSI is enabled. Its content is driven onto the lower word of the data bus of the MSI memory write transaction.

24.9.1.24 Port Mapping Register (MAP)—Offset 90h

Port Mapping Register

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD										SPD5	SPD4	SPD3	SPD2	SPD1	SPD0	RSVD													PCD			

Bit Range	Default and Access	Field Name (ID): Description
31:22	0h RO	Reserved.
21	0h RW/O	**SATA Port 5 Disable (SPD5):** Similar to SPD0 but for port 5. This bit is RO-one if ANY of the fuse/strap/GPIO condition below is true else this bit is RW-zero. PCIe/ SATA muxing for port 5 based on fuses, soft straps and GPIO does not select SATA. This bit is only applicable to project(s): that has port 5 physically.

Bit Range	Default and Access	Field Name (ID): Description
20	0h RW/O	**SATA Port 4 Disable (SPD4):** Similar to SPD0 but for port 4. This bit is RO-one if ANY of the fuse/strap/GPIO condition below is true else this bit is RW-zero. PCIe/ SATA muxing for port 4 based on fuses, soft straps and GPIO does not select SATA. This bit is only applicable to project(s): that has port 4 physically.
19	0h RW/O	**SATA Port 3 Disable (SPD3):** Similar to SPD0 but for port 3. This bit is RO-one if ANY of the fuse/strap/GPIO condition below is true else this bit is RW-zero. 1.Fuse FFSATA7 (disable port 2 and 3). 2.Fuse FFSATA8 (disable port 1 and 3). 3. PCIe/SATA muxing for port 3 based on fuses, soft straps and GPIO does not select SATA. This bit is only applicable to project(s): that has port 3 physically.
18	0h RW/O	**SATA Port 2 Disable (SPD2):** Similar to SPD0 but for port 2. This bit is RO-one if ANY of the fuse/strap/GPIO condition below is true else this bit is RW-zero. 1.Fuse FFSATA7 (disable port 2 and 3). 2.PCIe/SATA muxing for port 3 based on fuses, soft straps and GPIO does not select SATA. This bit is only applicable to project(s): that has port 3 physically.
17	0h RW/O	**SATA Port 1 Disable (SPD1):** Similar to SPD0 but for port 1. This bit is RO-one if ANY of the fuse/strap/GPIO condition below is true else this bit is RW-zero. 1.Fuse FFSATA8 (disable port 1 and 3). 2.PCIe/SATA muxing for port 3 based on fuses, soft straps and GPIO does not select SATA. This bit is only applicable to project(s): that has port 1 physically.
16	0h RW/O	**SATA Port 0 Disable (SPD0):** A 1 prevents the SATA port from being enabled via config PCS.PxE. Write of 1 to PCS.PxE has no effect when the corresponding SPD[x] bit is 1.In preventing a port(s) from being enabled, BIOS shall first configure MAP.SPDx. And only then BIOS configures the PCS.PxE. This field is not reset by FLR. This bit is only applicable to project(s): that has port 0 physically.
15:6	0h RO	Reserved.
5:0	0h RW	**Port Clock Disable (PCD):** When any of these bits is set to 1, the backbone clock driven to the associated port logic is gated and will not toggle. When this bit is cleared to 0, all clocks to the associated port logic will operate normally. Assignment of the bits is: Bit 5: Port 5, this bit is only applicable to project(s): that has port 5 physically; Bit 4: Port 4, this bit is only applicable to project(s): that has port 4 physically; Bit 3: Port 3, this bit is only applicable to project(s): that has port 3 physically; Bit 2: Port 2, this bit is only applicable to project(s): that has port 2 physically; Bit 1: Port 1, this bit is only applicable to project(s): that has port 1 physically; Bit 0: Port 0, this bit is only applicable to project(s): That has port 0 physically. If a particular port is not available, software shall set the corresponding bit to 1. Software can also set the corresponding bit(s) to 1 after disabling particular port(s). Software cannot set the PCD[port x]=1 if the corresponding config PCS.PxE=1 or AHCI MMIO GHC.PI[x]=1.

24.9.1.25 Port Control and Status (PCS)—Offset 94h

Port Control and Status. By default, the SATA ports are set (by hardware) to the disabled state (e.g., bits[5:0] == '0') as a result of an initial power on reset. When enabled by software, the ports can transition between the on, partial, and slumber states and can detect devices. When disabled, the port is in the off state and cannot detect any devices. This register is not reset by FLR.

Note: AHCI specific notes: If an AHCI-aware or RAID enabled operating system is being booted then system BIOS shall insure that all supported SATA ports are enabled prior to passing control to the OS. Once the AHCI aware OS is booted it becomes the enabling/ disabling policy owner for the individual SATA ports. This is accomplished by manipulating a port's PxSCTL.DET and PxCMD.SUD fields. Because an AHCI or RAID aware OS will typically not have knowledge of the PxE bits and because the PxE bits act as master on/off switches for the ports, pre-boot software must insure that these bits are set to '1' prior to booting the OS, regardless as to whether or not a device is currently on the port.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD										P5P	P4P	P3P	P2P	P1P	P0P	RSVD										P5E	P4E	P3E	P2E	P1E	P0E

Bit Range	Default and Access	Field Name (ID): Description
31:22	0h RO	Reserved.
21	0h RO	**Port 5 Present (P5P):** Same as P0P, except for port 5. This bit is only applicable to project(s): That has port 5 physically.
20	0h RO	**Port 4 Present (P4P):** Same as P0P, except for port 4. This bit is only applicable to project(s): That has port 4 physically.
19	0h RO	**Port 3 Present (P3P):** Same as P0P, except for port 3. This bit is only applicable to project(s): That has port 3 physically.
18	0h RO	**Port 2 Present (P2P):** Same as P0P, except for port 2. This bit is only applicable to project(s): That has port 2 physically.
17	0h RO	**Port 1 Present (P1P):** Same as P0P, except for port 1. This bit is only applicable to project(s): That has port 1 physically.
16	0h RO	**Port 0 Present (P0P):** When set, the SATA controller has detected the presence of a device on port 0. It may change at any time. Clearing P0E bit leads to clearing of this bit after implementation delay. ***Note:*** For system software that intends to clear all PCS.PxE bits that are previously 1 to 0 and then to 1 again in two consecutive write cycles, software shall poll on this bit being 0 before setting P0E bit to 1. This bit is only applicable to project(s): That has port 0 physically.
15:6	0h RO	Reserved.
5	0h RW	**Port 5 Enabled (P5E):** When MAP.SPD[5] is 1, this bit is reserved and is read-only 0. Otherwise if none of the above is true, this field is RW and the definition of this bit is the same as P0E, except for port 5. This bit takes precedence over P5CMD.SUD. This bit is only applicable to project(s): That has port 5 physically.
4	0h RW	**Port 4 Enabled (P4E):** When MAP.SPD[4] is 1, this bit is reserved and is read-only 0. Otherwise if none of the above is true, this bit is RW and same as P0E but for port 4 and takes precedence over P4CMD.SUD. This bit is only applicable to project(s): That has port 4 physically.
3	0h RW	**Port 3 Enabled (P3E):** When MAP.SPD[3] is 1, this bit is reserved and is read-only 0. Otherwise if none of the above is true, this bit is RW and same as P0E but for port 3 and takes precedence over P3CMD.SUD. This bit is only applicable to project(s): That has port 3 physically.

Bit Range	Default and Access	Field Name (ID): Description
2	0h RW	**Port 2 Enabled (P2E):** When MAP.SPD[2] is 1, this bit is reserved and is read-only 0. Otherwise if none of the above is true, this bit is RW and same as P0E but for port 2 and takes precedence over P2CMD.SUD. This bit is only applicable to project(s): That has port 2 physically.
1	0h RW	**Port 1 Enabled (P1E):** When MAP.SPD[1] is 1, this bit is reserved and read-only 0. Otherwise if none of the above is true, this bit is RW and same as P0E but for port 1 and takes precedence over P1CMD.SUD. This bit is only applicable to project(s): That has port 1 physically.
0	0h RW	**Port 0 Enabled (P0E):** When MAP.SPD[0] is 1, this bit is reserved and read-only 0. When set, the port is enabled. When cleared, the port is disabled. When enabled, the port can transition between the on, partial, and slumber states and can detect devices. When disabled, the port is in the off state and cannot detect any devices. This bit takes precedence over P0CMD.SUD. ***Note:*** The recommendation for software code that intends to clear all PCS.PxE bits that are previously 1 to 0 and then to 1 again immediately shall refer to the polling requirement as described in P0P register bit. At any time that BIOS or software is clearing PCS.PxE from 1 to 0, due to time needed for port staggering hardware process (up to 6 ports) to complete, BIOS and software shall delay the write to set the MAP.PCD register by 1.4us. This bit is only applicable to project(s): That has port 0 physically. This bit is only applicable to project(s): That has port 0 physically.

24.9.1.26 SATA General Configuration (SATAGC)—Offset 9Ch

Type: CFG Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
REGLOCK	RSVD															SMS	DPPEE	WRRSELMPS			CPEE	SCFD	URRE	URD	AIE	DEVIDSEL	FLRCSSEL	MSS		ASSEL		

Bit Range	Default and Access	Field Name (ID): Description
31	0h RW/O	**Register Lock (REGLOCK):** BIOS can set this bit to 1 to lock the following registers with RW/L attribute: CAP,CP, MID.NEXT, PIE.NEXT, SATACR0.NEXT. Once locked the register attribute of above list changes from RW/L to RO holding the existing value. BIOS is required to program this field to 1 prior to handle off to OS. If BIOS needs the SATA host controller to change operation a few times (i.e., changing CC.SCC mode) and need different capability structures for each specific operation mode, BIOS need not activate the lock until BIOS is ready to hand off to OS. BIOS may need to separate write access to this byte offset (x9Fh) from write to the lower 3-byte of the dword (x9C-9Eh) if there is a need to program the lower 3-byte dword location early during boot process. This field is not reset by FLR.
30:17	0h RO	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
16	0h RW	**SATA Mode Select (SMS):** Software (SW) programs these bits to control the mode in which the SATA HBA should operate: 0b = AHCI mode 1b = RAID mode ***Notes:*** SW shall not manipulate SATAGC.SMS during runtime operation; i.e., the OS will not do this. The BIOS may choose to switch from one mode to another during POST; AHCI mode may be selected when RAID feature is enabled by fuse; RAID mode may only be selected when FFSATA5 and FFSATA3 (concatenated value not indicating No RAID); This register field is not reset by FLR.
15	0h RW	**Data Phase Parity Error Eenable (DPPEE):** When 1, IOSF data phase parity error handling is enabled. When 0, the data phase parity error handling is disabled
14:12	0h RW	**Write Request_Size Select/Max_Payload_Size (WRRSELMPS):** These two bits select the max write request size that SATA host controller will initiate for DMA write to memory. SATA host controller will internally break up larger write request based on these bits. The request is address-aligned to the selected size. Defined encodings for this field are: 000b = 128 address aligned bytes max payload size; 111b = 64 address aligned bytes max payload size. All other values are reserved for SATA host controller. This field is not reset by FLR.
11	0h RW	**Command Parity Error Enable (CPEE):** When 1, command parity error handing is enable. When 0 the command parity error handling is disable
10	0h RW	**SATA Controller Function Disable (SCFD):** BIOS program this bit to 1 to disable the SATA Controller function. When 0, SATA Controller function is enabled. When disable, SATA Host Controller will not claimed the register access targeting its Configuration Space. In IOSF primary Fabric Decode scheme, it's expected BIOS also program the corresponding bit used by the Fabric Decoder accordingly hence both SATA SIP and Fabric Decoder are in sync, and BIOS need to program this bit before programming the one in Fabric Decoder. Once this bit is set, BIOS is not able to revert it back to Function Enable until next round of platform reset.
9	0h RW	**Unsupported Request Reporting Enable (URRE):** If set to 1 by software, it allows reporting of an Unsupported Request as a system error. If both URRE and PCI configuration SERR# Enable registers are set to a 1, then the agent must set the Signaled System Error bit in the PCI Status register and send a DO_SERR message in IOSF-SB interface.
8	0h RW/1C	**Unsupported Request Detected (URD):** Set to 1 by hardware upon detecting an Unsupported Request on IOSF Primary interface that is not considered Advisory Non-Fatal. Cleared to 0 by SW.
7	0h RW/O	**Alternate ID Enable (AIE):** When programmed to 0, HW will report the following device id's: 2822h for desktop or 282Ah for mobile. When programmed to a 1, HW will not report these device IDs. ***Notes:*** Programming this bit to a 1 will prevent the Windows in-box version of the Intel AHCI driver from loading on the platform - will require that the user perform an 'F6' install of the Intel driver that is appropriate for the reported DID. This field is applicable when the AHCI is configured for RAID mode of operation. It has no impact for AHCI and IDE modes of operation. BIOS is recommended to program this bit prior to programming the MAP.SMS field to reflect RAID. This field is reset by PLTRST# and BIOS is required to reprogram the value (either 0 or 1) after resuming from S3, S4 or S5.
6	0h RW/O	**AIE0 DevID Selection (DEVIDSEL):** This register allows BIOS to select Device ID when AIE=0 and Server Feature (SATA AIE DEVIDSEL) Disable Fuse =0. This bit only has effect in Desktop SKU. In Mobile SKU this bit has no effect at all. Refer to config register offset 09h PI for usage. ***Notes:*** WBG BIOS is required to program this field to 1 together with the write to the AIE bit in a single configuration write cycle. LPT BIOS is required to program this bit to 0 together with the write to the AIE bit in a single configuration write cycle. When Server Feature (SATA AIE DEVIDSEL) Disable Fuse is programmed to 1, this disables the writeability of this DEVIDSEL register bit, and becomes RO with a value of 0, which only allows a choice of 2822h. This field is not reset by FLR.
5	0h RW/O	**FLR Capability Selection (FLRCSSEL):** This allows the FLR Capability to be bypassed. Refer to config offset B0h. BIOS is required to program this bit to 1 and config offset A8h SATACR0.NEXT to 00h. This field is not reset by FLR.

Bit Range	Default and Access	Field Name (ID): Description
4:3	0h RW/O	**MXTBA Size Select (MSS):** These 2 bits select the size of the Memory space for the MSI-X Table defined in BAR 0 (Configuration space offset 10h). MSI-X Table Memory space size is 32k when MSS[1:0]=00, 16k when MSS[1:0]=01, 8k when MSS[1:0]=10. This field is not reset by FLR.
2:0	0h RW/O	**ABAR Size Select (ASSEL):** These 3 bits select the size of the Memory space for the ABAR in BAR 5 (Configuration space offset 24h). ABAR Memory space size is 2k when ASSEL[2:0]=000, 16k when ASSEL[2:0]=001, 32k when ASSEL[2:0]=010, 64k when ASSEL[2:0]=011, 128k when ASSEL[2:0]=100, 256k when ASSEL[2:0]=101, 512k when ASSEL[2:0]=110. This field is not reset by FLR.

24.9.1.27 SATA Initialization Register Index (SIRI)—Offset A0h

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 5

Default: 00h

7			4				0
0	0	0	0	0	0	0	0
IDX						RSVD	

Bit Range	Default and Access	Field Name (ID): Description
7:2	0h RW	**Index (IDX):** 6-bit index pointer into the 256-byte space. Data is written into the SIRD register and read from the SIRD register. This point to a DWord register. The byte enables on the SIRD register affect what will be written. Refer to SATA Initialization Register section for detail of the register space.
1:0	0h RO	Reserved.

24.9.1.28 SATA Initialization Register Data (SIRD)—Offset A4h

Type: CFG Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DTA																															

Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Data (DTA):** 32-bit data value that is written to the register pointed to by SIRI, or read from the register pointed to by SIRI.

24.9.1.29 Serial ATA Capability Register 0 (SATACR0)—Offset A8h

Note that the SATACR0.NEXT is not changed from RO to become RWO because there is an existing method (SATAGC.FLRCSSEL bit) to bypass the FLR Capability structure. And FLR Capability ID.NEXT is already indicating end of capability structure, it does not need change to be RWO.

Type: CFG Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 100012h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
RSVD								MAJREV				MINREV				NEXT								CAP							

Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23:20	1h RO	**Major Revision (MAJREV):** Major revision number of the SATA Capability Pointer implemented.
19:16	0h RO	**Minor Revision (MINREV):** Minor revision number of the SATA Capability Pointer implemented.
15:8	0h RW/L	**Next Capability Pointer (NEXT):** 00h indicating the final item in the Capability List. The RW/L register attribute allows for flexibility in determining the capability structure available in this PCI function. Refer to SATAGC.REGLOCK description in order to lock the register to become RO. This register is not reset by FLR.
7:0	12h RO	**Capability ID (CAP):** The value of 12h has been assigned by the PCI SIG to designate the SATA Capability pointer.

24.9.1.30 Serial ATA Capability Register 1 (SATACR1)—Offset ACh

Type: CFG Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 0000000048h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0
RSVD																BAROFST												BARLOC			

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:4	4h RO	**BAR Offset (BAROFST):** Indicates the offset into the BAR where the AHCI Index/ Data pair are located (in Dword granularity). The Index and Data I/O registers are located at offset 10h within the I/O space defined by LBAR (BAR4). A value of 004h indicates offset 10h. 000h = 0h offset; 001h = 4h offset; 002h = 8h offset; 003h = Ch offset; 004h = 10h offset; ...; FFFh = 3FFFh offset (max 16 KB)
3:0	8h RO	**BAR Location (BARLOC):** Indicates the absolute PCI Configuration Register address of the BAR containing the Index/Data pair (in Dword granularity). The Index and Data I/O registers reside within the space defined by LBAR (BAR4) in the SATA controller. A value of 8h indicates offset 20h, which is LBAR (BAR4). 0000 - 0011b = reserved; 0100b = 10h => BAR0; 0101b = 14h => BAR1; 0110b = 18h => BAR2; 0111b = 1Ch => BAR3; 1000b = 20h => AIDPBA; 1001b = 24h => BAR5; 1010 - 1110b = reserved; 1111b = Index/Data pair in PCI Configuration space which is not supported.

24.9.1.31 FLR Capability ID (SP)—Offset B0h

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 5

Default: 00xxh

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
NCP								CID							

Bit Range	Default and Access	Field Name (ID): Description
15:8	00h RO	**Next Capability Pointer (NCP):** 00h indicating the final item in the Capability List.
7:0	RO	**Capability ID(CID)**: (to BD determined if 00h, or 13H)

24.9.1.32 FLR Capability Length and Version —Offset B2h

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 5

Default: 0306h

15			12				8				4				0
0	0	0	0	0	0	1	1	0	0	0	0	0	1	1	0
NCP								CID							

Bit Range	Default and Access	Field Name (ID): Description
15:10	00h RO	**Reserved (RSVD):**
9	1b RWO	**FLR Capability ID(FLRCAP)**: When set to 1b indicates that Function Level Reset is supported.

Bit Range	Default and Access	Field Name (ID): Description
8	1b RWO	TXP Capability (TXPCAP): A '1' in this bit indicates support for the Transactions pending (TXP) bit.
7:0	06h RO	Capability Length (CAPLEN): This field indicates the # of bytes as required by the PCI Express Specification and is 06h for FLR Capability

24.9.1.33 FLR Control —Offset B4h

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 5

Default: 0000h

15			12				8				4				0
0	0	0	0	0	0	1	1	0	0	0	0	0	1	1	0
RSVD							TXP	RSVD							IFLR

Bit Range	Default and Access	Field Name (ID): Description
15:9	00h RO	**Reserved (RSVD):**
8	0b RO	**Transactions Pending (TXP)**: When set to 1b indicates that the controller has issued non-posted request which has not completed. A 0b indicates that completions for all non-posted requests have been received by the controller
7:1	00h RO	**Reserved (RSVD).**
0	0b RW	**Initial FLR (IFLR):** A write of 1b initiates an FLR transition.

24.9.1.34 Scratch Pad (SP)—Offset C0h

Type: CFG Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DT																															

Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Data (DT):** This is a read/write register that is available for software to use. No hardware action is taken on this register.

24.9.1.35 MSI-X Identifiers (MXID)—Offset D0h

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 5

Default: 0011h

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
NEXT								CID							

Bit Range	Default and Access	Field Name (ID): Description
15:8	0h RW/L	**Next Pointer (NEXT):** Indicates the next item in the list. This may be other capability pointers or it may be the last item in the list.The RW/L register attribute allows for flexibility in determining the capability structure available in this PCI function. Refer to SATAGC.REGLOCK description in order to lock the register to become RO. This register is not reset by FLR.
7:0	11h RO	**Capability ID (CID):** Capabilities ID indicates this is an MSI-X capability.

24.9.1.36 MSI-X Message Control (MXC)—Offset D2h

MSI-X Message Control

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 5

Default: 0000h

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MXE	FM	RSVD								TS					

Bit Range	Default and Access	Field Name (ID): Description
15	0h RW	**MSI-X Enable (MXE):** If set to '1' and the MSI Enable bit in the MSI Message Control register is cleared to '0', the function is permitted to use MSI-X to request service and is prohibited from using its INTx# pin (if implemented). If cleared to '0', the function is prohibited from using MSI-X to request service.
14	0h RW	**Function Mask (FM):** If set to '1', all of the vectors associated with the function are masked, regardless of their per vector Mask bit states. If cleared to '0', each vector's Mask bit determines whether the vector is masked or not. Setting or clearing the MSI-X Function Mask bit has no effect on the state of the per vector Mask bits.
13:11	0h RO	Reserved.
10:0	0h RO	**Table Size (TS):** This value indicates the size of the MSI-X Table as the value n, which is encoded as n - 1. For example, a returned value of 3h corresponds to a table size of 4.

24.9.1.37 MSI-X Table Offset/Table BIR (MXT)—Offset D4h

Type: CFG Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 00000000h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
TO																													TBIR		

Bit Range	Default and Access	Field Name (ID): Description
31:3	0h RO	**Table Offset (TO):** Used as an offset from the address contained by one of the function's Base Address registers to point to the base of the MSI-X Table. The lower three Table BIR bits are masked off (cleared to 000b) by system software to form a 32-bit Qword-aligned offset.
2:0	0h RO	**Table BIR (TBIR):** This field indicates which one of a function's Base Address registers, located beginning at 10h in Configuration Space, is used to map the function's MSI-X Table into system memory. A read-only value of '0' means 10h.

24.9.1.38 MSI-X PBA Offset/PBA BIR (MXP)—Offset D8h

Type: CFG Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
PBAO																													PBIR		

Bit Range	Default and Access	Field Name (ID): Description
31:3	0h RO	**PBA Offset (PBAO):** Used as an offset from the address contained by one of the function's Base Address registers to point to the base of the MSI-X PBA. The lower three PBA BIR bits are masked off (cleared to 000b) by software to form a 32-bit Qword-aligned offset.
2:0	1h RO	**PBA BIR (PBIR):** This field indicates which one of a function's Base Address registers, located beginning at 10h in Configuration Space, is used to map the function's MSI-X PBA into system memory. A read-only value of '1' means 14h.

24.9.1.39 BIST FIS Control/Status (BFCS)—Offset E0h

Type: CFG Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																P5BFI	P4BFI	P3BFI	P2BFI	BFS	BFF	P1BFI	P0BFI	BFP					RSVD			

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15	0h RW	**Port 5 BIST FIS Initiate (P5BFI):** When a rising edge is detected on this bit, the SATA controller will initiate a BIST FIS to the device on port 5, using the parameters specified in this register and BFTD1 and BFTD2. The BIST FIS will only be initiated if a device is present and not in the partial or slumber states. After a BIST FIS is successfully completed, software must disable and re-enable the PCS.P5E prior to attempting additional BIST FISes or to return the SATA controller to a normal operational mode. If the BIST FIS fails, as indicated by BFF in this register, software can clear and then set this bit to initiate another BIST FIS. This bit is only applicable to project(s): That has port 5 physically.
14	0h RW	**Port 4 BIST FIS Initiate (P4BFI):** When a rising edge is detected on this bit, the SATA controller will initiate a BIST FIS to the device on port 4, using the parameters specified in this register and BFTD1 and BFTD2. The BIST FIS will only be initiated if a device is present and not in the partial or slumber states. After a BIST FIS is successfully completed, software must disable and re-enable PCS.P4E prior to attempting additional BIST FISes or to return the SATA controller to a normal operational mode. If the BIST FIS fails, as indicated by BFF in this register, software can clear and then set this bit to initiate another BIST FIS. This bit is only applicable to project(s): That has port 4 physically.
13	0h RW	**Port 3 BIST FIS Initiate (P3BFI):** When a rising edge is detected on this bit, the SATA controller will initiate a BIST FIS to the device on port 3, using the parameters specified in this register and BFTD1 and BFTD2. The BIST FIS will only be initiated if a device is present and not in the partial or slumber states. After a BIST FIS is successfully completed, software must disable and re-enable the PCS.P3E prior to attempting additional BIST FISes or to return the SATA controller to a normal operational mode. If the BIST FIS fails, as indicated by BFF in this register, software can clear and then set this bit to initiate another BIST FIS. This bit is only applicable to project(s): That has port 3 physically.
12	0h RW	**Port 2 BIST FIS Initiate (P2BFI):** When a rising edge is detected on this bit, the SATA controller will initiate a BIST FIS to the device on port 2, using the parameters specified in this register and BFTD1 and BFTD2. The BIST FIS will only be initiated if a device is present and not in the partial or slumber states. After a BIST FIS is successfully completed, software must disable and re-enable PCS.P2E prior to attempting additional BIST FISes or to return the SATA controller to a normal operational mode. If the BIST FIS fails, as indicated by BFF in this register, software can clear and then set this bit to initiate another BIST FIS. This bit is only applicable to project(s): That has port 2 physically.
11	0h RW/1C	**BIST FIS Successful (BFS):** This bit is set any time a BIST FIS transmitted by the SATA controller receives an R_OK completion status from the device.
10	0h RW/1C	**BIST FIS Failed (BFF):** This bit is set any time that a BIST FIS transmitted by the SATA controller receives an R_ERR completion status from the device.
9	0h RW	**Port 1 BIST FIS Initiate (P1BFI):** When a rising edge is detected on this bit, the SATA controller will initiate a BIST FIS to the device on port 1, using the parameters specified in this register and BFTD1 and BFTD2. The BIST FIS will only be initiated if a device is present and not in the partial or slumber states. After a BIST FIS is successfully completed, software must disable and re-enable PCS.P1E prior to attempting additional BIST FISes or to return the SATA controller to a normal operational mode. If the BIST FIS fails, as indicated by BFF in this register, software can clear and then set this bit to initiate another BIST FIS. This bit is only applicable to project(s): That has port 1 physically.

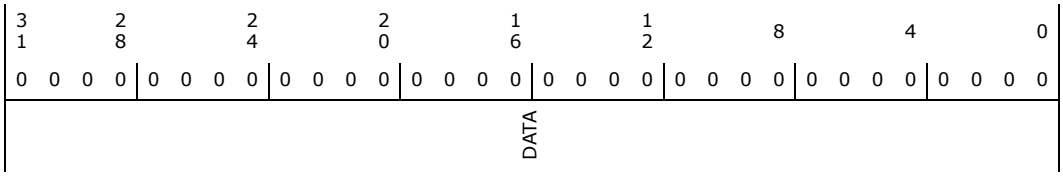
Bit Range	Default and Access	Field Name (ID): Description
8	0h RW	**Port 0 BIST FIS Initiate (P0BFI):** When a rising edge is detected on this bit, the SATA controller will initiate a BIST FIS to the device on port 0, using the parameters specified in this register and BFTD1 and BFTD2. The BIST FIS will only be initiated if a device is present and not in the partial or slumber states. After a BIST FIS is successfully completed, software must disable and re-enable PCS.P0E prior to attempting additional BIST FISes or to return the SATA controller to a normal operational mode. If the BIST FIS fails, as indicated by BFF in this register, software can clear and then set this bit to initiate another BIST FIS. This bit is only applicable to project(s): That has port 0 physically.
7:2	0h RW	**BIST FIS Parameters (BFP):** These bits form the contents of the upper 6 bits of the BIST FIS Pattern Definition in the BIST FIS transmitted by the SATA controller. This field is not port specific - its contents will be used for any BIST FIS initiated on the SATA controller. The specific bit definitions are: Bit 7 (T) Far End Transmit mode; bit 6 (A) Align Bypass mode; bit 5 (S) Bypass Scrambling; bit 4 (L) Far End Retimed Loopback; bit 3 (F) Far End Analog Loopback; bit 2 (P) Primitive bit for use with Transmit mode.
1:0	0h RO	Reserved.

24.9.1.40 BIST FIS Transmit Data 1 (BFTD1)—Offset E4h

Type: CFG Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 00000000h



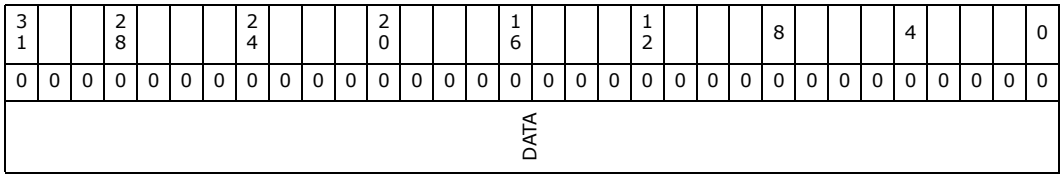
Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Data (DATA):** The data programmed into this register will form the contents of the second DW of any BIST FIS initiated by the SATA controller. This register is not port specific - its contents will be used for BIST FIS initiated on any port. Although the 2nd and 3rd DWs of the BIST FIS are only meaningful when the T bit of the BIST FIS is set to indicate Far-End Transmit mode, this register's contents will be transmitted as the BIST FIS 2nd DW regardless of whether or not the T bit is indicated in the BFCS register.

24.9.1.41 BIST FIS Transmit Data 2 (BFTD2)—Offset E8h

Type: CFG Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 00000000h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Data (DATA):** The data programmed into this register will form the contents of the third DW of any BIST FIS initiated by the SATA controller. This register is not port specific - its contents will be used for BIST FIS initiated on any port. Although the 2nd and 3rd DWs of the BIST FIS are only meaningful when the T bit of the BIST FIS is set to indicate Far-End Transmit mode, this register's contents will be transmitted as the BIST FIS 3rd DW regardless of whether or not the T bit is set in the BFCS register.

24.9.2 SSATA ABAR Registers Summary

Table 24-7. Summary of SSATA ABAR Registers (Sheet 1 of 3)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	HBA Capabilities (GHC_CAP)—Offset 0h	FF36FF05h
4h	7h	Global HBA Control (GHC)—Offset 4h	80000000h
8h	Bh	Interrupt Status Register (IS)—Offset 8h	0h
Ch	Fh	Ports Implemented (GHC_PI)—Offset Ch	0h
10h	13h	AHCI Version (VS)—Offset 10h	10300h
1Ch	1Fh	Enclosure Management Location (EM_LOC)—Offset 1Ch	1600002h
20h	23h	Enclosure Management Control (EM_CTL)—Offset 20h	7010000h
24h	27h	HBA Capabilities Extended (GHC_CAP2)—Offset 24h	3Ch
A0h	A3h	Vendor Specific (VSP)—Offset A0h	48h
A4h	A7h	Vendor Specific Capabilities Register (VS_CAP)—Offset A4h	1002DEh
C0h	C3h	RAID Platform ID (RPID)—Offset C0h	311C02h
C4h	C5h	Premium Feature Block (PFB)—Offset C4h	0h
C8h	C9h	SW Feature Mask (SFM)—Offset C8h	3Fh
100h	103h	Port 0 Command List Base Address (P0CLB)—Offset 100h	0h
104h	107h	Port 0 Command List Base Address Upper 32-bits (P0CLBU)—Offset 104h	0h
108h	10Bh	Port 0 FIS Base Address (P0FB)—Offset 108h	0h
10Ch	10Fh	Port 0 FIS Base Address Upper 32-Bits (P0FBU)—Offset 10Ch	0h
110h	113h	Port 0 Interrupt Status (P0IS)—Offset 110h	0h
114h	117h	Port 0 Interrupt Enable (P0IE)—Offset 114h	0h
118h	11Bh	Port 0 Command (P0CMD)—Offset 118h	4h
120h	123h	Port 0 Task File Data (P0TFD)—Offset 120h	9h
124h	127h	Port 0 Signature (P0SIG)—Offset 124h	FFFFFFFFh
128h	12Bh	Port 0 Serial ATA Status (P0SSTS)—Offset 128h	0h
12Ch	12Fh	Port 0 Serial ATA Control (P0SCTL)—Offset 12Ch	0h
130h	133h	Port 0 Serial ATA Error (P0SERR)—Offset 130h	0h
134h	137h	Port 0 Serial ATA Active (P0SACT)—Offset 134h	0h
138h	13Bh	Port 0 Command Issue (P0CI)—Offset 138h	0h
13Ch	13Fh	Port 0 SNotification (P0SNTF)—Offset 13Ch	0h
144h	147h	Port 0 Device Sleep (P0DEVSLP)—Offset 144h	1E022852h
180h	183h	Port 1 Command List Base Address (P1CLB)—Offset 180h	0h
184h	187h	Port 1 Command List Base Address Upper 32-bits (P1CLBU)—Offset 184h	0h
188h	18Bh	Port 1 FIS Base Address (P1FB)—Offset 188h	0h

Table 24-7. Summary of SSATA ABAR Registers (Sheet 2 of 3)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
18Ch	18Fh	Port 1 FIS Base Address Upper 32-bits (P1FBU)—Offset 18Ch	0h
190h	193h	Port 1 Interrupt Status (P1IS)—Offset 190h	0h
194h	197h	Port 1 Interrupt Enable (P1IE)—Offset 194h	0h
198h	19Bh	Port 1 Command (P1CMD)—Offset 198h	0h
1A0h	1A3h	Port 1 Task File Data (P1TFD)—Offset 1A0h	0h
1A4h	1A7h	Port 1 Signature (P1SIG)—Offset 1A4h	0h
1A8h	1ABh	Port 1 Serial ATA Status (P1SSTS)—Offset 1A8h	0h
1ACh	1AFh	Port 1 Serial ATA Control (P1SCTL)—Offset 1ACh	0h
1B0h	1B3h	Port 1 Serial ATA Error (P1SERR)—Offset 1B0h	0h
1B4h	1B7h	Port 1 Serial ATA Active (P1SACT)—Offset 1B4h	0h
1B8h	1BBh	Port 1 Command Issue (P1CI)—Offset 1B8h	0h
1BCh	1BFh	Port 1 SNotification (P1SNTF)—Offset 1BCh	0h
1C4h	1C7h	Port 1 Device Sleep (P1DEVSLP)—Offset 1C4h	0h
200h	203h	Port 2 Command List Base Address (P2CLB)—Offset 200h	0h
204h	207h	Port 2 Command List Base Address Upper 32-bits (P2CLBU)—Offset 204h	0h
208h	20Bh	Port 2 FIS Base Address (P2FB)—Offset 208h	0h
20Ch	20Fh	Port 2 FIS Base Address Upper 32-bits (P2FBU)—Offset 20Ch	0h
210h	213h	Port 2 Interrupt Status (P2IS)—Offset 210h	0h
214h	217h	Port 2 Interrupt Enable (P2IE)—Offset 214h	0h
218h	21Bh	Port 2 Command (P2CMD)—Offset 218h	0h
220h	223h	Port 2 Task File Data (P2TFD)—Offset 220h	0h
224h	227h	Port 2 Signature (P2SIG)—Offset 224h	0h
228h	22Bh	Port 2 Serial ATA Status (P2SSTS)—Offset 228h	0h
22Ch	22Fh	Port 2 Serial ATA Control (P2SCTL)—Offset 22Ch	0h
230h	233h	Port 2 Serial ATA Error (P2SERR)—Offset 230h	0h
234h	237h	Port 2 Serial ATA Active (P2SACT)—Offset 234h	0h
238h	23Bh	Port 2 Command Issue (P2CI)—Offset 238h	0h
23Ch	23Fh	Port 2 SNotification (P2SNTF)—Offset 23Ch	0h
244h	247h	Port 2 Device Sleep (P2DEVSLP)—Offset 244h	0h
280h	283h	Port 3 Command List Base Address (P3CLB)—Offset 280h	0h
284h	287h	Port 3 Command List Base Address Upper 32-bits (P3CLBU)—Offset 284h	0h
288h	28Bh	Port 3 FIS Base Address (P3FB)—Offset 288h	0h
28Ch	28Fh	Port 3 FIS Base Address Upper 32-bits (P3FBU)—Offset 28Ch	0h
290h	293h	Port 3 Interrupt Status (P3IS)—Offset 290h	0h
294h	297h	Port 3 Interrupt Enable (P3IE)—Offset 294h	0h
298h	29Bh	Port 3 Command (P3CMD)—Offset 298h	0h
2A0h	2A3h	Port 3 Task File Data (P3TFD)—Offset 2A0h	0h
2A4h	2A7h	Port 3 Signature (P3SIG)—Offset 2A4h	0h
2A8h	2ABh	Port 3 Serial ATA Status (P3SSTS)—Offset 2A8h	0h
2ACh	2AFh	Port 3 Serial ATA Control (P3SCTL)—Offset 2ACh	0h
2B0h	2B3h	Port 3 Serial ATA Error (P3SERR)—Offset 2B0h	0h

Table 24-7. Summary of SSATA ABAR Registers (Sheet 3 of 3)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
2B4h	2B7h	Port 3 Serial ATA Active (P3SACT)—Offset 2B4h	0h
2B8h	2BBh	Port 3 Commands Issued (P3CI)—Offset 2B8h	0h
2BCh	2BFh	Port 3 SNotification (P3SNTF)—Offset 2BCh	0h
2C4h	2C7h	Port 3 Device Sleep (P3DEVSLP)—Offset 2C4h	0h
300h	303h	Port 4 Command List Base Address (P4CLB)—Offset 300h	0h
304h	307h	Port 4 Command List Base Address Upper 32-bits (P4CLBU)—Offset 304h	0h
308h	30Bh	Port 4 FIS Base Address (P4FB)—Offset 308h	0h
30Ch	30Fh	Port 4 FIS Base Address Upper 32-bits (P4FBU)—Offset 30Ch	0h
310h	313h	Port 4 Interrupt Status (P4IS)—Offset 310h	0h
314h	317h	Port 4 Interrupt Enable (P4IE)—Offset 314h	0h
318h	31Bh	Port 4 Command (P4CMD)—Offset 318h	0h
320h	323h	Port 4 Task File Data (P4TFD)—Offset 320h	0h
324h	327h	Port 4 Signature (P4SIG)—Offset 324h	0h
328h	32Bh	Port 4 Serial ATA Status (P4SSTS)—Offset 328h	0h
32Ch	32Fh	Port 4 Serial ATA Control (P4SCTL)—Offset 32Ch	0h
330h	333h	Port 4 Serial ATA Error (P4SERR)—Offset 330h	0h
334h	337h	Port 4 Serial ATA Active (P4SACT)—Offset 334h	0h
338h	33Bh	Port 4 Commands Issued (P4CI)—Offset 338h	0h
33Ch	33Fh	Port 4 SNotification (P4SNTF)—Offset 33Ch	0h
344h	347h	Port 4 Device Sleep (P4DEVSLP)—Offset 344h	0h
380h	383h	Port 5 Command List Base Address (P5CLB)—Offset 380h	0h
384h	387h	Port 5 Command List Base Address Upper 32-bits (P5CLBU)—Offset 384h	0h
388h	38Bh	Port 5 FIS Base Address (P5FB)—Offset 388h	0h
38Ch	38Fh	Port 5 FIS Base Address Upper 32-bits (P5FBU)—Offset 38Ch	0h
390h	393h	Port 5 Interrupt Status (P5IS)—Offset 390h	0h
394h	397h	Port 5 Interrupt Enable (P5IE)—Offset 394h	0h
398h	39Bh	Port 5 Command (P5CMD)—Offset 398h	0h
3A0h	3A3h	Port 5 Task File Data (P5TFD)—Offset 3A0h	0h
3A4h	3A7h	Port 5 Signature (P5SIG)—Offset 3A4h	0h
3A8h	3ABh	Port 5 Serial ATA Status (P5SSTS)—Offset 3A8h	0h
3ACh	3AFh	Port 5 Serial ATA Control (P5SCTL)—Offset 3ACh	0h
3B0h	3B3h	Port 5 Serial ATA Error (P5SERR)—Offset 3B0h	0h
3B4h	3B7h	Port 5 Serial ATA Active (P5SACT)—Offset 3B4h	0h
3B8h	3BBh	Port 5 Commands Issued (P5CI)—Offset 3B8h	0h
3BCh	3BFh	Port 5 SNotification (P5SNTF)—Offset 3BCh	0h
3C4h	3C7h	Port 5 Device Sleep (P5DEVSLP)—Offset 3C4h	0h
580h	583h	Enclosure Management Message Format (EM_MF)—Offset 580h	0h
584h	587h	Enclosure Management LED (EM_LED)—Offset 584h	0h

24.9.2.1 HBA Capabilities (GHC_CAP)—Offset 0h

HBA Capabilities. This register indicates basic capabilities of the HBA to driver software. The RWO bits in this register are only cleared upon PLTRST#. This register is not reset by FLR.

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: FF36FF05h

31			28				24				20				16				12				8				4				0
1	1	1	1	1	1	1	1	0	0	1	1	0	1	1	0	1	1	1	1	1	1	1	0	0	0	0	0	1	0	1
S64A	SCQA	SSNTF	SMPS	SSS	SALP	SAL	SCLO	ISS				RSVD	SAM	SPM	FBSS	PMD	SSC	PSC	NCS				CCCS	EMS	SXS	NP						

Bit Range	Default and Access	Field Name (ID): Description
31	1h RW/O	**Supports 64-bit Addressing (S64A):** Indicates the S-ATA controller can access 64-bit data structures. The 32-bit upper bits of the port DMA Descriptor, the PRD Base, and each PRD entry are read/write.
30	1h RW/O	**Supports Native Command Queuing Acceleration (SCQA):** Indicates the SATA controller supports Serial-ATA Native Command Queueing. The HBA will handle DMA Setup FISes in hardware, including support for auto-activate optimization through the FIS.
29	1h RW/O	**Supports SNotification Register (SSNTF):** When set to 1, indicates that the HBA supports the PxSNTF (SNotification) register and its associated functionality. When cleared to 0., the HBA does not support the PxSNTF (SNotification) register and its associated functionality.
28	1h RW/O	**Supports Mechanical Presence Switch (SMPS):** When set to 1, the HBA supports mechanical presence switches on its ports for use in hot plug operations. When cleared to 0, this function is not supported. This value is loaded by the BIOS prior to OS initialization.
27	1h RW/O	**Supports Staggered Spin-up (SSS):** Indicates whether the S-ATA controller supports staggered spin-up on its ports, for use in balancing power spikes. This value is loaded by platform BIOS prior to OS initialization.
26	1h RW/O	**Supports Aggressive Link Power Management (SALP):** Indicates the S-ATA controller supports auto-generating link requests to the partial or slumber states when there are no commands to process. When cleared to 0, software shall treat the PxCMD.ALPE and PxCMD.ASP bits as reserved.
25	1h RW/O	**Supports Activity LED (SAL):** Indicates the S-ATA controller supports a single output pin (SATALED#) which indicates activity.
24	1h RW/O	**Supports Command List Override (SCLO):** When set to 1, indicates that the HBA supports the PxCMD.CLO bit and it's associated function. When cleared to 0., The HBA is not capable of clearing the BSY and DRQ bits in the Status register in order to issue a software reset if these bits are still set from a previous operation.
23:20	3h RW/O	**Interface Speed Support (ISS):** Indicates the maximum speed the S-ATA controller can support on its ports. These encodings match the system software programmable PxSCTL.DET.SPD field. 0000 = Reserved 0001 = Gen 1 (1.5 Gbps) 0010 = Gen 2 (3 Gbps) 0011 = Gen 3 (6 Gbps) 0100 - 1111 = Reserved. ***Note:*** If (FFSATA0p0, FFSATA0p1, FFSATA0p2, FFSATA0p3, FFSATA0p4 and FFSATA0p5) is 1, this field is RWO defaulting to 0010 and ignores software write value of 0011. If either FFSATA0p0, FFSATA0p1, FFSATA0p2, FFSATA0p3, FFSATA0p4 or FFSATA0p5 is 0, this field is RWO defaulting to 0011.
19	0h RO	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
18	1h RO	**Supports AHCI mode only (SAM):** The SATA controller may optionally support AHCI access mechanism only. A value of 0 indicates that in addition to the native AHCI mechanism (via ABAR), the SATA controller implements a legacy, task-file based register interface such as SFF-8038i. A value of 1 indicates that the SATA controller does not implement a legacy, task-file based register interface.
17	1h RO	**Supports Port Multiplier (SPM):** The SATA controller may optionally support command-based switching Port Multipliers. BIOS must clear this bit if Port Multipliers are not supported.
16	0h RO	**FIS-based Switching Supported (FBSS):** Not supported.
15	1h RO	**PIO Multiple DRQ Block (PMD):** If set to 1, the HBA supports multiple DRQ block data transfers for the PIO command protocol.
14	1h RW/O	**Slumber State Capable (SSC):** The SATA controller supports the slumber state.
13	1h RW/O	**Partial State Capable (PSC):** The SATA controller supports the partial state.
12:8	1Fh RO	**Number of Command Slots (NCS):** 1Fh indicating support for 32 slots.
7	0h RO	**Command Completion Coalescing Supported (CCCS):** When set to 1, indicates that the HBA supports command completion coalescing. When command completion coalescing is supported, the HBA has implemented the CCC_CTL and the CCC_PORTS global HBA registers. When cleared to 0, indicates that the HBA does not support command completion coalescing and the CCC_CTL and CCC_PORTS global HBA registers are not implemented.
6	0h RO	**Enclosure Management Supported (EMS):** When set to 1, indicates that the HBA supports enclosure management. When enclosure management is supported, the HBA has implemented the EM_LOC and EM_CTL global HBA registers. When cleared to 0, indicates that the HBA does not support enclosure management and the EM_LOC and EM_CTL global HBA registers are not implemented.
5	0h RW/O	**Supports External SATA (SXS):** When set to 1, indicates that the HBA has one or more Serial ATA ports that has a signal only connector that is externally accessible. If this bit is set, software may refer to the PxCMD.ESP bit to determine whether a specific port has its signal connector externally accessible as a signal only connector (i.e., power is not part of that connector). When the bit is cleared to 0, indicates that the HBA has no Serial ATA ports that have a signal only connector externally accessible.
4:0	5h RO	**Number of Ports (NP):** 0's based value indicating the maximum number of ports supported. Note that the number of ports indicated in this field may be more than the number of ports indicated in the PI register. Number of ports shall be dependent on MAP.SC, fuses (FFSATA7, FFSATA8) and PCIe/SATA muxing configuration where if ANY of these parameter disable a particular port then that port is disabled and not counted. The maximum number of ports supported by SIP is 6and the least is 0 (i.e., Function Disable). In the case of 0 port configuration, the value of NP is a don't care (while implementation has it fixed as 05h). Any combination in between is supported by SATA host controller. Indicates the number of supported ports.

24.9.2.2 Global HBA Control (GHC)—Offset 4h

Global HBA Control. This register controls various global actions of the HBA.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 80000000h

31			28				24				20				16				12				8				4			0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
AE	RSVD																											MRSM	IE	HR

Bit Range	Default and Access	Field Name (ID): Description
31	1h RO	**AHCI Enable (AE):** When set, indicates that an AHCI driver is loaded and communication to the HBA shall be via AHCI mechanisms. This can be used by an HBA that supports both legacy mechanisms (such as SFF-8038i) and AHCI to know when the HBA is running under an AHCI driver. When set, software shall only talk to the HBA using AHCI. The HBA will not have to allow command processing via both AHCI and legacy mechanisms. When cleared, software will only communicate with the HBA using legacy mechanisms. Software shall set this bit to 1 before accessing other AHCI registers. ***Note:*** The implementation of this bit is dependent upon the value of the CAP.SAM bit. If CAP.SAM is 0, then GHC.AE should be RW and shall have a reset value of 0. If CAP.SAM is 1, then AE shall be read only and shall have a reset value of 1.
30:3	0h RO	Reserved.
2	0h RO	**MSI Revert to Single Message (MRSM):** When set to 1 by hardware, indicates that the HBA requested more than one MSI vector but has reverted to using the first vector only. When this bit is cleared to 0, the HBA has not reverted to single MSI mode (i.e., hardware is already in single MSI mode, software has allocated the number of messages requested, or hardware is sharing interrupt vectors if MC.MME < MC.MMC). The HBA may revert to single MSI mode when the number of vectors allocated by the host is less than the number requested. This bit shall only be set to 1 when the following conditions hold: MC.MSIE = 1 (MSI is enabled); MC.MMC > 0 (multiple messages requested); MC.MME > 0 (more than one message allocated); MC.MME != MC.MMC (messages allocated not equal to number requested). When this bit is set to 1, single MSI mode operation is in use and software is responsible for clearing bits in the IS register to clear interrupts. This bit shall be cleared to 0 by hardware when any of the four conditions stated is false. This bit is also cleared to 0 when MC.MSIE = 1 and MC.MME = 0h. In this case, the hardware has been programmed to use single MSI mode, and is not reverting to that mode. The HBA shall always revert to single MSI mode when the number of vectors allocated by the host is less than the number requested. Value of MRSM is a don't care when GHC.HR=1.
1	0h RW	**Interrupt Enable (IE):** This global bit enables interrupts from the HBA. When cleared (reset default), all interrupt sources from all ports are disabled. When set, interrupts are enabled.
0	0h RW/1S	**HBA Reset (HR):** When set by SW, this bit causes an internal reset of the HBA. All state machines that relate to data transfers and native command queuing will return to an idle condition, and all ports will be re-initialized via COMRESET. When the HBA has performed the reset action, it will reset this bit to 0. A software write of 0 will have no effect. For a description on which bits are reset when this bit is set, see the *AHCI Specification*, Section 10.3.3.

24.9.2.3 Interrupt Status Register (IS)—Offset 8h

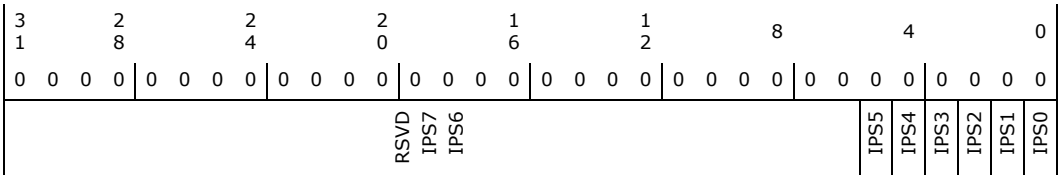
Interrupt Status Register. This register indicates which of the ports within the controller have an interrupt pending and require service.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 00000000h



Bit Range	Default and Access	Field Name (ID): Description
31:6	0h RO	Reserved.
5	0h RW/1C	**Interrupt Pending Status Port 5 (IPS5):** If set, indicates that port 5 has an interrupt pending. Software can use this information to determine which ports require service after an interrupt. This bit is only applicable to project(s): That has port 5 physically.
4	0h RW/1C	**Interrupt Pending Status Port 4 (IPS4):** If set, indicates that port 4 has an interrupt pending. Software can use this information to determine which ports require service after an interrupt. This bit is only applicable to project(s): That has port 4 physically.
3	0h RW/1C	**Interrupt Pending Status Port 3 (IPS3):** If set, indicates that port 3 has an interrupt pending. Software can use this information to determine which ports require service after an interrupt. This bit is only applicable to project(s): That has port 3 physically.
2	0h RW/1C	**Interrupt Pending Status Port 2 (IPS2):** If set, indicates that port 2 has an interrupt pending. Software can use this information to determine which ports require service after an interrupt. This bit is only applicable to project(s): That has port 2 physically.
1	0h RW/1C	**Interrupt Pending Status Port 1 (IPS1):** If set, indicates that port 1 has an interrupt pending. Software can use this information to determine which ports require service after an interrupt. This bit is only applicable to project(s): That has port 1 physically.
0	0h RW/1C	**Interrupt Pending Status Port 0 (IPS0):** If set, indicates that port 0 has an interrupt pending. Software can use this information to determine which ports require service after an interrupt. This bit is only applicable to project(s): That has port 0 physically.

24.9.2.4 Ports Implemented (GHC_PI)—Offset Ch

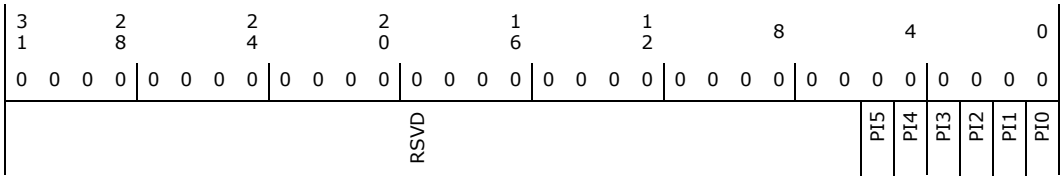
Ports Implemented. This register indicates which ports are exposed to the HBA. It is loaded by platform BIOS. It indicates which ports that the device supports are available for software to use. Any available port may not be implemented. This register is not reset by FLR.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:6	0h RO	Reserved.
5	0h RW/O	**Port 5 Implemented (PI5):** If set, then port 5 is available for use. If cleared, the port is not available for use. This bit is reserved and is read-only 0 if below condition is true else it's RWO-zero: Refer to CAP.NP where port 5 is not available.
4	0h RW/O	**Port 4 Implemented (PI4):** If set, then port 4 is available for use. If cleared, the port is not available for use. This bit is reserved and is read-only 0 if below condition is true else it's RWO-zero: Refer to CAP.NP where port 4 is not available.
3	0h RW/O	**Port 3 Implemented (PI3):** If set, then port 3 is available for use. If cleared, the port is not available for use. This bit is reserved and is read-only 0 if below condition is true else it's RWO-zero: Refer to CAP.NP where port 3 is not available.
2	0h RW/O	**Port 2 Implemented (PI2):** If set, then port 2 is available for use. If cleared, the port is not available for use. This bit is reserved and is read-only 0 if below condition is true else it's RWO-zero: Refer to CAP.NP where port 2 is not available.
1	0h RW/O	**Port 1 Implemented (PI1):** If set, then port 1 is available for use. If cleared, the port is not available for use. This bit is reserved and is read-only 0 if below condition is true else it's RWO-zero: Refer to CAP.NP where port 1 is not available.
0	0h RW/O	**Port 0 Implemented (PI0):** If set, then port 0 is available for use. If cleared, the port is not available for use. This bit is reserved and is read-only 0 if below condition is true else it's RWO-zero: Refer to CAP.NP where port 0 is not available.

24.9.2.5 AHCI Version (VS)—Offset 10h

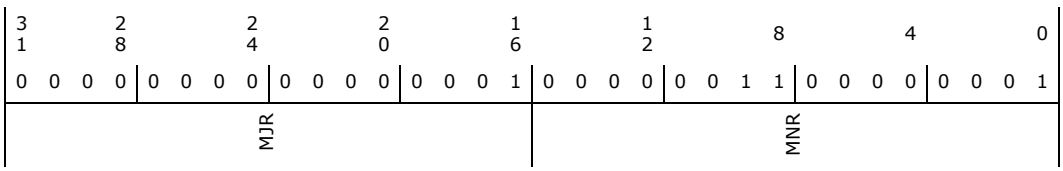
AHCI Version. This register indicates the major and minor version of the *AHCI Specification*. It is BCD encoded. The upper two bytes represent the major version number, and the lower two bytes represent the minor version number. Example: Version 3.12 would be represented as 00030102h.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 00010301h



Bit Range	Default and Access	Field Name (ID): Description
31:16	1h RO	**Major Version Number (MJR):** Indicates the major version is 1
15:0	300h RO	**Minor Version Number (MNR):** Indicates the minor version is 30

24.9.2.6 Enclosure Management Location (EM_LOC)—Offset 1Ch

Enclosure Management Location. The enclosure management location register identifies the location and size of the enclosure management message buffer.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 01600002h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
OFST																SZ															

Bit Range	Default and Access	Field Name (ID): Description
31:16	160h RO	**Offset (OFST):** The offset of the message buffer in Dwords from the beginning of the ABAR.
15:0	2h RO	**Buffer Size (SZ):** Specifies the size of the transmit message buffer area in Dwords. If both transmit and receive buffers are supported, then the transmit buffer begins at ABAR[EM_LOC.OFST*4] and the receive buffer directly follows it. If both transmit and receive buffers are supported, both buffers are of the size indicated in the Buffer Size field. A value of 0 is invalid. Note that SATA controller only supports transmit buffer.

24.9.2.7 Enclosure Management Control (EM_CTL)—Offset 20h

Enclosure Management Control. This register is used to control and obtain status for the enclosure management interface. The register includes information on the attributes of the implementation, enclosure management messages supported, the status of the interface, whether any messages are pending, and is used to initiate sending messages. This register is not implemented if enclosure management is not supported (i.e., CAP.EMS = 0).

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 07010000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD				ATTR_PM	ATTR_ALHD	ATTR_XMT	ATTR_SMB	RSVD				SUPP_SGPIO	SUPP_SES2	SUPP_SAFTE	SUPP_LED	RSVD				RST	CTL_TM	RSVD										STS_MR

Bit Range	Default and Access	Field Name (ID): Description
31:28	0h RO	Reserved.
27	0h RO	**Port Multiplier Support (ATTR_PM):** The HBA does not support enclosure management messages for devices attached via a Port Multiplier. Software should use the Serial ATA enclosure management bridge that is built into many Port Multipliers for enclosure services with these devices. For more information on Serial ATA enclosure management bridges, refer to the Serial ATA II: Extensions to Serial ATA 1.0a revision 1.2 specification.
26	1h RW/O	**Activity LED Hardware Driven (ATTR_ALHD):** If set to 1, the HBA drives the activity LED for the LED message type in hardware and does not utilize software settings for this LED. The HBA does not begin transmitting the hardware based activity signal until after software has written CTL.TM=1 after a reset condition.
25	1h RO	**Transmit Only (ATTR_XMT):** If set to 1, the HBA only supports transmitting messages and does not support receiving messages. If cleared to 0, the HBA supports transmitting and receiving messages.
24	1h RO	**Single Message Buffer (ATTR_SMB):** If set to 1, the HBA has one message buffer that is shared for messages to transmit and messages received. In this case, unsolicited receive messages are not supported and it is software's responsibility to manage access to this buffer. If cleared to 0, there are separate receive and transmit buffers such that unsolicited messages could be supported.
23:20	0h RO	Reserved.
19	0h RO	**SGPIO Enclosure Management Messages (SUPP_SGPIO):** If set to 1, the HBA supports the SGPIO register interface message type.
18	0h RO	**SES-2 Enclosure Management Messages (SUPP_SES2):** If set to 1, the HBA supports the SES-2 message type.
17	0h RO	**SAF-TE Enclosure Management Messages (SUPP_SAFTE):** If set to 1, the HBA supports the SAF-TE message type.
16	1h RO	**LED Message Types (SUPP_LED):** If set to 1, the HBA supports the LED message type defined in LED Message Type.
15:10	0h RO	Reserved.
9	0h RW/1S	**Reset (RST):** When set to 1 by software, the HBA shall reset all enclosure management message logic and take all appropriate reset actions to ensure messages can be transmitted/received after the reset. After the HBA completes the reset operation, the HBA shall set the value to 0. A write of 0 by software to this field shall have no effect.
8	0h RW/1S	**Transmit Message (CTL_TM):** When set to 1 by software, the HBA shall transmit the message contained in the message buffer. When the message is completely sent, the HBA shall clear this bit to 0. A write of 0 to this bit by software shall have no effect. Software shall not change the contents of the message buffer while CTL.TM is set to 1.
7:1	0h RO	Reserved.
0	0h RO	**Message Received (STS_MR):** Message received is not supported.

24.9.2.8 HBA Capabilities Extended (GHC_CAP2)—Offset 24h

HBA Capabilities Extended. This register indicates basic capabilities of the HBA to driver software. The RWO bits in this register are only cleared upon PLTRST#. This register is not reset by FLR.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 0000003Ch

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0
RSVD																										DESO	SADM	SDS	APST	RSVD	BOH

Bit Range	Default and Access	Field Name (ID): Description
31:6	0h RO	Reserved.
5	1h RW/O	**DEVSLP Entrance from Slumber Only (DESO):** This field specifies that the HBA shall only assert DEVSLP if the interface is in Slumber. When this bit is set to 1, the HBA shall ignore software directed entrance to DEVSLP via PxCMD.ICC unless PxSSTS.IPM = 6h. When this bit is cleared to 0, the HBA may enter DEVSLP from any link state (active, Partial, or Slumber). BIOS is required to program this field to 1.
4	1h RW/O	**Supports Aggressive DEVSLP Management (SADM):** When set to 1, the HBA supports hardware assertion of the DEVSLP signal after the idle timeout expires. When cleared to 0, this function is not supported and software shall treat the PxDEVSLP.ADSE field as reserved. ***Note:*** If PHY I/O PM Disable Fuse is 1, this register will read 0. Else this register will read 1 with RWO attribute.
3	1h RW/O	**Supports DEVSLP (SDS):** When set to 1, the HBA supports the DEVSLP feature. When cleared to 0, DEVSLP is not supported. ***Note:*** If PHY I/O PM Disable Fuse is 1, this register will read 0. Else this register will read 1 with RWO attribute.
2	1h RW/O	**Automatic Partial to Slumber Transitions (APST):** When set to 1, the HBA supports Automatic Partial to Slumber Transitions. When cleared to 0, Automatic Partial to Slumber Transition is not supported. ***Note:*** If SATA PHY PM Disable Fuse is 1, this register will read only 0. Else this register will read 1 with RWO attribute.
1	0h RO	Reserved.
0	0h RO	**BIOS/OS Handoff (BOH):** Not supported.

24.9.2.9 Vendor Specific (VSP)—Offset A0h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 00000048h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0
RSVD																									SFMS	PFS	PT	SRPIR	RSVD		

Bit Range	Default and Access	Field Name (ID): Description
31:7	0h RO	Reserved.
6	1h RO	**Software Feature Mask Supported (SFMS):** Set to 1 if the platform is enabled for premium storage features mask (SFM) as described in VS MMIO space at offset ABAR[RPID.(OFST*4)+4].
5	0h RO	**Premium Features Supported (PFS):** Set to 1 if the platform is enabled for premium storage features (PFB) as described in VS MMIO space at offset ABAR[RPID.OFST*4]. Set to 0 if the platform is not enabled for premium storage features.
4	0h RO	**Platform Type (PT):** 0b for servers.
3	1h RO	**Supports RAID Platform ID Reporting (SRPIR):** If set to 1, then indicates that the RAID Platform ID is reported via ABAR + C0h. Set to 0 if this ID is not reported via MMIO space.
2:0	0h RO	Reserved.

24.9.2.10 Vendor Specific Capabilities Register (VS_CAP)—Offset A4h

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 1002DEh

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	1	1	1	1	0
RSVD				NRMO												RSVD			MSL												NRMBE	

Bit Range	Default and Access	Field Name (ID): Description
31:28	0h RO	Reserved.
27:16	10h RW/O	**NVM Remapped Register Offset (NRMO):** Specifies the offset (in 128B unit) within ABAR as to where the PCIe NAND memory BAR register space is remapped. For example, NRMO=1 means ABAR + 128B. This allows the remapped offset to shift between ABAR + 0B, and ABAR + 512KK - 128B, with 128B step. The remapped offset into the AHCI memory space must not overlap with the memory space used for SATA. The remapped offset into the AHCI memory space and the remapped size must not exceed the allocated AHCI memory space of the integrated AHCI controller. Only valid when NRMBE = 1. The reset default of this field is 10h, this places the start of the PCIe NAND memory BAR register space at ABAR + 2K. This field is not reset by FLR.
15:13	0h RO	Reserved.
12:1	16Fh RW/O	**Memory Space Limit. (MSL):** This field specifies the size (in 128B unit) of the remapped memory space for the PCIe NAND device. It is a 0-based field. For example, MSL=1 means 256B. This allows the remapped size to shift between 128B and 512K with the step of 128B.Memory BAR offset from 0 to MSL of the PCIe NAND device are remapped under the integrated AHCI controller memory space.The remapped offset into the AHCI memory space and the value programmed in this field must not exceed the allocated AHCI memory space of the integrated AHCI controller. Only valid when NRMBE = 1. The reset default of this field is 16Fh which specifies the size of the remapped memory space as 34k.This field is not reset by FLR.
0	0h RW/O	**PCIe NAND Memory BAR Remapped Enable (NRMBE):** Set to 1 if a PCIe NAND device is present and remapping of its memory BAR register space is enabled. Cleared to 0 if there is no PCIe NAND device present or the remapping of its memory BAR register space is disabled. This bit is not reset by FLR.

24.9.2.11 RAID Platform ID (RPID)—Offset C0h

RAID Platform ID. This register is used by the Intel Matrix Storage Manager OROM to match the features supported by the OROM with the platform on which the OROM is executing. This prevents the use of an OROM designed for newer chipsets from being use on older chipsets as this could reduce up-sell potential.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: A256h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0	1	1	1	0	0	0	0	0	0	0	0	1	0
OFST																RPID																

Bit Range	Default and Access	Field Name (ID): Description
31:16	31h RO	**Offset (OFST):** The offset of the Premium Feature Block (PFB) in DWords from the beginning of the ABAR. SFM follows directly after PFB.
15:0	A184/ A204h RO	**RAID Platform ID (RPID):** Specifies the DID value that has been assigned to the platform. This is the same DID that is reported by the SATA controller when SATAGC.AIE is set to 1 except that the DID is always reported through this register, regardless if the programming of SATAGC.AIE.

24.9.2.12 Premium Feature Block (PFB)—Offset C4h

Premium Feature Block.

Note: Bits 4-0 are not bit-mapped to individual fuses and/or soft SKU settings; rather a single fuse FFSATA5 and FFSATA 3 /soft sku is used to indicate support for all of these features (refer to VSP.PFS). These registers indicate to the Intel Rapid Storage Technology AHCI driver that those premium RAID features that can be supported on the platform.

Type: MEM Register
(Size: 16 bits)

Device: 17
Function: 5

Default: 0000h

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD														SEA	SOI

Bit Range	Default and Access	Field Name (ID): Description
15:2	0h RO	Reserved.
1	0h RO	**Supports Email Alert (SEA):** Read value is the same as VSP.PFS. If set to 1, then this feature is supported as part of the premium feature set.
0	0h RO	**Supports OEM IOCTL (SOI):** Read value is the same as VSP.PFS. If set to 1, then this feature is supported as part of the premium feature set.

24.9.2.13 SW Feature Mask (SFM)—Offset C8h

SW Feature Mask. The following will be programmed by the BIOS when VS_CAP.SFMS == 1. The feature mask is used by SW to determine which non-premium features shall be supported by SW. These register bits are not reset by FLR since they are programmed by BIOS.

Access Method

Type: MEM Register
(Size: 16 bits)

Device: 17
Function: 5

Default: 3Fh

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
RSVD				OROM_UI_Normal_Delay		Smart_Response_Technology	IRRT_Only_on_ESATA	LED_Locate	HDDUNLOCK	OROM_UI_and_BANNER	IRRT	R5	R10	R1	R0

Bit Range	Default and Access	Field Name (ID): Description
15:12	0h RO	Reserved.
11:10	0h RW/O	**OROM UI Normal Delay. (OROM_UI_Normal_Delay):** Values of these bits specify the delay of the OROM UI Splash Screen in a normal status. 00 = 2 secs (default and previous value); 01 = 4 secs; 10 = 6 secs; 11 = 8 secs. If bit 5 == 0, then these values are disregarded.
9	0h RW/O	**Smart Response Technology. (Smart_Response_Technology):** If set to '1', then Smart Response Technology is enabled. If cleared to '0', the feature is disabled.
8	0h RW/O	**RRT Only on ESATA (IRRT_Only_on_ESATA):** If set to 1, then only RRT volumes can span internal and external SATA ports (e.g., eSATA). If cleared to 0, then any RAID volume can span internal and external SATA ports (e.g., eSATA).
7	0h RW/O	**LED Locate (LED_Locate):** If set to 1, then LED/SGPIO hardware is attached and the ping to locate feature is enabled in the OS.
6	0h RW/O	**HDDUNLOCK (HDDUNLOCK):** If set to 1, then HDD password unlock is enabled in the OS.
5	1h RW/O	**OROM UI and BANNER (OROM_UI_and_BANNER):** If set to 1, then the OROM UI is displayed. When cleared to 0, the OROM UI and BANNER are not displayed if all disks and volumes have a normal status.
4	1h RW/O	**RRT (IRRT):** If set to 1, then Rapid Recovery Technology is enabled.
3	1h RW/O	**R5 (R5):** If set to 1, then RAID5 is enabled
2	1h RW/O	**R10 (R10):** If set to 1, then RAID10 is enabled
1	1h RW/O	**R1 (R1):** If set to 1, then RAID1 is enabled
0	1h RW/O	**R0 (R0):** If set to 1, then RAID0 is enabled

24.9.2.14 Port 0 Command List Base Address (P0CLB)—Offset 100h

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: xxxxxx00h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
CLB																							RSVD								

Bit Range	Default and Access	Field Name (ID): Description
31:10	xxxxh RW	**Command List Base Address (CLB):** Indicates the 32-bit base for the command list for this port. This base is used when fetching commands to execute. This address must be 1K aligned as indicated by bits 31:10 being read/write. Note that these bits are not reset on a HBA reset. These bits are not reset on an HBA reset and at startup are undefined.
9:0	0h RO	Reserved.

24.9.2.15 Port 0 Command List Base Address Upper 32-bits (P0CLBU)—Offset 104h

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: XXXXXXXXh

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
CLBU																															

Bit Range	Default and Access	Field Name (ID): Description
31:0	xxxxxxxxh RW	**Command List Base Address Upper (CLBU):** Indicates the upper 32-bits for the command list base address for this port. This base is used when fetching commands to execute. Note that these bits are not reset on a HBA reset.

24.9.2.16 Port 0 FIS Base Address (P0FB)—Offset 108h

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: xxxxxx00h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
FB																							RSVD								

Bit Range	Default and Access	Field Name (ID): Description
31:8	0h RW	**FIS Base Address (FB):** Indicates the 32-bit base for received FISes. This address must be 256-byte aligned as indicated by bits 31:08 being read/write. When FIS-based switching is in use, this structure is 4KB in length and the address shall be 4KB aligned. Note that these bits are not reset on a HBA reset.
7:0	0h RO	Reserved.

24.9.2.17 Port 0 FIS Base Address Upper 32-bits (P0FBU)—Offset 10Ch

Port 0 FIS Base Address Upper 32-bits

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: xxxxxxxxh

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
FBU																															

Bit Range	Default and Access	Field Name (ID): Description
31:0	xxxxxxxxh RW	**FIS Base Address Upper (FBU):** Indicates the upper 32-bits for the received FIS base for this port. Note that these bits are not reset on a HBA reset.

24.9.2.18 Port 0 Interrupt Status (P0IS)—Offset 110h

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
CPDS	TFES	HBFS	HBDS	IFS	INFS	RSVD	OFS	IPMS	PRCS	RSVD														DMPS	PCS	DPS	UFS	SDBS	DSS	PSS	DHRS

Bit Range	Default and Access	Field Name (ID): Description
31	0h RO	**Cold Presence Detect Status (CPDS):** The SATA controller does not support cold presence detect.
30	0h RW/1C	**Task File Error Status (TFES):** This bit is set whenever the status register is updated by the device and the error bit (bit 0 of the Status field in the received FIS) is set.
29	0h RW/1C	**Host Bus Fatal Error Status (HBFS):** Indicates that the HBA encountered a host bus error that it cannot recover from, such as a bad software pointer. In PCI, such an indication would be a target or master abort.
28	0h RW/1C	**Host Bus Data Error Status (HBDS):** Indicates that the HBA encountered a data error (uncorrectable ECC / parity) when reading from or writing to system memory.
27	0h RW/1C	**Interface Fatal Error Status (IFS):** Indicates that the HBA encountered an error on the SATA interface which caused the transfer to stop.
26	0h RW/1C	**Interface Non-fatal Error Status (INFS):** Indicates that the HBA encountered an error on the SATA interface but was able to continue operation.
25	0h RO	Reserved.
24	0h RW/1C	**Overflow Status (OFS):** Indicates that the HBA received more bytes from a device than was specified in the PRD table for the command.
23	0h RW/1C	**Incorrect Port Multiplier Status (IPMS):** Port multiplier is not supported. Multiplier.
22	0h RO	**PhyRdy Change Status (PRCS):** When set to one indicates the internal PhyRdy signal changed state. This bit reflects the state of PxSERR.DIAG.N. This bit is RO and is only cleared when PxSERR.DIAG.N is cleared.
21:8	0h RO	Reserved.
7	0h RW/1C	**Device Mechanical Presence Status (DMPS):** When set, indicates that a platform mechanical presence switch has been opened or closed, which may lead to a change in the connection state of the device. This bit is only valid in systems that support mechanical presence switch (CAP.SMPS and PxCMD.MPSP are set).
6	0h RO	**Port Connect Change Status (PCS):** 1=Change in Current Connect Status. 0=No change in Current Connect Status. This bit reflects the state of PxSERR.DIAG.X. This bit is only cleared when PxSERR.DIAG.X is cleared.
5	0h RW/1C	**Descriptor Processed (DPS):** A PRD with the I. bit set has transferred all of its data.
4	0h RO	**Unknown FIS Interrupt (UFS):** When set to 1 indicates that an unknown FIS was received and has been copied into system memory. This bit is cleared to 0 by software clearing the PxSERR.DIAG.F bit to 0. Note that this bit does not directly reflect the PxSERR.DIAG.F bit. PxSERR.DIAG.F is set immediately when an unknown FIS is detected, whereas this bit is set when the FIS is posted to memory. Software should wait to act on an unknown FIS until this bit is set to 1 or the two bits may become out of sync.

Bit Range	Default and Access	Field Name (ID): Description
3	0h RW/1C	**Set Device Bits Interrupt (SDBS):** A Set Device Bits FIS has been received with the I. bit set and has been copied into system memory.
2	0h RW/1C	**DMA Setup FIS Interrupt (DSS):** A DMA Setup FIS has been received with the I. bit set and has been copied into system memory.
1	0h RW/1C	**PIO Setup FIS Interrupt (PSS):** A PIO Setup FIS has been received with the I. bit set, it has been copied into system memory, and the data related to that FIS has been transferred. This bit shall be set even if the data transfer resulted in an error.
0	0h RW/1C	**Device to Host Register FIS Interrupt (DHRS):** A D2H register FIS has been received with the I. bit set, and has been copied into system memory.

24.9.2.19 Port 0 Interrupt Enable (P0IE)—Offset 114h

Port 0 Interrupt Enable

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
CPDS	TFEE	HBFE	HBDE	IFE	INFE	RSVD	OFE	IPME	PRCE	RSVD													DMPE	PCE	DPE	UFE	SDBE	DSE	PSE	DHRE

Bit Range	Default and Access	Field Name (ID): Description
31	0h RO	**Cold Presence Detect Enable (CPDS):** The SATA controller does not support cold presence detect.
30	0h RW	**Task File Error Enable (TFEE):** When set, GHC.IE is set, and P0S.TFES is set, the HBA shall generate an interrupt.
29	0h RW	**Host Bus Fatal Error Enable (HBFE):** When set, GHC.IE is set, and P0IS.HBFS is set, the HBA shall generate an interrupt.
28	0h RW	**Host Bus Data Error Enable (HBDE):** When set, GHC.IE is set, and P0IS.HBDS is set, the HBA shall generate an interrupt.
27	0h RW	**Interface Fatal Error Enable (IFE):** When set, GHC.IE is set, and P0IS.IFS is set, the HBA shall generate an interrupt.
26	0h RW	**Interface Non-fatal Error Enable (INFE):** When set, GHC.IE is set, and P0IS.INFS is set, the HBA shall generate an interrupt.
25	0h RO	Reserved.
24	0h RW	**Overflow Enable (OFE):** When set, and GHC.IE and P0IS.OFS are set, the HBA shall generate an interrupt.
23	0h RW	**Incorrect Port Multiplier Enable (IPME):** When set, and GHC.IE and P0IS.IPMS are set, the HBA shall generate an interrupt. BIOS is required to program this field to '0'. The same applies to AHCI driver.
22	0h RW	**PhyRdy Change Interrupt Enable (PRCE):** When set, and GHC.IE is set, and PxIS.PRCS is set, the HBA shall generate an interrupt.

Bit Range	Default and Access	Field Name (ID): Description
21:8	0h RO	Reserved.
7	0h RW	**Device Mechanical Enable (DMPE):** When set, and P0IS.DMPS is set, the HBA shall generate an interrupt.
6	0h RW	**Port Change Interrupt Enable (PCE):** When set, GHC.IE is set, and P0IS.PCS is set, the HBA shall generate an interrupt.
5	0h RW	**Descriptor Processed Interrupt Enable (DPE):** When set, GHC.IE is set, and P0IS.DPS is set, the HBA shall generate an interrupt.
4	0h RW	**Unknown FIS Interrupt Enable (UFE):** When set, GHC.IE is set, and PxIS.UFS is set to 1, the HBA shall generate an interrupt.
3	0h RW	**Set Device Bits FIS Interrupt Enable (SDBE):** When set, GHC.IE is set, and P0IS.SDBS is set, the HBA shall generate an interrupt.
2	0h RW	**DMA Setup FIS Interrupt Enable (DSE):** When set, GHC.IE is set, and P0IS.DSS is set, the HBA shall generate an interrupt.
1	0h RW	**PIO Setup FIS Interrupt Enable (PSE):** When set, GHC.IE is set, and P0IS.PSS is set, the HBA shall generate an interrupt.
0	0h RW	**Device to Host Register FIS Interrupt Enable (DHRE):** When set, GHC.IE is set, and P0IS.DHRS is set, the HBA shall generate an interrupt.

24.9.2.20 Port 0 Command (P0CMD)—Offset 118h

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 4h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
ICC				ASP	ALPE	DLAE	ATAPI	APSTE	FBSCP	ESP	CPD	MPSP	HPCP	PMA	RSVD	CR	FR	MPSS	CCS				RSVD	PSP	RSVD	FRE	CLO	POD	SUD	ST		

Bit Range	Default and Access	Field Name (ID): Description
31:28	0h RW	**Interface Communication Control (ICC):** This is a four bit field which can be used to control reset and power states of the interface. If the Link layer is currently in the L_IDLE state, writes to this field shall cause the HBA to initiate a transition to the interface power management state requested. If the Link layer is not currently in the L_IDLE state, writes to this field shall have no effect.
27	0h RW	**Aggressive Slumber Partial (ASP):** When set to 1, and the ALPE bit is set, the HBA shall aggressively enter the Slumber state when it clears a bit in the PxCI or PxSACT register and the register values are then PxCI = 0h and PxSACT = 0h. When cleared, and the ALPE bit is set, the HBA will aggressively enter the Partial state when it clears a bit in the PxCI or PxSACT register and the register values are then PxCI = 0h and PxSACT = 0h. If CAP.SALP is cleared to 0., software shall treat this bit as reserved.
26	0h RW	**Aggressive Link Power Management Enable (ALPE):** When set, the HBA will aggressively enter a lower link power state (Partial or Slumber) based upon the setting of the ASP bit. Software shall only set this bit to 1 if CAP.SALP is set to 1. If CAP.SALP is cleared to 0, software shall treat this bit as reserved. BIOS is recommended to program this field to 1.

Bit Range	Default and Access	Field Name (ID): Description
25	0h RW	**Drive LED on ATAPI Enable (DLAE):** When set, the HBA will drive the LED pin active for commands regardless of the state of PxCMD.ATAPI. When cleared, the HBA will only drive the LED pin active for commands if PxCMD.ATAPI is set to 0. This bit is set by software
24	0h RW	**Device is ATAPI (ATAPI):** When set, the connected device is an ATAPI device. This bit is used by the HBA to control whether or not to generate the desktop LED when commands are active.
23	0h RW	**Automatic Partial to Slumber Transitions Enabled (APSTE):** When set to 1., the HBA may perform Automatic Partial to Slumber Transitions. When cleared to 0. the port shall not perform Automatic Partial to Slumber Transitions. Software shall only set this bit to 1. if CAP2.APST is set to 1.; if CAP2.APST is cleared to 0. software shall treat this bit as reserved.
22	0h RO	**FIS-based Switching Capable Port (FBSCP):** The SSATA controller does not support FIS-Based Switching.
21	0h RW/O	**External SATA Port (ESP):** When set to 1, indicates that this port is routed externally and will be used with an external SATA device. When set to 1, CAP.SXS must also be set to 1. When cleared (0), indicates that this port is not routed externally and supports internal SATA devices only. If ESP is set to '1', then the port may experience hot plug events.
20	0h RO	**Cold Presence Detection (CPD):** The SATA controller does not support cold presence detect.
19	0h RW/O	**Mechanical Presence Switch Attached to Port (MPSP):** If set to 1, the platform supports a mechanical presence switch attached to this port. If cleared to 0, the platform does not support a mechanical presence switch attached to this port.
18	0h RW/O	**Hot Plug Capable Port (HPCP):** This indicates whether the this port is connected to a device which can be hot plugged. SATA by definition is hot-pluggable, but not all platforms are constructed to allow the device to be removed (it may be screwed into the chassis, for example). This bit can be used by system software to indicate a feature such as eject device to the end-user.
17	0h RO	**Port Multiplier Attached (PMA):** When set, a Port Multiplier is attached to the HBA for this port. When cleared, a Port Multiplier is not attached to the HBA for this port. This bit is a read only 0. when CAP.PMS = 0., and read/write when CAP.PMS = 1. Note that this bit is set by software; hardware does not auto-detect that a Port Multiplier is attached.
16	0h RO	Reserved.
15	0h RO	**Command List Running (CR):** When this bit is set it indicates that the command list DMA engine for the port is running.
14	0h RO	**FIS Receive Running (FR):** When this bit is set it indicates that the FIS Receive DMA engine for the port is running. Note to software: When FR bit stays set, read the PxIS.PCS and PxSERR.DET.X to service the PCS interrupt accordingly if any
13	0h RO	**Mechanical Presence Switch State (MPSS):** The MPSS bit reports the state of a mechanical presence switch attached to this port. If CAP.SMPS is set to 1 and the mechanical presence switch is closed then this bit is cleared to 0. If CAP.SMPS is set to 1 and the mechanical presence switch is open then this bit is set to 1. If CAP.SMPS is set to 0 then this bit is cleared to 0. Software should only use this bit if both CAP.SMPS and PxCMD.MPSP are set to 1.
12:8	0h RO	**Current Command Slot (CCS):** Indicates the current command slot the HBA is processing. This field is valid when the PxCMD.ST bit is set, and is constantly updated by the HBA. This field can be updated as soon as the HBA recognizes an active command slot, or at some point soon after when it begins processing the command. When PxCMD.ST transitions from a 1 to a 0, the HBA will reset this field to 0. After PxCMD.ST transitions from 0 to 1, the highest priority slot to issue from next is command slot 0. After the first command has been issued, the highest priority slot to issue from next is PxCMD.CCS + 1. For example, after the HBA has issued its first command, if PxCMD.CCS = 0h and PxCI is set to 3h, the next command that will be issued is from command slot 1.
7	0h RO	Reserved.
6	0h RO	**PHYSLP Present (PSP):** If set to '1', the platform supports PHYSLP on this port. If cleared to '0', the platform does not support PHYSLP on this port. This bit may only be set to '1' if CAP2.SPS is set to '1'.
5	0h RO	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
4	0h RW	**FIS Receive Enable (FRE):** When set, the HBA may post received FISes into the FIS receive area pointed to by PxFB and PxFBU. When cleared, received FISes are not accepted by the HBA, except for the first D2H register FIS after the initialization sequence.
3	0h RW/1S	**Command List Override (CLO):** Setting this bit to 1 causes PxTFD.STS.BSY and PxTFD.STS.DRQ to be cleared to 0. This allows a software reset to be transmitted to the device regardless of whether the BSY and DRQ bits are still set in the PxTFD.STS register. The HBA sets this bit to 0 when PxTFD.STS.BSY and PxTFD.STS.DRQ have been cleared to 0. A write to this register with a value of 0 shall have no effect.
2	1h RO	**Power On Device (POD):** The SATA controller does not support cold presence detect.
1	0h RW	**Spin-Up Device (SUD):** This bit is read/write for HBAs that support staggered spin-up via CAP.SSS. This bit is read only 1. for HBAs that do not support staggered spin-up. On an edge detect from 0. to 1, the HBA shall start a COMRESET initialization sequence to the device. Clearing this bit causes no action on the interface. Clearing this bit to0 does not cause any OOB signal to be sent on the interface. When this bit is cleared to 0. and PxSCTL.DET = 0h, the HBA will enter listen mode.
0	0h RW	**Start (ST):** When set, the HBA may process the command list. When cleared, the HBA may not process the command list. Whenever this bit is changed from a 0 to a 1, the HBA starts processing the command list at entry 0. Whenever this bit is changed from a 1 to a 0, the PxCI and PxSACT register is cleared by the HBA upon the HBA putting the controller into an idle state. PxTFD shall be updated also.

24.9.2.21 Port 0 Task File Data (P0TFD)—Offset 120h

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 00000009h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1
RSVD																ERR								STS_BSY	RSVD				STS_DRQ	RSVD		STS_ERR

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:8	0h RO	**Error (ERR):** Contains the latest copy of the task file error register.
7	0h RO	**Status Busy (STS_BSY):** Status - Indicates the interface is busy.
6:4	0h RO	Reserved.
3	1h RO	**Status Drq (STS_DRQ):** Status - Indicates a data transfer is requested.
2:1	0h RO	Reserved.
0	1h RO	**Status Err (STS_ERR):** Status - Indicates an error during the transfer.

24.9.2.22 Port 0 Signature (P0SIG)—Offset 124h

Port 0 Signature

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: FFFFFFFFh

31			28				24				20				16				12				8				4				0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
SIG																															

Bit Range	Default and Access	Field Name (ID): Description
31:0	FFFFFFFFh RO	**Signature (SIG):** Contains the signature received from a device on the first D2H Register FIS.

24.9.2.23 Port 0 Serial ATA Status (P0SSTS)—Offset 128h

Port 0 Serial ATA Status

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																				IPM				SPD				DET			

Bit Range	Default and Access	Field Name (ID): Description
31:12	0h RO	Reserved.
11:8	0h RO	**Interface Power Management (IPM):** Indicates the current interface state
7:4	0h RO	**Current Interface Speed (SPD):** Indicates the negotiated interface communication speed.
3:0	0h RO	**Device Detection (DET):** Indicates the interface device detection and PHY state.

24.9.2.24 Port 0 Serial ATA Control (P0SCTL)—Offset 12Ch

Port 0 Serial ATA Control

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD												PMP				SPM				IPM				SPD				DET			

Bit Range	Default and Access	Field Name (ID): Description
31:20	0h RO	Reserved.
19:16	0h RW	**Port Multiplier Port (PMP):** This field is not used by AHCI.
15:12	0h RW	**Select Power Management (SPM):** This field is not used by AHCI.
11:8	0h RW	**Interface Power Management Transitions Allowed (IPM):** Indicates which power states the HBA is allowed to transition to. If an interface power management state is not allowed via this register field, the HBA will not initiate that state and the HBA will PMNAKP any request from the device to enter that state.
7:4	0h RW	**Speed Allowed (SPD):** Indicates the highest allowable speed of the interface.
3:0	0h RW	**Device Detection Initialization (DET):** Controls the HBA.s device detection and interface initialization.

24.9.2.25 Port 0 Serial ATA Error (P0SERR)—Offset 130h

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DIAG																ERR																

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RW/1C	**Diagnostics (DIAG):** Contains diagnostic error information for use by diagnostic software in validating correct operation or isolating failure modes.
15:0	0h RW/1C	**Error (ERR):** The ERR field contains error information for use by host software in determining the appropriate response to the error condition. If one or more of bits 11:8 of this register are set, the controller will stop the current transfer.

24.9.2.26 Port 0 Serial ATA Active (P0SACT)—Offset 134h

Port 0 Serial ATA Active

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DS																															

Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/1S	**Device Status (DS):** System software sets this bit for native command queuing commands prior to setting the PxCI.CI bit in the same command slot entry. This field is cleared via the Set Device Bits FIS.

24.9.2.27 Port 0 Command Issue (P0CI)—Offset 138h

Port 0 Commands Issued

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
CI																															

Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Commands Issued (CI):** This field is set by software to indicate to the HBA that a command has been built in system memory for a command slot and may be sent to the device. When the HBA receives a FIS which clears the BSY, ERR, and DRQ bits for the command, it clears the corresponding bit in this register for that command slot. Bits in this field shall only be set to 1 by software when PxCMD.ST is set to 1.

24.9.2.28 Port 0 SNotification (P0SNTF)—Offset 13Ch

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																PMN															

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:0	0h RW/1C	**PM Notify (PMN):** This field indicates whether a particular device with the corresponding PM Port number issued a Set Device Bits FIS to the host with the Notification bit set. PM Port 0h sets bit 0. PM Port Fh sets bit 15. Individual bits are cleared by software writing 1's to the corresponding bit positions. Note that, while this field is reset to default on a HBA Reset, it is not reset by COMRESET or SRST

24.9.2.29 Port 0 Device Sleep (P0DEVSLP)—Offset 144h

Port 0 Device Sleep

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 1E022852h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	1	1	1	1	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	0	0	0	1	0	1	0	0	1	0
RSVD			DM				DITO										MDAT							DETO							DSP	ADSE

Bit Range	Default and Access	Field Name (ID): Description
31:29	0h RO	Reserved.
28:25	Fh RW/O	**DITO Multiplier (DM):** 0's based value that specifies the DITO multiplier that the HBA applies to the specified DITO value, effectively extending the range of DITO from 1ms to 16368ms. A value of 0h indicates a multiplier of 1. A maximum multiplier of 16 may be applied. The HBA computes the total idle timeout as a product of DM and DITO (i.e., DITO actual = DITO * DM).
24:15	4h RW	**DEVSLP Idle Timeout (DITO):** This field specifies the amount of the time (in approximate 1ms granularity) that the HBA shall wait before driving the DEVSLP signal.Hardware reloads its port specific DEVSLP timer with this value each time the port transitions out of DEVSLP state. For example: from DEVSLP to active or PxDEVSLP.ADSE transitions from 0 to a 1. If CAP2.SDS or CAP2.SADM or PxDEVSLP.DSP is cleared to 0 then these bits are read-only 0h and software shall treat these bits as reserved. Software shall only set this value when PxCMD.ST is cleared to 0 and PxDEVSLP.ADSE is cleared to 0.
14:10	Ah RW	**DEVSLP Minimum Assertion Time (MDAT):** This field specifies the minimum amount of time (in 1ms granularity) that the HBA must assert the DEVSLP signal before it may be de-asserted. The nominal value is 10ms and the minimum is 1ms depending on device identification information. If CAP2.SDS is cleared to 0 or PxDEVSLP.DSP is cleared to 0 then these bits are read-only 0h and software shall treat these bits as reserved. Software shall only set this value when PxCMD.ST is cleared to 0, PxDEVSLP.ADSE is cleared to 0, and prior to setting PxCMD.ICC to 8h.
9:2	14h RW	**DEVSLP Exit Timeout (DETO):** This field specifies the maximum duration (in approximate 1ms granularity) from DEVSLP de-assertion until the device is ready to accept OOB. The nominal value is 20ms while the max value is 255ms depending on device identification information. If CAP2.SDS is cleared to 0 or PxDEVSLP.DSP is cleared to 0 then these bits are read-only 0h and software shall treat these bits as reserved. Software shall only set this value when PxCMD.ST is cleared to 0, PxDEVSLP.ADSE is cleared to 0, and prior to setting PxCMD.ICC to 8h.
1	1h RW/O	**DEVSLP Present (DSP):** If set to '1', the platform supports DEVSLP on this port. If cleared to '0', the platform does not support DEVSLP on this port. This bit may only be set to '1' if CAP2.SDS is set to '1'.DSP is mutually exclusive with the PxCMD.HPCP bit and PxCMD.ESP bit. Note that these bits are not reset on a HBA reset.
0	0h RW	**Aggressive DEVSLP Enable (ADSE):** This bit is read/write for HBAs that support aggressive DEVSLP management (CAP2. SADM = ?1?). When this bit is set to ?1?, the HBA shall assert the DEVSLP signal after the port has been idle (PxCI = 0h and PxSACT = 0h) for the amount of time specified by the PxDEVSLP.DITO register and the interface is in Slumber (PxSSTS.IPM = 6h). When this bit is cleared to ?0?, the HBA does not enter DEVSLP unless software directed via PxCMD.ICC. This bit shall only be set to ?1? if PxDEVSLP.DSP is set to 1. If this bit is set to 1 and software clears the bit to 0, then the HBA shall de-assert the DEVSLP signal if asserted. Note that these bits are not reset on a HBA reset. BIOS is recommended to program this field to 1 if the platform support the DEVSLP feature. If CAP2.SDS is cleared to 0 or CAP2.SADM is cleared to 0, or if PxDEVSLP.DSP is cleared to 0 then these bits are read-only 0h and software shall treat these bits as reserved.

24.9.2.30 Port 1 Command List Base Address (P1CLB)—Offset 180h

Same bit definition as P0CLB.

24.9.2.31 Port 1 Command List Base Address Upper 32-bits (P1CLBU)—Offset 184h

Same bit definition as P0CLBU.

24.9.2.32 Port 1 FIS Base Address (P1FB)—Offset 188h

Same bit definition as P0FB.

24.9.2.33 Port 1 FIS Base Address Upper 32-bits (P1FBU)—Offset 18Ch

Same bit definition as P0FBU.

24.9.2.34 Port 1 Interrupt Status (P1IS)—Offset 190h

Same bit definition as P0IS.

24.9.3 Port 1 Interrupt Enable (P1IE)—Offset 194h

Same bit definition as P0IE.

24.9.3.1 Port 1 Command (P1CMD)—Offset 198h

Same bit definition as P0CMD.

24.9.3.2 Port 1 Task File Data (P1TFD)—Offset 1A0h

Same bit definition as P0TFD.

24.9.3.3 Port 1 Signature (P1SIG)—Offset 1A4h

Same bit definition as P0SIG.

24.9.3.4 Port 1 Serial ATA Status (P1SSTS)—Offset 1A8h

Same bit definition as P0SSTS.

24.9.3.5 Port 1 Serial ATA Control (P1SCTL)—Offset 1ACh

Same bit definition as P0SCTL.

24.9.3.6 Port 1 Serial ATA Error (P1SERR)—Offset 1B0h

Same bit definition as P0SERR.

24.9.3.7 Port 1 Serial ATA Active (P1SACT)—Offset 1B4h

Same bit definition as P0SACT.

24.9.3.8 Port 1 Command Issue (P1CI)—Offset 1B8h

Same bit definition as P0CI.

24.9.3.9 Port 1 SNotification (P1SNTF)—Offset 1BCh

Same bit definition as P0SNTF.

24.9.3.10 Port 1 Device Sleep (P1DEVSLP)—Offset 1C4h

Same bit definition as P0DEVSLP.

24.9.3.11 Port 2 Command List Base Address (P2CLB)—Offset 200h

Same bit definition as P0CLB.

24.9.3.12 Port 2 Command List Base Address Upper 32-bits (P2CLBU)—Offset 204h

Same bit definition as P0CLBU.

24.9.3.13 Port 2 FIS Base Address (P2FB)—Offset 208h

Same bit definition as P0FB.

24.9.3.14 Port 2 FIS Base Address Upper 32-bits (P2FBU)—Offset 20Ch

Same bit definition as P0FBU.

24.9.3.15 Port 2 Interrupt Status (P2IS)—Offset 210h

Same bit definition as P0IS.

24.9.3.16 Port 2 Interrupt Enable (P2IE)—Offset 214h

Same bit definition as P0IE.

24.9.3.17 Port 2 Command (P2CMD)—Offset 218h

Same bit definition as P0CMD.

24.9.3.18 Port 2 Task File Data (P2TFD)—Offset 220h

Same bit definition as P0TFD.

24.9.3.19 Port 2 Signature (P2SIG)—Offset 224h

Same bit definition as P0SIG.

24.9.3.20 Port 2 Serial ATA Status (P2SSTS)—Offset 228h

Same bit definition as P0SSTS.

24.9.3.21 Port 2 Serial ATA Control (P2SCTL)—Offset 22Ch

Same bit definition as P0SCTL.

24.9.3.22 Port 2 Serial ATA Error (P2SERR)—Offset 230h

Same bit definition as P0SERR.

24.9.3.23 Port 2 Serial ATA Active (P2SACT)—Offset 234h

Same bit definition as P0SACT.

24.9.3.24 Port 2 Command Issue (P2CI)—Offset 238h

Same bit definition as P0CI.

24.9.3.25 Port 2 SNotification (P2SNTF)—Offset 23Ch

Same bit definition as P0SNTF.

24.9.3.26 Port 2 Device Sleep (P2DEVSLP)—Offset 244h

Same bit definition as P0DEVSLP.

24.9.3.27 Port 3 Command List Base Address (P3CLB)—Offset 280h

Same bit definition as P0CLB.

24.9.3.28 Port 3 Command List Base Address Upper 32-bits (P3CLBU)—Offset 284h

Same bit definition as P0CLBU.

24.9.3.29 Port 3 FIS Base Address (P3FB)—Offset 288h

Same bit definition as P0FB.

24.9.3.30 Port 3 FIS Base Address Upper 32-bits (P3FBU)—Offset 28Ch

Same bit definition as P0FBU.

24.9.3.31 Port 3 Interrupt Status (P3IS)—Offset 290h

Same bit definition as P0IS.

24.9.3.32 Port 3 Interrupt Enable (P3IE)—Offset 294h

Same bit definition as P0IE.

24.9.3.33 Port 3 Command (P3CMD)—Offset 298h

Same bit definition as P0CMD.

24.9.3.34 Port 3 Task File Data (P3TFD)—Offset 2A0h

Same bit definition as P0TFD.

24.9.3.35 Port 3 Signature (P3SIG)—Offset 2A4h

Same bit definition as P0SIG.

24.9.3.36 Port 3 Serial ATA Status (P3SSTS)—Offset 2A8h

Same bit definition as P0SSTS.

24.9.3.37 Port 3 Serial ATA Control (P3SCTL)—Offset 2ACh

Same bit definition as P0SCTL.

24.9.3.38 Port 3 Serial ATA Error (P3SERR)—Offset 2B0h

Same bit definition as P0SERR.

24.9.3.39 Port 3 Serial ATA Active (P3SACT)—Offset 2B4h

Same bit definition as P0SACT.

24.9.3.40 Port 3 Commands Issued (P3CI)—Offset 2B8h

Same bit definition as P0CI.

24.9.3.41 Port 3 SNotification (P3SNTF)—Offset 2BCh

Same bit definition as P0SNTF.

24.9.3.42 Port 3 Device Sleep (P3DEVSLP)—Offset 2C4h

Same bit definition as P0DEVSLP.

24.9.3.43 Port 4 Command List Base Address (P4CLB)—Offset 300h

Same bit definition as P0CLB.

24.9.3.44 Port 4 Command List Base Address Upper 32-bits (P4CLBU)—Offset 304h

Same bit definition as P0CLBU.

24.9.3.45 Port 4 FIS Base Address (P4FB)—Offset 308h

Same bit definition as P0FB.

24.9.3.46 Port 4 FIS Base Address Upper 32-bits (P4FBU)—Offset 30Ch

Same bit definition as P0FBU.

24.9.3.47 Port 4 Interrupt Status (P4IS)—Offset 310h

Same bit definition as P0IS.

24.9.3.48 Port 4 Interrupt Enable (P4IE)—Offset 314h

Same bit definition as P0IE.

24.9.3.49 Port 4 Command (P4CMD)—Offset 318h

Same bit definition as P0CMD.

24.9.3.50 Port 4 Task File Data (P4TFD)—Offset 320h

Same bit description as P0TFD.

24.9.3.51 Port 4 Signature (P4SIG)—Offset 324h

Same bit description as P0SIG.

24.9.3.52 Port 4 Serial ATA Status (P4SSTS)—Offset 328h

Same bit description as P0SSTS.

24.9.3.53 Port 4 Serial ATA Control (P4SCTL)—Offset 32Ch

Same bit description as P0SCTL.

24.9.3.54 Port 4 Serial ATA Error (P4SERR)—Offset 330h

Same bit description as P0SERR.

24.9.3.55 Port 4 Serial ATA Active (P4SACT)—Offset 334h

Same bit description as P0SACT.

24.9.3.56 Port 4 Commands Issued (P4CI)—Offset 338h

Same bit description as P0CI.

24.9.3.57 Port 4 SNotification (P4SNTF)—Offset 33Ch

Same bit description as P0SNTF.

24.9.3.58 Port 4 Device Sleep (P4DEVSLP)—Offset 344h

Same bit description as P0DEVSLP.

24.9.3.59 Port 5 Command List Base Address (P5CLB)—Offset 380h

Same bit description as P0CLB.

24.9.3.60 Port 5 Command List Base Address Upper 32-bits (P5CLBU)—Offset 384h

Same bit definition as P0CLBU.

24.9.3.61 Port 5 FIS Base Address (P5FB)—Offset 388h

Same bit definition as P0FB.

24.9.3.62 Port 5 FIS Base Address Upper 32-bits (P5FBU)—Offset 38Ch

Same bit definition as P0FBU.

24.9.3.63 Port 5 Interrupt Status (P5IS)—Offset 390h

Same bit definition as P0IS.

24.9.3.64 Port 5 Interrupt Enable (P5IE)—Offset 394h

Same bit definition as P0IE.

24.9.3.65 Port 5 Command (P5CMD)—Offset 398h

Same bit definition as P0CMD.

24.9.3.66 Port 5 Task File Data (P5TFD)—Offset 3A0h

Same bit definition as P0TFD.

24.9.3.67 Port 5 Signature (P5SIG)—Offset 3A4h

Same bit definition as P0SIG.

24.9.3.68 Port 5 Serial ATA Status (P5SSTS)—Offset 3A8h

Same bit definition as P0SSTS.

24.9.3.69 Port 5 Serial ATA Control (P5SCTL)—Offset 3ACh

Same bit definition as P0SCTL.

24.9.3.70 Port 5 Serial ATA Error (P5SERR)—Offset 3B0h

Same bit definition as P0SERR.

24.9.3.71 Port 5 Serial ATA Active (P5SACT)—Offset 3B4h

Same bit definition as P0SACT.

24.9.3.72 Port 5 Commands Issued (P5CI)—Offset 3B8h

Same bit definition as P0CI.

24.9.3.73 Port 5 SNotification (P5SNTF)—Offset 3BCh

Same bit definition as P0SNTF.

24.9.3.74 Port 5 Device Sleep (P5DEVSLP)—Offset 3C4h

Same bit definition as P0DEVSLP.

24.9.3.75 Enclosure Management Message Format (EM_MF)—Offset 580h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD				MTYPE				DSIZE								MSIZE								RSVD								

Bit Range	Default and Access	Field Name (ID): Description
31:28	0h RO	Reserved.
27:24	0h RW	**Message Type (MTYPE):** Specifies the type of the message. The message types are: 0h = LED; 1h = SAF-TE; 2h = SES-2; 3h = SGPIO (register based interface); All other values reserved.
23:16	0h RW	**Data Size (DSIZE):** Specifies the data size in bytes. If the message (enclosure services command) has a data buffer that is associated with it that is transferred, the size of that data buffer is specified in this field. If there is no separate data buffer, this field shall have a value of '0'. The data directly follows the message in the message buffer. This value should always be '0'.
15:8	0h RW	**Message Size (MSIZE):** Specifies the size of the message in bytes. The message size does not include the one Dword header. A value of '0'. is invalid. The message size is always 4 bytes.
7:0	0h RO	Reserved.

24.9.3.76 Enclosure Management LED (EM_LED)—Offset 584h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
VAL																PM								HBA								

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RW	**Value (VAL):** This field describes the state of each LED for a particular location. There are three LEDs that may be supported by the HBA. Each LED has 3 bits of control. LED values are: 000b - LED shall be off 001b - LED shall be solid on as perceived by human eye; All other values reserved. The LED bit locations are: Bits 2:0 - Activity LED (may be driven by hardware) Bits 5:3 - Vendor Specific LED (e.g., locate) Bits 8:6 - Vendor Specific LED (e.g., fault) Bits 15:9 - Reserved. Vendor specific message is: Bit 3:0 - Vendor Specific Pattern; Bit 15:4 - Reserved. ***Note:*** If Activity LED Hardware Driven (ATTR.ALHD) bit is set, host will output the hardware LED value sampled internally and will ignore software written activity value on bit [2:0]. Since Enclosure Management does not support port multiplier based LED message, the LED message will be generated independently based on respective port's operation activity. Vendor specific LED values Locate (Bits 5:3) and Fault (Bits 8:6) always are driven by software.
15:8	0h RW	**Port Multiplier Information (PM):** Specifies slot specific information related to Port Multiplier. Bits 3:0 specify the Port Multiplier port number for the slot that requires the status update. If a Port Multiplier is not attached to the device in the affected slot, the Port Multiplier port number shall be '0'. Bits 7:4 are reserved. SATA does not support LED messages for devices behind a Port Multiplier. This byte should be 0.
7:0	0h RW	**HBA Information (HBA):** Specifies slot specific information related to the HBA. Bits 4:0 - HBA port number for the slot that requires the status update. Bit 5 - If set to '1', Value is a vendor specific message that applies to the entire enclosure. If cleared to '0', Value applies to the port specified in bits 4:0. Bits 7:6 - Reserved.

24.9.4 SATA AIDP Registers Summary

Table 24-8. Summary of SATA AIDP Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
10h	13h	AHCI Index Register (INDEX)—Offset 10h	0h
14h	17h	AHCI Data Register (DATA)—Offset 14h	0h

24.9.4.1 AHCI Index Register (INDEX)—Offset 10h

AHCI Index Register. This registers are only available if CC.SCC is not 01h to index into all memory registers defined in Memory Registers and the message buffer used for enclosure management. If CC.SCC is 01h, these AHCI Index Data Pair registers are not accessible and SINDEX/SDATA register pair shall be used to index into a subset of the memory registers defined in (See Memory Registers for more information on which registers could be indexed).

Access Method

Type: I/O Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD															INDEX															RSVD		

Bit Range	Default and Access	Field Name (ID): Description
31:19	0h RO	Reserved.
18:2	0h RW	**Index (INDEX):** This Index register is used to select the Dword offset of the Memory Mapped AHCI register to be accessed. A Dword, Word or Byte access is specified by the active byte enables of the I/O access to the Data register.
1:0	0h RO	Reserved.

24.9.4.2 AHCI Data Register (DATA)—Offset 14h

AHCI Data Register. This registers are index into all memory registers defined in Memory Registers and the message buffer used for enclosure management.

Access Method

Type: I/O Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DATA																															

Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Data (DATA):** This Data register is a window through which data is read or written to the memory mapped register pointed to by the Index register. Note that a physical register is not actually implemented as the data is actually stored in the memory mapped registers. Since this is not a physical register, the default value is the same as the default value of the register pointed to by Index.

24.9.5 SATA MXTBA Registers Summary

Table 24-9. Summary of SATA MXTBA Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	MSI-X Table Entries 0 Message Lower Address (MXTE0MLA)—Offset 0h	0h
4h	7h	MSI-X Table Entries 0 Message Upper Address (MXTE0MUA)—Offset 4h	0h
8h	Bh	MSI-X Table Entries 0 Message Data (MXTE0MD)—Offset 8h	0h
Ch	Fh	MSI-X Table Entries 0 Vector Control (MXTE0VC)—Offset Ch	1h

24.9.5.1 MSI-X Table Entries 0 Message Lower Address (MXTE0MLA)—Offset 0h

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MXMLA																															RSVD

Bit Range	Default and Access	Field Name (ID): Description
31:2	0h RW	**MSI-X message lower address (MXMLA):** Specifies the lower 32-bit of the DWORD-aligned address (Addr[31:02]) of the MSI-X Message
1:0	0h RO	Reserved.

24.9.5.2 MSI-X Table Entries 0 Message Upper Address (MXTE0MUA)—Offset 4h

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 00000000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MXMUA																															

Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**MSI-X message upper 32-bit address (MXMUA):** Specifies the lower 32-bit of the DWORD-aligned address (Addr[31:02]) of the MSI-X Message.

24.9.5.3 MSI-X Table Entries 0 Message Data (MXTE0MD)—Offset 8h

MSI-X Table Entries 0 Message Data

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 00000000h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MXMD																															

Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**MSI-X message Data (MXMD):** Specifies the 32-bit Data of the MSI-X Message.

24.9.5.4 MSI-X Table Entries 0 Vector Control (MXTE0VC)—Offset Ch

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 00000001h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
RSVD																															MXVM

Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	1h RW	**MSI-X vector Mask (MXVM):** When this bit is set, the function is prohibited from sending a message using this MSI-X Table entry. However, any other MSI-X Table entries programmed with the same vector will still be capable of sending an equivalent message unless they are also masked. This bit's state after reset is 1 (entry is masked).

24.9.6 SATA MXPBA Registers Summary

Table 24-10. Summary of SATA MXPBA Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	MSI-X Pending Bit Array QW 0 (MXPQW0_DW0)—Offset 0h	0h

24.9.6.1 MSI-X Pending Bit Array QW 0 (MXPQW0_DW0)—Offset 0h

Type: MEM Register
(Size: 32 bits)

Device: 17
Function: 5

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																															MXVP

Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h RO	**MSI-X vector Pending (MXVP):** For each Pending Bit that is set, the function has a pending message for the associated MSI-X Table entry. Pending bits that have no associated MSI-X Table entry are reserved. After reset, the state of reserved Pending bits must be 0. Software should never write, and should only read Pending Bits. If software writes to Pending Bits, the result is undefined. Each Pending Bit's state after reset is 0 (no message pending).

§

25 System Management Interface and SMLink

25.1 Acronyms

Acronyms	Description
BMC	Baseboard Management Controller

25.2 References

None

25.3 Overview

The PCH provides six SMLink interfaces, SMLink0B and SMLink0-4. These interfaces are intended for system management and are controlled by the Intel ME or IE. See the System Management chapter for more detail.

25.4 Signal Description

Name	Type	Description
INTRUDER#	I	**Intruder Detect**: This signal can be set to disable the system if the box is detected open.
GPP_C4_SML0DATA_IE	I/OD	**System Management Link 0 Data:** SMBus link to external device for transmitting/receiving data. When using the 1 GbE LAN with Intel ME 11, this would be connected to an external PHY. With SPS F/W, this is used to connect to other devices. An external pull-up is required. This can be controller by either the Intel ME or the IE.
GPP_D14_SML0BDATA_IE	I/OD	**Second Instant of System Management Link 0 Data:** SMBus link to external device for transmitting/receiving data. This pin can be controller by either the Intel ME or the IE. External pull-up is required.
GPP_C3_SML0CLK_IE	I/OD	**System Management Link 0 Clock:** SMBus link clock to external device. When using the 1GbE LAN with Intel ME 11, this would be connected to an external PHY. With SPS F/W, this is used to connect to other devices. An external pull-up is required. This can be controller by either the Intel ME or the IE.
GPP_D13_SML0BCLK_IE	I/OD	**Second Instant of System Management Link 0 Clock:**SMBus link clock to external device. This pin can be controller by either the Intel ME or the IE. External pull-up is required.
GPP_C5_SML0ALERT_IE#	I/OD	**SMLink 0 Alert:** Alert signal for SMLink0 External pull-up resistor is required.
GPP_D16_SML0BALERT_IE#	I/OD	**Second Instant of SMlink 0 Alert:** Alert signal for SMLink0B. External pull-up is required.
GPP_C6_SML1CLK_IE	I/OD	**System Management Link 1 Clock:** SMBus link clock to external device. This pin can be controller by either the Intel ME or the IE. External pull-up is required.
GPP_C7_SML1DATA_IE	I/OD	**System Management Link 1 Data:** External pull-up resistor is required. This signal is controlled by either the Intel ME or the IE.

Name	Type	Description
GPP_B23_MEIE_SML1ALERT#_PHOT#	I/OD	**System Management 1 Alert:** Alert for the device on SMLink1. A soft-strap determines the native function SML1ALERT# or PCHHOT# usage. External pull-up resistor is required on this pin.
GPP_H10_SML2CLK_IE	I/OD	**System Management Link 2 Clock:** External pull-up resistor is required. This signal is controlled by either the Intel ME or the IE.
GPP_H11_SML2DATA_IE	I/OD	**System Management Link 2 Data:** External pull-up resistor is required. This signal is controlled by either the Intel ME or the IE.
GPP_H12_SML2ALERT_IE#	I/OD	**System Management Link 2 Alert:** External pull-up resistor is required on this pin. This signal is controlled by either the Intel ME or the IE.
GPP_H13_SML3CLK_IE	I/OD	**System Management Link 3 Clock:** External pull-up resistor is required. This signal is controlled by either the Intel ME or the IE.
GPP_H14_SML3DATA_IE	I/OD	**System Management Link 3 Data:** External pull-up resistor is required. This signal is controlled by either the Intel ME or the IE.
GPP_H15 SML3ALERT_IE#	I/OD	**System Management Link 3 Alert:** External pull-up resistor is required on this pin. This signal is controlled by either the Intel ME or the IE.
GPP_H16 SML4CLK_IE	I/OD	**System Management Link 4 Clock:** External pull-up resistor is required. This signal is controlled by either the Intel ME or the IE.
GPP_H17_SML4DATA_IE	I/OD	**System Management Link 4 Data:** External pull-up resistor is required. This signal is controlled by either the Intel ME or the IE.
GPP_H18_SML4ALERT_IE#	I/OD	**System Management Link 4 Alert:** External pull-up resistor is required. This signal is controlled by either the Intel ME or the IE.

25.5 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
SML[2:0]ALERT#	Pull-down	9K - 50K	The internal pull-down resistor is disable after RSMRST# de-asserted.
Note: Integrated pull-ups and pull-downs will not be valid until all the voltages have reached a valid level.			

25.6 I/O Signal Planes and States

Signal Name	Power Plane	During Reset	Immediately After Reset	S3/S4/S5	Deep Sx
SML[4:0]DATA SML0BDATA	Primary	Undriven	Undriven	Undriven	Off
SML[4:0]CLK SML0BCLK	Primary	Undriven	Undriven	Undriven	Off
SML0BALERT# SML[3:5]ALERT#	Primary	Undriven	Undriven	Undriven	Off
SML[2:0]ALERT#	Primary	Internal Pull-down	Driven Low	Internal Pull-down	Off

25.7 Functional Description

The PCH provides six SMLink interfaces, SMLink0B and SMLink0-4. These interfaces are intended for system management and are controlled by the Intel ME or IE. See the System Management chapter for more detail.

With the addition of three more SMlink buses, there are three more SMT controllers.

SMLink0 is used for integrated LAN, connected to the external PHY, when using Intel ME 11. If SPS-4 is used, the SMLink0 is used as a connection to the external BMC. When an Intel LAN PHY is connected to SMLink0, a soft strap must be set to indicate that the PHY is connected to SMLink0. The interface will be running at the frequency of up to 1 MHz depending on different factors such as board routing or bus loading when the Fast Mode is enabled using a soft strap. Refer to the latest SPI Programming Guide for more detail.

All six SMlink bus can be accessed by either ME or IE. SMT2 has the option for three connections, Smlink0, Smlink0B and the TCO slave. It is the only SMT controller that can do so. SMT2 can not simultaneously connect to SMlink0 and Smlink0B. The current SPS driver only supports connecting SMT2 to SMlink0, not SMlink0B. This permits IE to use SMLink0B

25.8 SMBus TCO I/O Registers Summary

The TCO I/O registers reside in a 32-byte range that starts from the I/O Base Address described in the TCOBAR register in the SMBus PCI Configuration space.

Table 25-1. Summary of SMBus TCO I/O Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	1h	TCO_RLD Register (TRLD)—Offset 0h	4h
2h	2h	TCO_DAT_IN Register (TDI)—Offset 2h	0h
3h	3h	TCO_DAT_OUT Register (TDO)—Offset 3h	0h
4h	5h	TCO1_STS Register (TSTS1)—Offset 4h	0h
6h	7h	TCO2_STS Register (TSTS2)—Offset 6h	0h
8h	9h	TCO1_CNT Register (TCTL1)—Offset 8h	0h
Ah	Bh	TCO2_CNT Register (TCTL2)—Offset Ah	8h
Ch	Dh	TCO_MESSAGE1 and TCO_MESSAGE2 (TMSG)—Offset Ch	0h
Eh	Eh	TCO_WDSTATUS Register (TWDS)—Offset Eh	0h
10h	10h	LEGACY_ELIM Register (LE)—Offset 10h	3h
12h	13h	TCO_TMR Register (TTMR)—Offset 12h	4h

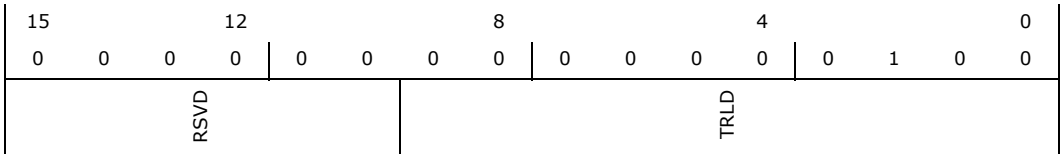
25.8.1 TCO_RLD Register (TRLD)—Offset 0h

Access Method

Type: I/O Register
(Size: 16 bits)

Device: 31
Function: 4

Default: 4h



Bit Range	Default and Access	Field Name (ID): Description
15:10	0h RO	Reserved.
9:0	4h RW	**TCORLD (TRLD):** Reading this register will return the current count of the TCO timer. Writing any value to this register will reload the timer to prevent the timeout.

25.8.2 TCO_DAT_IN Register (TDI)—Offset 2h

Access Method

Type: I/O Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
TDI							

Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW	**TCO_DAT_IN (TDI):** Data Register for passing commands from the OS to the SMI handler. Writes to this register will cause an SMI and set the OS_TCO_SMI bit in the TCO_STS register.

25.8.3 TCO_DAT_OUT Register (TDO)—Offset 3h

Access Method

Type: I/O Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
TDO							

Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW	**TCO_DAT_OUT (TDO):** Data Register for passing commands from the SMI handler to the OS. Writes to this register will set the TCO_INT_STS bit in the TCO_STS register. It also causes an interrupt, as selected by the TCO_IRQ_SEL bits.

25.8.4 TCO1_STS Register (TSTS1)—Offset 4h

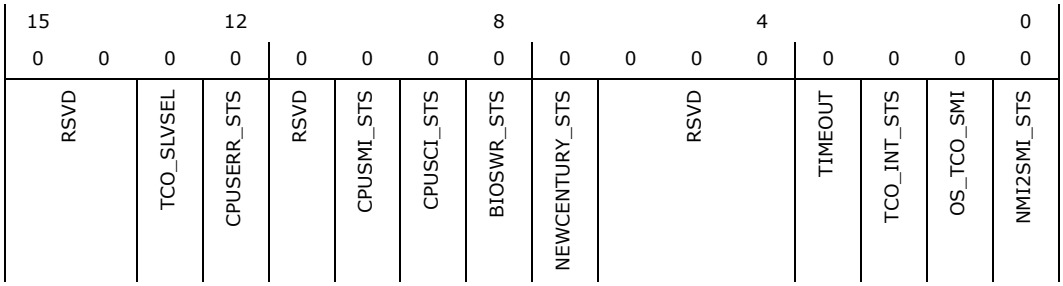
Unless otherwise indicated, these bits are sticky and are cleared by writing a 1 to the corresponding bit position.

Access Method

Type: I/O Register
(Size: 16 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:14	0h RO	Reserved.
13	0h RO/Strap	**TCO Slave Select (TCO_SLVSEL):** This register bit indicates the value of TCO Slave Select Soft Strap.
12	0h RW/1C	**CPUSERR_STS (CPUSERR_STS):** This bit is set to 1 if the CPU complex sends a DMI special cycle message indicating that it wants to cause an SERR#. The software must read the MCH to find out why it wanted the SERR#. Software must write a 1 back to this bit to clear it.
11	0h RO	Reserved.
10	0h RW/1C	**CPUSMI_STS (CPUSMI_STS):** This bit is set to 1 if the CPU complex sends a DMI special cycle message indicating that it wants to cause an SMI. The software must read the CPU to find out why it wanted the SMI. Software must write a 1 back to this bit to clear it.
9	0h RW/1C	**CPUSCI_STS:** This bit is set to 1 if the CPU complex sends a DMI special cycle message indicating that it wants to cause an SCI. The software must read the CPU to find out why it wanted the SCI. Software must write a 1 back to this bit to clear it.
8	0h RW/1C	**BIOSWR_STS:** Intel PCH sets this bit to 1 and generates an SMI# to indicate an illegal attempt to write to the BIOS located in the FWH that is accessed over the LPC. This occurs when either: a. The BIOSWP bit is changed from 0 to 1 and the LE bit is also set, or b. Any write is attempted to the BIOS and the BIOSWP bit is also set. This bit does not get set to 1 when: 1. a or b above occurs on eSPI controller. 2. a or b above occurs on SPI Flash controller. ***Note:*** On write cycles attempted to the 4 MB lower alias to the BIOS space, the BIOSWR_STS bit will not be set.

Bit Range	Default and Access	Field Name (ID): Description
7	0h RW/1C	**NEWCENTURY_STS (NEWCENTURY_STS):** This bit will be set when the year rolls over from 1999 to 2000. If the bit is already 1, it will remain 1. This bit can be cleared either by software writing a 1 back to the bit position, or by RTCRST# going active. When this bit is set, an SMI# will be generated. However, this will not be a wake event (i.e., if the system is in a sleeping state when the NEWCENTURY_STS bit is set, the system will not wake up). ***Notes:*** This bit 7 is not valid when the RTC battery is first put in (or if the RTC battery does not provide sufficient power when the system is unplugged). Software can determine that the RTC well was not maintained by checking the RTC_PWR_STS bit (GEN_PMCON_3 register in the Power Management Controller, D31:F2:A4, bit 2) or by other means (such as doing a checksum on the RTC RAM array). If the RTC well is determined to not have been maintained, the BIOS should set the time to a legal value and then clear the NEWCENTURY_STS bit. This bit may take up to 3 RTCCLKs for the bit to be cleared when a 1 is written to the bit to clear it. After writing a 1 to the NEWCENTURY_STS bit, software should also not exit the SMI handler until after the bit has been cleared. This is to make sure the SMI is not re-entered. BIOS Assumption: When booting, the BIOS checks the NEWCENTURY_STS bit. If set, the BIOS should increment the value in the RTC RAM register associated with the century. The BIOS should then clear the NEWCENTURY_STS bit. This scenario would occur if the system was asleep when the century rolls over. If the system is in an S0 state (not sleeping) and the SMI# occurs with the NEWCENTURY_STS bit sets, the SMI handler should increment the value in the RTC RAM register and clear the NEWCENTURY_STS bit.
6:4	0h RO	Reserved.
3	0h RW/1C	**TIMEOUT (TIMEOUT):** Bit set to 1 by Intel PCH to indicate that the SMI was caused by TCO timer reaching 0. ***Note:*** The SMI handler should clear this bit to prevent an immediate re-entry to the SMI handler.
2	0h RW/1C	**TCO_INT_STS (TCO_INT_STS):** Bit set to 1 when SMI handler caused the interrupt by writing to the TCO_DAT_OUT register.
1	0h RW/1C	**OS_TCO_SMI:** Bit set to 1 when OS code caused an SMI# by writing to the TCO_DAT_IN register.
0	0h RO/V	**NMI2SMI_STS:** The PCH sets this bit when an SMI# occurs because an event occurred that would otherwise have caused an NMI.

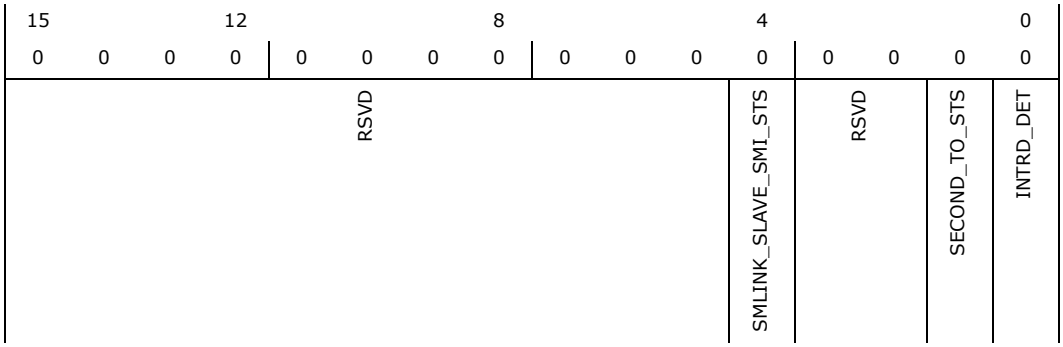
25.8.5 TCO2_STS Register (TSTS2)—Offset 6h

Access Method

Type: I/O Register
(Size: 16 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:5	0h RO	Reserved.
4	0h RW/1C	**SMLINK_SLAVE_SMI_STS:** The PCH will set this bit to 1 when it receives the SMI message (encoding 08h in the command type) on the SMLinks Slave Interface. Software clears the bit by writing a 1 to this bit position. This bit is in the resume well. It is reset by RSMRST#, but not by the PCI Reset associated with exit from S3-S5 states. This allows the software (presumably BIOS) to get the interrupt, see this new bit set, and decidedly go into the pre-determined (by local policy) sleep state.
3:2	0h RO	Reserved.
1	0h RW/1C	**SECOND_TO_STS:** Intel PCH sets this bit to 1 to indicate that the TIMEOUT bit had been (or is currently) set and a second timeout occurred before the TCO_RLD register was written. If this bit is set and the NO_REBOOT config bit is 0, then the Intel PCH will reboot the system after the second timeout. The reboot is done by asserting PLTRST#. This bit is only cleared by writing a 1 to this bit or by a RSMRST#.
0	0h RW/1C	**INTRD_DET (INTRD_DET):** The bit is set to 1 by the PCH to indicate that an intrusion was detected. This bit is cleared by writing a 1 to this bit or by RTCRST#. ***Notes:*** This bit has a recovery time. After writing a 1 to this bit position (to clear it), the bit may be read back as a 1 for up 65 microseconds before it is read as a 0. Software must be aware of this recovery time when reading this bit after clearing it. If the INTRUDER# signal is active when the software attempts to clear the INTRD_DET bit, the bit will remain as a 1, and the SMI# will be generated again immediately. The SMI handler can clear the INTRD_SEL bits to avoid further SMIs. However, if the INTRUDER# signals goes inactive and then active again, there will not be further SMIs. If the INTRUDER# signal goes inactive some point after the INTRD_DET bit is written as a 1, then the INTRD_DET signal will go to a 0 when INTRUDER# input signal goes inactive. Note that this is slightly different than a classic sticky bit, since most sticky bits would remain active indefinitely when the signal goes active and would immediately go inactive when a 1 is written to the bit.

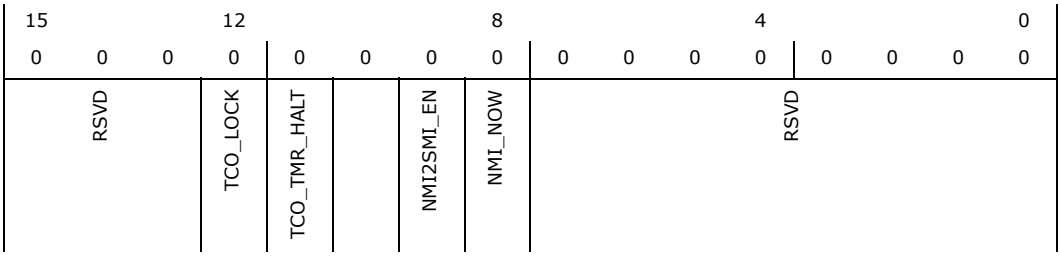
25.8.6 TCO1_CNT Register (TCTL1)—Offset 8h

Access Method

Type: I/O Register
(Size: 16 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:13	0h RO	Reserved.
12	0h RW	**TCO_LOCK:** When set to 1, this bit prevents writes from changing the TCO_EN bit (in offset 30h of Power Management I/O space). Once this bit is set to 1, it can not be cleared by software writing a 0 to this bit location. A core-well reset is required to change this bit from 1 to 0. This bit defaults to 0.
11	0h RW	**TCO_TMR_HALT:** 1 = The TCO timer will halt. It will not count, and thus cannot reach a value that would cause an SMI# or to cause the SECOND_TO_STS bit to be set. This will also prevent rebooting. 0 = The TCO timer is enabled to count. This is the default.
10	0h RW	**Reserved**
9	0h RW	**NMI2SMI_EN:** Setting this bit 1 forces all NMIs to instead cause an SMI#, and will be reported in the TCO1_STS register. NMI2SMI_EN bit is set AND the NMI_EN# bit is set to 0, the NMI# will be routed to cause an SMI#. No NMI will be caused. However, if the GBL_SMI_EN bit is not set, then no SMI# will be generated, either. If NMI2SMI_EN is set but the NMI_EN# bit is set to 1, then no NMI or SMI# will be generated. The following table shows the possible combinations: NMI_EN#, GBL_SMI_EN 00: No SMI# based on NMI events (since no SMI# at all because SMI_EN = 0) 01: SMI# will be caused based on NMI events 10: No SMI# at all because SMI_EN is 0 11: No SMI# based on NMI events because NMI_EN#=1
8	0h RW	**NMI_NOW:** Writing a 1 to this bit causes an NMI. This allows the BIOS or SMI handler to force entry to the NMI handler. The NMI handler is expected to clear this bit. Another NMI will not be generated until the bit is cleared by writing a 1 back to the same bit position.
7:0	0h RO	Reserved.

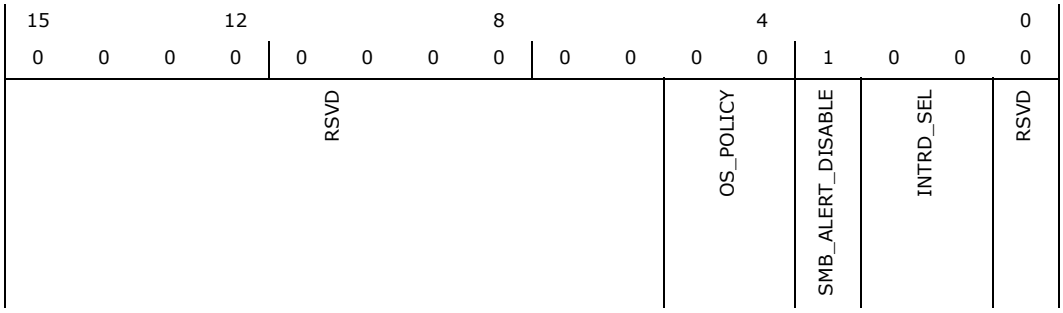
25.8.7 TCO2_CNT Register (TCTL2)—Offset Ah

Access Method

Type: I/O Register
(Size: 16 bits)

Device: 31
Function: 4

Default: 8h



Bit Range	Default and Access	Field Name (ID): Description
15:6	0h RO	Reserved.
5:4	0h RW	**OS_POLICY (OS_POLICY):** OS-based software writes to these bits to select the policy that the BIOS will use after the platform resets due the WDT. The following convention is recommended for the BIOS and OS: 00 Boot normally 01 Shut down 10 Do not load OS. Hold in pre-boot state and use LAN to determine next step 11 Reserved Implementation. ***Note:*** These are just scratch pad bits. They should not be reset when the TCO logic resets the platform due to Watchdog Timer.
3	1h RW	**SMB_ALERT_DISABLE:** Disables muxed GPIO/SMBALERT# signal as an alert source for the heartbeats and the SMBus slave. At reset (RSMRST# pin assertion only), this bit is set and the muxed GPIO/SMBALERT# alerts are disabled.
2:1	0h RW	**INTRD_SEL (INTRD_SEL):** Selects the action to take if the INTRUDER# signal goes active. 11: Reserved 01: Interrupt (as selected by TCO_INT_SEL). 10: SMI# 00 INTRUDER# doesn't cause SMI# or interrupt
0	0h RO	Reserved.

25.8.8 TCO_MESSAGE1 and TCO_MESSAGE2 (TMSG)—Offset Ch

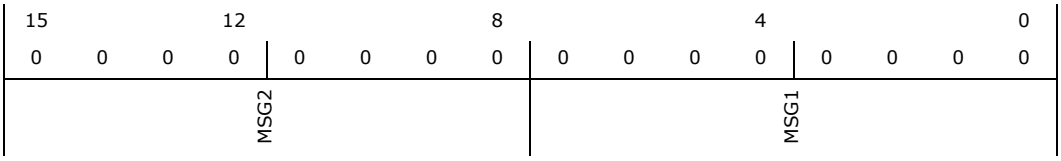
TCOBASE+0Ch (MSG1) TCOBASE+0Dh (MSG2) BIOS can write into these registers to indicate its boot progress. The external microcontroller can read these registers to monitor the boot progress.

Access Method

Type: I/O Register
(Size: 16 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:8	0h RW	**TCO_MESSAGE2 (MSG2)**
7:0	0h RW	**TCO_MESSAGE1 (MSG1)**

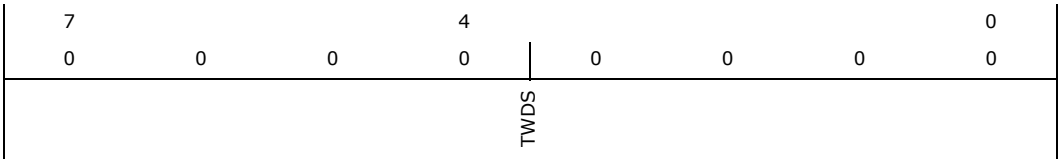
25.8.9 TCO_WDSTATUS Register (TWDS)—Offset Eh

Access Method

Type: I/O Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW	**TCO_WDSTATUS Register (TWDS):** The BIOS or system management software can write into this register to indicate more details on the boot progress. The register will rest to 00h based on a RSMRST# (but not PCI Reset). The external microcontroller can read this register to monitor boot progress.

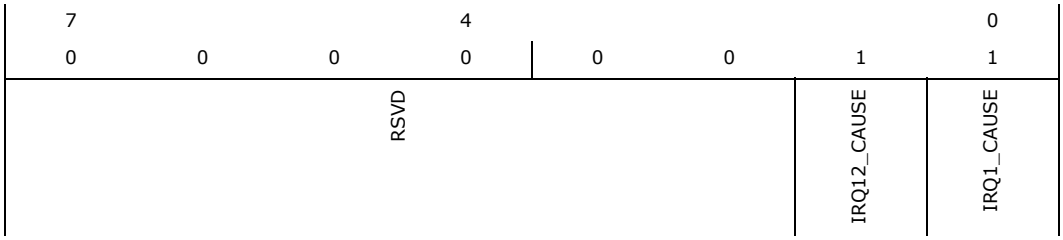
25.8.10 LEGACY_ELIM Register (LE)—Offset 10h

Access Method

Type: I/O Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 3h



Bit Range	Default and Access	Field Name (ID): Description
7:2	0h RO	Reserved.
1	1h RW	**IRQ12_CAUSE (IRQ12_CAUSE):** When software sets the bit to 1, IRQ12 will be high (asserted). When software sets the bit to 0, IRQ12 will be low (not asserted). Default for this bit is 1.
0	1h RW	**IRQ1_CAUSE (IRQ1_CAUSE):** When software sets the bit to 1, IRQ1 will be high (asserted). When software sets the bit to 0, IRQ1 will be low (not asserted). Default for this bit is 1.

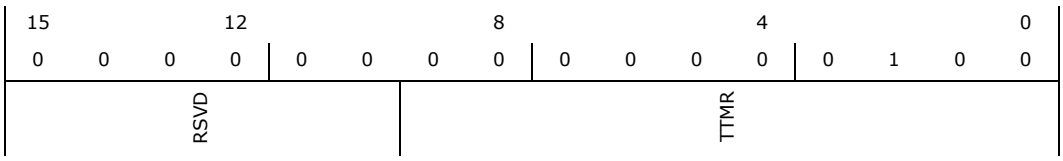
25.8.11 TCO_TMR Register (TTMR)—Offset 12h

Access Method

Type: I/O Register
(Size: 16 bits)

Device: 31
Function: 4

Default: 4h



Bit Range	Default and Access	Field Name (ID): Description
15:10	0h RO	Reserved.
9:0	4h RW	**TCOTMR (TTMR):** Value that is loaded into the timer each time the TCO_RLD register is written. Values of 0000h or 0001h will be ignored and should not be attempted. The timer is clocked at approximately 0.6 seconds, and thus allows timeouts ranging from 1.2 second to 613.8 seconds. ***Note:*** The timer has an error of ± 1 tick (0.6s).

§

26 Host System Management Bus Controller

26.1 Acronyms

Acronyms	Description
ARP	Address Resolution Protocol
CRC	Cyclic Redundancy Check
PEC	Package Error Checking
SMBus	System Management Bus
INTR	Interrupt Status Message

26.2 References

Specification	Location
System Management Bus (SMBus) Specification, Version 2.0	http://www.smbus.org/specs/

26.3 Overview

The PCH provides a System Management Bus 2.0 host controller as well as an SMBus Slave Interface. The PCH is also capable of operating in a mode in which it can communicate with I^2C compatible devices.

The host SMBus controller supports up to 100 kHz clock speed.

26.4 Signal Description

Name	Type	Description
GPP_C0_SMBCLK	I/OD	**SMBus Clock.** External pull-up resistor is required.
GPP_C1_SMBDATA	I/OD	**SMBus Data.** External pull-up resistor is required.
GPP_C2_SMBALERT#	I/OD	**SMBus Alert:** This signal is used to wake the system or generate SMI#. External pull-up resistor is required.

26.5 Integrated Pull-ups and Pull-downs

Signal	Resistor Type	Value	Notes
SMBALERT#	Pull-Down	9K - 50K	The integrated pull down is disabled after RSMRST# de-assertion.
Note: Integrated pull-ups and pull-downs will not be valid until all the voltages have reached a valid level.			

26.6 I/O Signal Planes and States

Signal Name	Power Plane	During Reset	Immediately After Reset	S3/S4/S5	Deep Sx
SMBDATA	Primary	Undriven	Undriven	Undriven	Off
SMBCLK	Primary	Undriven	Undriven	Undriven	Off
SMBALERT#	Primary	Internal Pull-down	Driven Low	Internal Pull-down	Off

26.7 Functional Description

The PCH provides an System Management Bus 2.0 host controller as well as an SMBus Slave Interface.

- **Host Controller:** Provides a mechanism for the processor to initiate communications with SMBus peripherals (slaves). The PCH is also capable of operating in a mode in which it can communicate with I^2C compatible devices.
- **Slave Interface:** Allows an external master to read from or write to the PCH. Write cycles can be used to cause certain events or pass messages, and the read cycles can be used to determine the state of various status bits. The PCH's internal host controller cannot access the PCH's internal Slave Interface.

26.7.1 Host Controller

The host SMBus controller supports up to 100 kHz clock speed and is clocked by the RTC clock.

The PCH can perform SMBus messages with either Packet Error Checking (PEC) enabled or disabled. The actual PEC calculation and checking is performed in SW. The SMBus host controller logic can automatically append the CRC byte if configured to do so.

The SMBus Address Resolution Protocol is supported by using the existing host controller commands through software, except for the Host Notify command (which is actually a received message).

The programming model of the host controller is combined into two portions: a PCI configuration portion, and a system I/O mapped portion. All static configurations, such as the I/O base address, is done using the PCI configuration space. Real-time programming of the Host interface is done in system I/O space.

The PCH SMBus host controller checks for parity errors as a target. If an error is detected, the detected parity error bit in the PCI Status Register is set. If bit 6 and bit 8 of the PCI Command Register are set, an SERR# is generated and the signaled SERR# bit in the PCI Status Register is set.

26.7.1.1 Host Controller Operation Overview

The SMBus host controller is used to send commands to other SMBus slave devices. Software sets up the host controller with an address, command, and, for writes, data and optional PEC; and then tells the controller to start. When the controller has finished transmitting data on writes, or receiving data on reads, it generates an SMI# or interrupt, if enabled.

The host controller supports eight command protocols of the SMBus interface (see *System Management Bus (SMBus) Specification*, Version 2.0): Quick Command, Send Byte, Receive Byte, Write Byte/Word, Read Byte/Word, Process Call, Block Read/Write, Block Write–Block Read Process Call, and Host Notify.

The SMBus host controller requires that the various data and command fields be setup for the type of command to be sent. When software sets the START bit, the SMBus Host controller performs the requested transaction, and interrupts the processor (or generates an SMI#) when the transaction is completed. Once a START command has been issued, the values of the "active registers" (Host Control, Host Command, Transmit Slave Address, Data 0, Data 1) should not be changed or read until the interrupt status message bit has been set (indicating the completion of the command). Any register values needed for computation purposes should be saved prior to issuing of a new command, as the SMBus host controller updates all registers while completing the new command.

Slave functionality, including the Host Notify protocol, is available on the SMBus pins.

Using the SMB host controller to send commands to the PCH SMB slave port is not supported.

26.7.1.2 Command Protocols

In all of the following commands, the Host Status Register (offset 00h) is used to determine the progress of the command. While the command is in operation, the HOST_BUSY bit is set. If the command completes successfully, the INTR bit will be set in the Host Status Register. If the device does not respond with an acknowledge, and the transaction times out, the DEV_ERR bit is set.

If software sets the KILL bit in the Host Control Register while the command is running, the transaction will stop and the FAILED bit will be set after the PCH forces a time-out. In addition, if KILL bit is set during the CRC cycle, both the CRCE and DEV_ERR bits will also be set.

Quick Command

When programmed for a Quick Command, the Transmit Slave Address Register is sent. The PEC byte is never appended to the Quick Protocol. Software should force the PEC_EN bit to 0 when performing the Quick Command. Software must force the I2C_EN bit to 0 when running this command. See Section 5.5.1 of the *System Management Bus (SMBus) Specification,* Version 2.0 for the format of the protocol.

Send Byte/Receive Byte

For the Send Byte command, the Transmit Slave Address and Device Command Registers are sent. For the Receive Byte command, the Transmit Slave Address Register is sent. The data received is stored in the DATA0 register. Software must force the I2C_EN bit to 0 when running this command.

The Receive Byte is similar to a Send Byte, the only difference is the direction of data transfer. See Sections 5.5.2 and 5.5.3 of the *System Management Bus (SMBus) Specification,* Version 2.0 for the format of the protocol.

Write Byte/Word

The first byte of a Write Byte/Word access is the command code. The next one or two bytes are the data to be written. When programmed for a Write Byte/Word command, the Transmit Slave Address, Device Command, and Data0 Registers are sent. In

addition, the Data1 Register is sent on a Write Word command. Software must force the I2C_EN bit to 0 when running this command. See Section 5.5.4 of the *System Management Bus (SMBus) Specification,* Version 2.0 for the format of the protocol.

Read Byte/Word

Reading data is slightly more complicated than writing data. First the PCH must write a command to the slave device. Then it must follow that command with a repeated start condition to denote a read from that device's address. The slave then returns one or two bytes of data. Software must force the I2C_EN bit to 0 when running this command.

When programmed for the read byte/word command, the Transmit Slave Address and Device Command Registers are sent. Data is received into the DATA0 on the read byte, and the DAT0 and DATA1 registers on the read word. See Section 5.5.5 of the *System Management Bus (SMBus) Specification,* Version 2.0 for the format of the protocol.

Process Call

The process call is so named because a command sends data and waits for the slave to return a value dependent on that data. The protocol is simply a Write Word followed by a Read Word, but without a second command or stop condition.

When programmed for the Process Call command, the PCH transmits the Transmit Slave Address, Host Command, DATA0 and DATA1 registers. Data received from the device is stored in the DATA0 and DATA1 registers.

The Process Call command with I2C_EN set and the PEC_EN bit set produces undefined results. Software must force either I2C_EN or PEC_EN to 0 when running this command. See Section 5.5.6 of the *System Management Bus (SMBus) Specification,* Version 2.0 for the format of the protocol.

Note: For process call command, the value written into bit 0 of the Transmit Slave Address Register needs to be 0.

Note: If the I2C_EN bit is set, the protocol sequence changes slightly: the Command Code (Bits [18:11] in the bit sequence) are not sent - as a result, the slave will not acknowledge (Bit 19 in the sequence).

Block Read/Write

The PCH contains a 32-byte buffer for read and write data which can be enabled by setting bit 1 of the Auxiliary Control register at offset 0Dh in I/O space, as opposed to a single byte of buffering. This 32-byte buffer is filled with write data before transmission, and filled with read data on reception. In the PCH, the interrupt is generated only after a transmission or reception of 32 bytes, or when the entire byte count has been transmitted/received.

The byte count field is transmitted but ignored by the PCH as software will end the transfer after all bytes it cares about have been sent or received.

For a Block Write, software must either force the I2C_EN bit or both the PEC_EN and AAC bits to 0 when running this command.

The block write begins with a slave address and a write condition. After the command code the PCH issues a byte count describing how many more bytes will follow in the message. If a slave had 20 bytes to send, the first byte would be the number 20 (14h), followed by 20 bytes of data. The byte count may not be 0. A Block Read or Write is allowed to transfer a maximum of 32 data bytes.

When programmed for a block write command, the Transmit Slave Address, Device Command, and Data0 (count) registers are sent. Data is then sent from the Block Data Byte register; the total data sent being the value stored in the Data0 Register.

On block read commands, the first byte received is stored in the Data0 register, and the remaining bytes are stored in the Block Data Byte register. See Section 5.5.7 of the *System Management Bus (SMBus) Specification,* Version 2.0 for the format of the protocol.

Note: For Block Write, if the I2C_EN bit is set, the format of the command changes slightly. The PCH will still send the number of bytes (on writes) or receive the number of bytes (on reads) indicated in the DATA0 register. However, it will not send the contents of the DATA0 register as part of the message. Also, the Block Write protocol sequence changes slightly: the Byte Count (bits 27:20 in the bit sequence) are not sent - as a result, the slave will not acknowledge (bit 28 in the sequence).

Note: When operating in I^2C mode (I2C_EN bit is set), the PCH will never use the 32-byte buffer for any block commands.

I^2C* Read

This command allows the PCH to perform block reads to certain I^2C devices, such as serial E^2PROMs. The SMBus Block Read supports the 7-bit addressing mode only.

However, this does not allow access to devices using the I^2C "Combined Format" that has data bytes after the address. Typically these data bytes correspond to an offset (address) within the serial memory chips.

Note: This command is supported independent of the setting of the I2C_EN bit. The I^2C Read command with the PEC_EN bit set produces undefined results. Software must force both the PEC_EN and AAC bit to 0 when running this command.

For I^2C Read command, the value written into bit 0 of the Transmit Slave Address Register (SMB I/O register, offset 04h) needs to be 0.

The format that is used for the command is shown in Table 26-1.

Table 26-1. I^2C* Block Read (Sheet 1 of 2)

Bit	Description
1	Start
8:2	Slave Address – 7 bits
9	Write
10	Acknowledge from slave
18:11	Send DATA1 register
19	Acknowledge from slave
20	Repeated Start
27:21	Slave Address – 7 bits
28	Read
29	Acknowledge from slave
37:30	Data byte 1 from slave – 8 bits
38	Acknowledge
46:39	Data byte 2 from slave – 8 bits
47	Acknowledge
–	Data bytes from slave/Acknowledge

Table 26-1. I²C* Block Read (Sheet 2 of 2)

Bit	Description
–	Data byte N from slave – 8 bits
–	NOT Acknowledge
–	Stop

The PCH will continue reading data from the peripheral until the NAK is received.

Block Write–Block Read Process Call

The block write-block read process call is a two-part message. The call begins with a slave address and a write condition. After the command code the host issues a write byte count (M) that describes how many more bytes will be written in the first part of the message. If a master has six bytes to send, the byte count field will have the value 6 (0000 0110b), followed by the six bytes of data. The write byte count (M) cannot be 0.

The second part of the message is a block of read data beginning with a repeated start condition followed by the slave address and a Read bit. The next byte is the read byte count (N), which may differ from the write byte count (M). The read byte count (N) cannot be 0.

The combined data payload must not exceed 32 bytes. The byte length restrictions of this process call are summarized as follows:

- M ≥ 1 byte
- N ≥ 1 byte
- M + N ≤ 32 bytes

The read byte count does not include the PEC byte. The PEC is computed on the total message beginning with the first slave address and using the normal PEC computational rules. It is highly recommended that a PEC byte be used with the Block Write-Block Read Process Call. Software must do a read to the command register (offset 2h) to reset the 32 byte buffer pointer prior to reading the block data register.

Note: There is no STOP condition before the repeated START condition, and that a NACK signifies the end of the read transfer.

Note: E32B bit in the Auxiliary Control register must be set when using this protocol.

See Section 5.5.8 of the *System Management Bus (SMBus) Specification,* Version 2.0 for the format of the protocol.

26.7.1.3 Bus Arbitration

Several masters may attempt to get on the bus at the same time by driving the SMBDATA line low to signal a start condition. The PCH continuously monitors the SMBDATA line. When the PCH is attempting to drive the bus to a 1 by letting go of the SMBDATA line, and it samples SMBDATA low, then some other master is driving the bus and the PCH will stop transferring data.

If the PCH sees that it has lost arbitration, the condition is called a collision. The PCH will set the BUS_ERR bit in the Host Status Register, and if enabled, generate an interrupt or SMI#. The processor is responsible for restarting the transaction.

When the PCH is a SMBus master, it drives the clock. When the PCH is sending address or command as an SMBus master, or data bytes as a master on writes, it drives data relative to the clock it is also driving. It will not start toggling the clock until the start or stop condition meets proper setup and hold time. The PCH will also ensure minimum time between SMBus transactions as a master.

Note: The PCH supports the same arbitration protocol for both the SMBus and the System Management (SMLink) interfaces.

26.7.1.4 Clock Stretching

Some devices may not be able to handle their clock toggling at the rate that the PCH as an SMBus master would like. They have the capability of stretching the low time of the clock. When the PCH attempts to release the clock (allowing the clock to go high), the clock will remain low for an extended period of time.

The PCH monitors the SMBus clock line after it releases the bus to determine whether to enable the counter for the high time of the clock. While the bus is still low, the high time counter must not be enabled. Similarly, the low period of the clock can be stretched by an SMBus master if it is not ready to send or receive data.

26.7.1.5 Bus Timeout (PCH as SMBus Master)

If there is an error in the transaction, such that an SMBus device does not signal an acknowledge, or holds the clock lower than the allowed Timeout time, the transaction will time out. The PCH will discard the cycle and set the DEV_ERR bit. The timeout minimum is 25 ms (800 RTC clocks). The Timeout counter inside the PCH will start after the last bit of data is transferred by the PCH and it is waiting for a response.

The 25 ms Timeout counter will not count under the following conditions:

1. BYTE_DONE_STATUS bit (SMBus I/O Offset 00h, Bit 7) is set
2. The SECOND_TO_STS bit (TCO I/O Offset 06h, Bit 1) is not set (this indicates that the system has not locked up).

26.7.1.6 Interrupts/SMI#

The PCH SMBus controller uses PIRQB# as its interrupt pin. However, the system can alternatively be set up to generate SMI# instead of an interrupt, by setting the SMBUS_SMI_EN bit.

Table 26-2, Table 26-3 and Table 26-4 specify how the various enable bits in the SMBus function control the generation of the interrupt, Host and Slave SMI, and Wake internal signals. The rows in the tables are additive, which means that if more than one row is true for a particular scenario then the results for all of the activated rows will occur.

Table 26-2. Enable for SMBALERT#

Event	INTREN (Host Control I/O Register, Offset 02h, Bit 0)	SMB_SMI_EN (Host Configuration Register, D31:F4:Offset 40h, Bit 1)	SMBALERT_DIS (Slave Command I/O Register, Offset 11h, Bit 2)	Result
SMBALERT# asserted low (always reported in Host Status Register, Bit 5)	X	X	X	Wake generated
	X	1	0	Slave SMI# generated (SMBUS_SMI_STS)
	1	0	0	Interrupt generated

Table 26-3. Enables for SMBus Slave Write and SMBus Host Events

Event	INTREN (Host Control I/O Register, Offset 02h, Bit 0)	SMB_SMI_EN (Host Configuration Register, D31:F3:Offset 40h, Bit 1)	Event
Slave Write to Wake/SMI# Command	X	X	Wake generated when asleep. Slave SMI# generated when awake (SMBUS_SMI_STS).
Slave Write to SMLINK_SLAVE_SMI Command	X	X	Slave SMI# generated when in the S0 state (SMBUS_SMI_STS)
Any combination of Host Status Register [4:1] asserted	0	X	None
	1	0	Interrupt generated
	1	1	Host SMI# generated

Table 26-4. Enables for the Host Notify Command

HOST_NOTIFY_INTREN (Slave Control I/O Register, Offset 11h, Bit 0)	SMB_SMI_EN (Host Config Register, D31:F4:Off40h, Bit 1)	HOST_NOTIFY_WKEN (Slave Control I/O Register, Offset 11h, Bit 1)	Result
0	X	0	None
X	X	1	Wake generated
1	0	X	Interrupt generated
1	1	X	Slave SMI# generated (SMBUS_SMI_STS)

26.7.1.7 SMBus CRC Generation and Checking

If the AAC bit is set in the Auxiliary Control register, the PCH automatically calculates and drives CRC at the end of the transmitted packet for write cycles, and will check the CRC for read cycles. It will not transmit the contents of the PEC register for CRC. The PEC bit must not be set in the Host Control register if this bit is set, or unspecified behavior will result.

If the read cycle results in a CRC error, the DEV_ERR bit and the CRCE bit in the Auxiliary Status register at Offset 0Ch will be set.

26.7.2 SMBus Slave Interface

The PCH SMBus Slave interface is accessed using the SMBus. The SMBus slave logic will not generate or handle receiving the PEC byte and will only act as a Legacy Alerting Protocol device. The slave interface allows the PCH to decode cycles, and allows an external microcontroller to perform specific actions.

Key features and capabilities include:

- Supports decode of three types of messages: Byte Write, Byte Read, and Host Notify.
- Receive Slave Address register: This is the address that the PCH decodes. A default value is provided so that the slave interface can be used without the processor having to program this register.
- Receive Slave Data register in the SMBus I/O space that includes the data written by the external microcontroller.
- Registers that the external microcontroller can read to get the state of the PCH.

- Status bits to indicate that the SMBus slave logic caused an interrupt or SMI# due to the reception of a message that matched the slave address.
  - Bit 0 of the Slave Status Register for the Host Notify command
  - Bit 16 of the SMI Status Register for all others

Note: The external microcontroller should not attempt to access the PCH SMBus slave logic until either:

- 800 milliseconds after both: RTCRST# is high and RSMRST# is high, OR
- The PLTRST# deasserts

If a master leaves the clock and data bits of the SMBus interface at 1 for 50 µs or more in the middle of a cycle, the PCH slave logic's behavior is undefined. This is interpreted as an unexpected idle and should be avoided when performing management activities to the slave logic.

Note: When an external microcontroller accesses the SMBus Slave Interface over the SMBus a translation in the address is needed to accommodate the least significant bit used for read/write control. For example, if the PCH slave address (RCV_SLVA) is left at 44h (default), the external micro controller would use an address of 88h/89h (write/read).

26.7.2.1 Format of Slave Write Cycle

The external master performs Byte Write commands to the PCH SMBus Slave I/F. The "Command" field (bits 11:18) indicate which register is being accessed. The Data field (bits 20:27) indicate the value that should be written to that register.

Table 26-5 has the values associated with the registers.

Table 26-5. Slave Write Registers

Register	Function
0	Command Register. See Table 26-6 for valid values written to this register.
1–3	Reserved
4	Data Message Byte 0
5	Data Message Byte 1
6–7	Reserved
8	Reserved
9–FFh	Reserved
Note: The external microcontroller is responsible to make sure that it does not update the contents of the data byte registers until they have been read by the system processor. The PCH overwrites the old value with any new value received. A race condition is possible where the new value is being written to the register just at the time it is being read. The PCH will not attempt to cover this race condition (that is, unpredictable results in this case).	

Table 26-6. Command Types (Sheet 1 of 2)

Command Type	Description
0	Reserved
1	**WAKE/SMI#.** This command wakes the system if it is not already awake. If system is already awake, an SMI# is generated. ***Note:*** The SMB_WAK_STS bit will be set by this command, even if the system is already awake. The SMI handler should then clear this bit.
2	**Unconditional Power Down.** This command sets the PWRBTNOR_STS bit, and has the same effect as the power button override occurring.

Table 26-6. Command Types (Sheet 2 of 2)

Command Type	Description
3	**HARD RESET WITHOUT CYCLING:** This command causes a hard reset of the system (does not include cycling of the power supply). This is equivalent to a write to the CF9h register with Bits 2:1 set to 1, but Bit 3 set to 0.
4	**HARD RESET SYSTEM.** This command causes a hard reset of the system (including cycling of the power supply). This is equivalent to a write to the CF9h register with Bits 3:1 set to 1.
5	**Disable the TCO Messages.** This command will disable the PCH from sending Heartbeat and Event messages. Once this command has been executed, Heartbeat and Event message reporting can only be re-enabled by assertion and deassertion of the RSMRST# signal.
6	**WD RELOAD:** Reload watchdog timer.
7	Reserved
8	**SMLINK_SLV_SMI.** When the PCH detects this command type while in the S0 state, it sets the SMLINK_SLV_SMI_STS bit. This command should only be used if the system is in an S0 state. If the message is received during S1–S5 states, the PCH acknowledges it, but the SMLINK_SLV_SMI_STS bit does not get set. ***Note:*** It is possible that the system transitions out of the S0 state at the same time that the SMLINK_SLV_SMI command is received. In this case, the SMLINK_SLV_SMI_STS bit may get set but not serviced before the system goes to sleep. Once the system returns to S0, the SMI associated with this bit would then be generated. Software must be able to handle this scenario.
9–FFh	Reserved.

26.7.2.2 Format of Read Command

The external master performs Byte Read commands to the PCH SMBus Slave interface. The "Command" field (bits 18:11) indicate which register is being accessed. The Data field (bits 30:37) contain the value that should be read from that register.

Table 26-7. Slave Read Cycle Format

Bit	Description	Driven By	Comment
1	Start	External Microcontroller	
2–8	Slave Address - 7 bits	External Microcontroller	Must match value in Receive Slave Address register
9	Write	External Microcontroller	Always 0
10	ACK	PCH	
11–18	Command code – 8 bits	External Microcontroller	Indicates which register is being accessed. See Table 26-8 for a list of implemented registers.
19	ACK	PCH	
20	Repeated Start	External Microcontroller	
21–27	Slave Address - 7 bits	External Microcontroller	Must match value in Receive Slave Address register
28	Read	External Microcontroller	Always 1
29	ACK	PCH	
30–37	Data Byte	PCH	Value depends on register being accessed. See Table 26-8 for a list of implemented registers.
38	NOT ACK	External Microcontroller	
39	Stop	External Microcontroller	

Table 26-8. Data Values for Slave Read Registers (Sheet 1 of 2)

Register	Bits	Description
0	7:0	Reserved for capabilities indication. Should always return 00h. Future chips may return another value to indicate different capabilities.
1	2:0	**System Power State** 000 = S0 001 = S1 010 = Reserved 011 = S3 100 = S4 101 = S5 110 = Reserved 111 = Reserved
	7:3	Reserved
2	3:0	Reserved
	7:4	Reserved
3	5:0	**Watchdog Timer current value** ***Note:*** The Watchdog Timer has ten bits, but this field is only six bits. If the current value is greater than 3Fh, the PCH will always report 3Fh in this field.
	7:6	Reserved
4	0	**Intruder Detect**. 1 = The Intruder Detect (INTRD_DET) bit is set. This indicates that the system cover has probably been opened.
	1	**Temperature Event**. 1 = BTI Temperature Event occurred. This bit will be set if the PCH's THRM# input signal is active. Else this bit will read "0."
	2	**DOA Processor Status**. This bit will be 1 to indicate that the processor is dead
	3	1 = **SECOND_TO_STS** bit set. This bit will be set after the second Timeout (SECOND_TO_STS bit) of the Watchdog Timer occurs.
	6:4	Reserved. Will always be 0, but software should ignore.
	7	**SMBALERT# Status.** Reflects the value of the SMBALERT# pin (when the pin is configured to SMBALERT#). Valid only if SMBALERT_DISABLE = 0. Value always returns 1 if SMBALERT_DISABLE = 1.
5	0	**FWH bad bit.** This bit will be 1 to indicate that the FWH read returned FFh, which indicates that it is probably blank.
	1	**Battery Low Status.** Reserved
	2	**SYS_PWROK Failure Status:** This bit will be 1 if the SYSPWR_FLR bit in the GEN_PMCON_2 register is set.
	3	Reserved
	4	Reserved
	5	**POWER_OK_BAD:** Indicates the failure core power well ramp during boot/resume. This bit will be active if the SLP_S3# pin is deasserted and PCH_PWROK pin is not asserted.
	6	**Thermal Trip**: This bit will shadow the state of processor Thermal Trip status bit (CTS). Events on signal will not create a event message
	7	Reserved: Default value is "X" ***Note:*** Software should not expect a consistent value when this bit is read through SMBUS/SMLink
6	7:0	Contents of the Message 1 register.
7	7:0	Contents of the Message 2 register.
8	7:0	Contents of the WDSTATUS register.
9	7:0	Seconds of the RTC
A	7:0	Minutes of the RTC
B	7:0	Hours of the RTC
C	7:0	"Day of Week" of the RTC
D	7:0	"Day of Month" of the RTC
E	7:0	Month of the RTC
F	7:0	Year of the RTC

Table 26-8. Data Values for Slave Read Registers (Sheet 2 of 2)

Register	Bits	Description
10h–FFh	7:0	Reserved

26.7.2.2.1 Behavioral Notes

According to SMBus protocol, Read and Write messages always begin with a Start bit—Address—Write bit sequence. When the PCH detects that the address matches the value in the Receive Slave Address register, it will assume that the protocol is always followed and ignore the Write bit (Bit 9) and signal an Acknowledge during bit 10. In other words, if a Start—Address—Read occurs (which is invalid for SMBus Read or Write protocol), and the address matches the PCH's Slave Address, the PCH will still grab the cycle.

Also according to SMBus protocol, a Read cycle contains a Repeated Start—Address—Read sequence beginning at Bit 20. Once again, if the Address matches the PCH's Receive Slave Address, it will assume that the protocol is followed, ignore bit 28, and proceed with the Slave Read cycle.

Note: An external microcontroller must not attempt to access the PCH's SMBus Slave logic until at least one second after both RTCRST# and RSMRST# are deasserted (high).

Note: Until at least one second after both RTCRST# and RSMRST# are deasserted (high).

26.7.2.3 Slave Read of RTC Time Bytes

The PCH SMBus slave interface allows external SMBus master to read the internal RTC's time byte registers.

The RTC time bytes are internally latched by the PCH's hardware whenever RTC time is not changing and SMBus is idle. This ensures that the time byte delivered to the slave read is always valid and it does not change when the read is still in progress on the bus. The RTC time will change whenever hardware update is in progress, or there is a software write to the RTC time bytes.

The PCH SMBus slave interface only supports Byte Read operation. The external SMBus master will read the RTC time bytes one after another. It is software's responsibility to check and manage the possible time rollover when subsequent time bytes are read.

For example, assuming the RTC time is 11 hours: 59 minutes: 59 seconds. When the external SMBus master reads the hour as 11, then proceeds to read the minute, it is possible that the rollover happens between the reads and the minute is read as 0. This results in 11 hours: 0 minutes instead of the correct time of 12 hours: 0 minutes. Unless it is certain that rollover will not occur, software is required to detect the possible time rollover by reading multiple times such that the read time bytes can be adjusted accordingly if needed.

26.7.2.4 Format of Host Notify Command

The PCH tracks and responds to the standard Host Notify command as specified in the *System Management Bus (SMBus) Specification,* Version 2.0. The host address for this command is fixed to 0001000b. If the PCH already has data for a previously-received host notify command which has not been serviced yet by the host software (as indicated by the HOST_NOTIFY_STS bit), then it will NACK following the host address byte of the protocol. This allows the host to communicate non-acceptance to the master and retain the host notify address and data values for the previous cycle until host software completely services the interrupt.

Note: Host software must always clear the HOST_NOTIFY_STS bit after completing any necessary reads of the address and data registers.

Table 26-9 shows the Host Notify format.

Table 26-9. Host Notify Format

Bit	Description	Driven By	Comment
1	Start	External Master	
8:2	SMB Host Address – seven bits	External Master	Always 0001_000
9	Write	External Master	Always 0
10	ACK (or NACK)	PCH	PCH NACKs if HOST_NOTIFY_STS is 1
17:11	Device Address – seven bits	External Master	Indicates the address of the master; loaded into the Notify Device Address Register
18	Unused – Always 0	External Master	7-bit-only address; this bit is inserted to complete the byte
19	ACK	PCH	
27:20	Data Byte Low – eight bits	External Master	Loaded into the Notify Data Low Byte Register
28	ACK	PCH	
36:29	Data Byte High – eight bits	External Master	Loaded into the Notify Data High Byte Register
37	ACK	PCH	
38	Stop	External Master	

26.7.2.5 Format of Read Command

The external master performs Byte Read commands to the PCH SMBus Slave interface. The "Command" field (bits 18:11) indicate which register is being accessed. The Data field (bits 30:37) contains the value that should be read from that register.

Table 26-10. Slave Read Cycle Format

Bit	Description	Driven By	Comment
1	Start	External Microcontroller	
2–8	Slave Address - seven bits	External Microcontroller	Must match value in Receive Slave Address register
9	Write	External Microcontroller	Always 0
10	ACK	PCH	
11–18	Command code – eight bits	External Microcontroller	Indicates which register is being accessed. See Table 26-11 for a list of implemented registers.
19	ACK	PCH	
20	Repeated Start	External Microcontroller	
21–27	Slave Address - seven bits	External Microcontroller	Must match value in Receive Slave Address register
28	Read	External Microcontroller	Always 1
29	ACK	PCH	
30–37	Data Byte	PCH	Value depends on register being accessed. See Table 26-11 for a list of implemented registers.
38	NOT ACK	External Microcontroller	
39	Stop	External Microcontroller	

Table 26-11. Data Values for Slave Read Registers (Sheet 1 of 2)

Register	Bits	Description
0	7:0	Reserved for capabilities indication. Should always return 00h. Future chips may return another value to indicate different capabilities.
1	2:0	**System Power State** 000 = S0 001 = S1 010 = Reserved 011 = S3 100 = S4 101 = S5 110 = Reserved 111 = Reserved
	7:3	Reserved
2	3:0	Reserved
	7:4	Reserved
3	5:0	**Watchdog Timer current value** ***Note:*** The Watchdog Timer has 10 bits, but this field is only 6 bits. If the current value is greater than 3Fh, the PCH will always report 3Fh in this field.
	7:6	Reserved
4	0	**Intruder Detect.** 1 = The Intruder Detect (INTRD_DET) bit is set. This indicates that the system cover has probably been opened.
	1	**Temperature Event.** 1 = BTI Temperature Event occurred. This bit will be set if the PCH's THRM# input signal is active. Else this bit will read "0."
	2	**DOA Processor Status**. This bit will be 1 to indicate that the processor is dead
	3	1 = **SECOND_TO_STS** bit set. This bit will be set after the second Timeout (SECOND_TO_STS bit) of the Watchdog Timer occurs.
	6:4	Reserved. Will always be 0, but software should ignore.
	7	**SMBALERT# Status.** Reflects the value of the GPIO11/SMBALERT# pin (when the pin is configured as SMBALERT#). Valid only if SMBALERT_DISABLE = 0. Value always return 1 if SMBALERT_DISABLE = 1. (high = 1, low = 0).
5	0	**FWH bad bit.** This bit will be 1 to indicate that the FWH read returned FFh, which indicates that it is probably blank.
	1	**Battery Low Status**. Reserved.
	2	**SYS_PWROK Failure Status:** This bit will be 1 if the SYSPWR_FLR bit in the GEN_PMCON_2 register is set.
	3	Reserved
	4	Reserved
	5	**POWER_OK_BAD.** Indicates the failure core power well ramp during boot/resume. This bit will be active if the SLP_S3# pin is deasserted and PCH_PWROK pin is not asserted.
	6	**Thermal Trip**. This bit will shadow the state of processor Thermal Trip status bit (CTS). Events on signal will not create a event message
	7	Reserved: Default value is "X" ***Note:*** Software should not expect a consistent value when this bit is read through SMBUS/SMLink
6	7:0	Contents of the Message 1 register.
7	7:0	Contents of the Message 2 register.
8	7:0	Contents of the WDSTATUS register.
9	7:0	Seconds of the RTC
A	7:0	Minutes of the RTC
B	7:0	Hours of the RTC
C	7:0	"Day of Week" of the RTC
D	7:0	"Day of Month" of the RTC
E	7:0	Month of the RTC
F	7:0	Year of the RTC

Table 26-11. Data Values for Slave Read Registers (Sheet 2 of 2)

Register	Bits	Description
10h–FFh	7:0	Reserved

Table 26-12. Enables for SMBus Slave Write and SMBus Host Events

Event	INTREN (Host Control I/O Register, Offset 02h, Bit 0)	SMB_SMI_EN (Host Configuration Register, D31:F3:Offset 40h, Bit 1)	Event
Slave Write to Wake/SMI# Command	X	X	Wake generated when asleep. Slave SMI# generated when awake (SMBUS_SMI_STS).
Slave Write to SMLINK_SLAVE_SMI Command	X	X	Slave SMI# generated when in the S0 state (SMBUS_SMI_STS)
Any combination of Host Status Register [4:1] asserted	0	X	None
	1	0	Interrupt generated
	1	1	Host SMI# generated

26.8 SMBus Configuration Registers Summary

Table 26-13. Summary of SMBus Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	1h	Vendor ID (VID)—Offset 0h	8086h
2h	3h	Device ID (DID)—Offset 2h	A223
4h	5h	Command (CMD)—Offset 4h	0h
6h	7h	Device Status (DS)—Offset 6h	280h
8h	8h	Revision ID (RID)—Offset 8h	0h
9h	9h	Programming Interface (PI)—Offset 9h	0h
Ah	Ah	Sub Class Code (SCC)—Offset Ah	5h
Bh	Bh	Base Class Code (BCC)—Offset Bh	Ch
10h	13h	SMBus Memory Base Address_31_0 (SMBMBAR_31_0)—Offset 10h	4h
14h	17h	SMBus Memory Base Address_63_32 (SMBMBAR_63_32)—Offset 14h	0h
20h	23h	SMB Base Address (SBA)—Offset 20h	1h
2Ch	2Dh	SVID (SVID)—Offset 2Ch	0h
2Eh	2Fh	SID (SID)—Offset 2Eh	0h
3Ch	3Ch	Interrupt Line (INTLN)—Offset 3Ch	0h
3Dh	3Dh	Interrupt Pin (INTPN)—Offset 3Dh	1h
40h	40h	Host Configuration (HCFG)—Offset 40h	0h
50h	53h	TCO Base Address (TCOBASE)—Offset 50h	1h
54h	57h	TCO Control (TCOCTL)—Offset 54h	0h

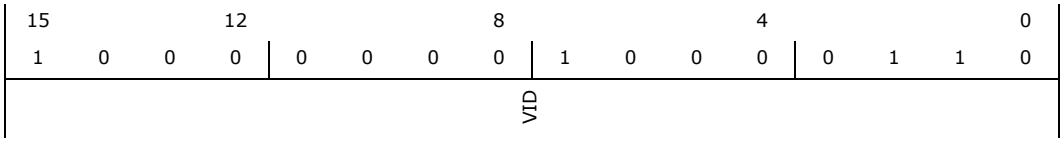
26.8.1 Vendor ID (VID)—Offset 0h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 4

Default: 8086h



Bit Range	Default and Access	Field Name (ID): Description
15:0	8086h RO	**Vendor ID (VID):** Value indicates Intel as the vendor

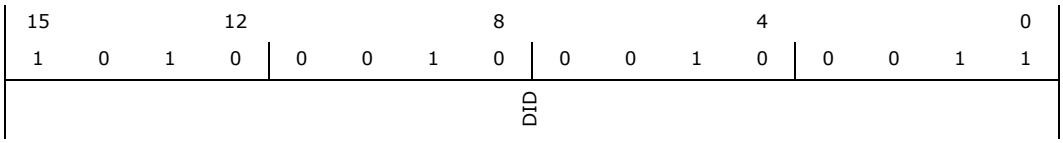
26.8.2 Device ID (DID)—Offset 2h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 4

Default: A223h



Bit Range	Default and Access	Field Name (ID): Description
15:0	A223h RO	**Device ID (DID):** Indicates the value assigned to the PCH SMBus controller.

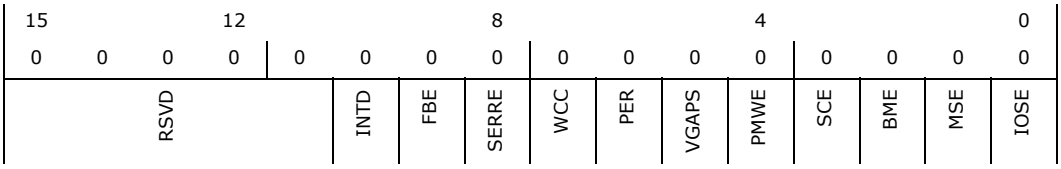
26.8.3 Command (CMD)—Offset 4h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:11	0h RO	Reserved.
10	0h RW	**Interrupt Disable (INTD):** 1 = Disables SMBus to assert its PIRQB# signal. Defaults to 0.
9	0h RW	**Fast Back to Back Enable (FBE):** Reserved as 0. Read Only.
8	0h RW	**SERR# Enable (SERRE):** 1 = Enables SERR# generation
7	0h RW	**Wait Cycle Control (WCC):** Reserved as 0. Read Only.
6	0h RW	**Parity Error Response (PER):** 1 = Sets Detected Parity Error bit when parity error is detected
5	0h RW	**VGA Palette Snoop (VGAPS):** Reserved as 0. Read Only.
4	0h RW	**Postable Memory Write Enable (PMWE):** Reserved as 0. Read Only.
3	0h RW	**Special Cycle Enable (SCE):** Reserved as 0. Read Only.
2	0h RW	**Bus Master Enable (BME):** Reserved as 0. Read Only.
1	0h RW	**Memory Space Enable (MSE):** 1= Enables memory mapped config space.
0	0h RW	**I/O Space Enable (IOSE):** 1= enables access to the SM Bus I/O space registers as defined by the Base Address Register.

26.8.4 Device Status (DS)—Offset 6h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 4

Default: 280h

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
DPE	SSE	RMA	RTA	STA	DEVT		DPED	FBC	UDF	C_66M	CLI	INTS	RSVD		

Bit Range	Default and Access	Field Name (ID): Description
15	0h RW/1C	**Detected Parity Error (DPE):** 1 = Parity error detected
14	0h RW/1C	**Signaled System Error (SSE):** 1 = System error detected
13	0h RO	**Received Master Abort (RMA):** Reserved as 0.
12	0h RO	**Received Target Abort (RTA):** Reserved as '0'.
11	0h RO	**Signaled Target-Abort Status (STA):** Reserved as 0.
10:9	1h RO	**DEVSEL# Timing Status (DEVT):** This 2-bit field defines the timing for DEVSEL# assertion. These read only bits indicate the Intel PCH's DEVSEL# timing when performing a positive decode. ***Note:*** Intel PCH generates DEVSEL# with medium time.
8	0h RO	**Data Parity Error Detected (DPED):** Reserved as 0.
7	1h RO	**Fast Back-to-Back Capable (FBC):** Reserved as '1'.
6	0h RO	**User Definable Features (UDF):** Reserved as 0.
5	0h RO	**66 MHz Capable (C_66M):** Reserved as 0.
4	0h RO	**Capabilities List Indicator (CLI):** Hardwired to 0 because there are no capability list structures in this function.
3	0h RO	**Interrupt Status (INTS):** This bit indicates that an interrupt is pending. It is independent from the state of the Interrupt Enable bit in the command register.
2:0	0h RO	Reserved.

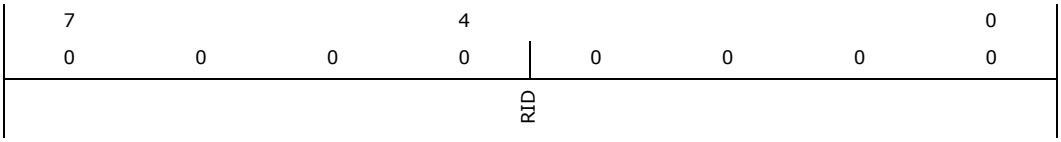
26.8.5 Revision ID (RID)—Offset 8h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO	**Revision ID (RID):** The value reported in this register depends on the global revision ID for the PCH.

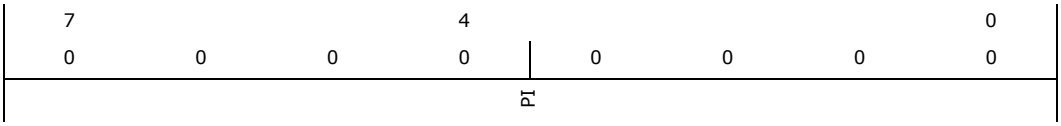
26.8.6 Programming Interface (PI)—Offset 9h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO	**Programming Interface (PI):** No programming interface defined.

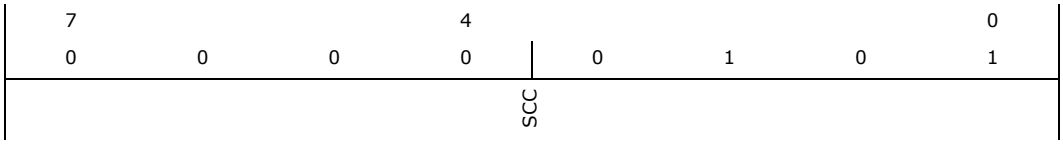
26.8.7 Sub Class Code (SCC)—Offset Ah

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 5h



Bit Range	Default and Access	Field Name (ID): Description
7:0	5h RO	**Sub Class Code (SCC):** A value of 05h indicates that this device is a SM Bus serial controller.

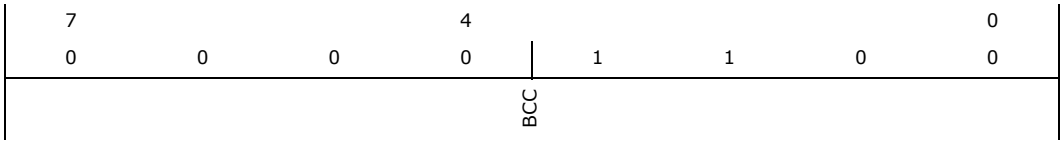
26.8.8 Base Class Code (BCC)—Offset Bh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 4

Default: Ch



Bit Range	Default and Access	Field Name (ID): Description
7:0	Ch RO	**Base Class Code (BCC):** A value of 0Ch indicates that this device is a serial controller

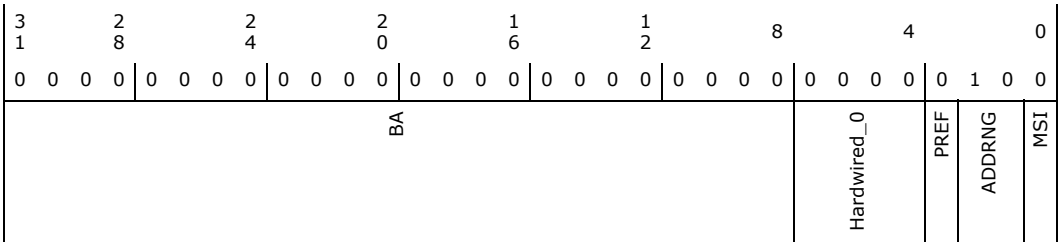
26.8.9 SMBus Memory Base Address_31_0 (SMBMBAR_31_0)—Offset 10h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 4

Default: 4h



Bit Range	Default and Access	Field Name (ID): Description
31:8	0h RW	**Base Address (BA):** Provides the 32 byte system memory base address for the Intel PCH SMB logic.
7:4	0h RO	**Hardwired_0 (Hardwired_0):** Hardwired to 0.
3	0h RO	**Prefetchable (PREF):** Hardwired to 0. Indicated that SMBMBAR is not pre-fetchable
2:1	2h RO	**Address Range (ADDRNG):** Indicates that this SMBMBAR can be located anywhere in 64 bit address space
0	0h RO	**Memory Space Indicator (MSI):** Indicates that the SMB logic is memory mapped.

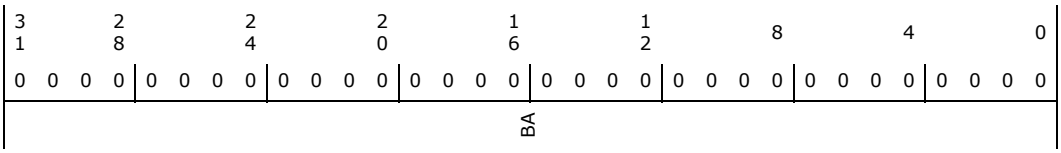
26.8.10 SMBus Memory Base Address_63_32 (SMBMBAR_63_32)—Offset 14h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Base Address (BA):** Bits 63-32 of SMbus Memory Base Address

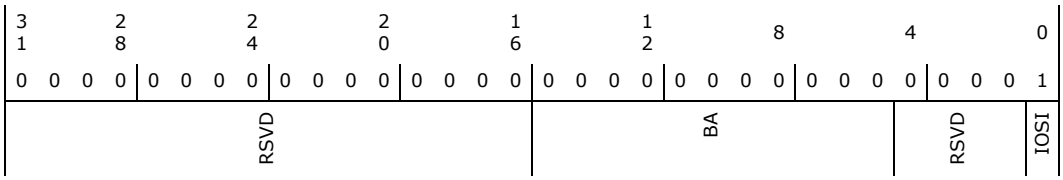
26.8.11 SMB Base Address (SBA)—Offset 20h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 4

Default: 1h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:5	0h RW	**Base Address (BA):** Provides the 32 byte t system I/O base address for the SMB logic.
4:1	0h RO	Reserved.
0	1h RO	**IO Space Indicator (IOSI):** This read-only bit always is 1, indicating that the SMB logic is I/O mapped.

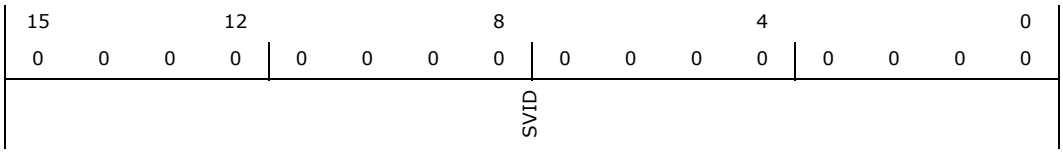
26.8.12 SVID (SVID)—Offset 2Ch

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:0	0h RW/O	**SVID (SVID):** BIOS sets the value in this register to identify the Subsystem Vendor ID. The SMBus SVID register, in combination with the SMBus Subsystem ID register, enables the operating system to distinguish each subsystem from the others. ***Note:*** Software can write to this register only once per core well reset. Writes should be done as a single 16-bit cycle.

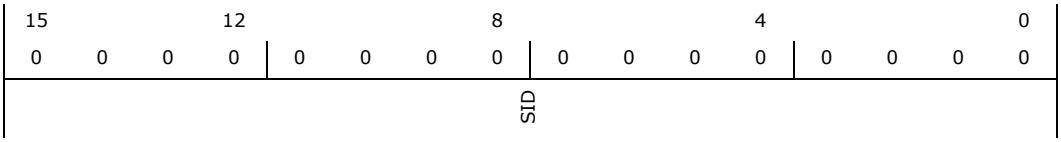
26.8.13 SID (SID)—Offset 2Eh

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:0	0h RW/O	**SID (SID):** BIOS can write to this register to identify the Subsystem ID. The SID register, in combination with the SVID, enable the operating system to distinguish each subsystem from other(s). ***Note:*** The software can write to this register only once per core well reset. Writes should be done as a single 16-bit cycle.

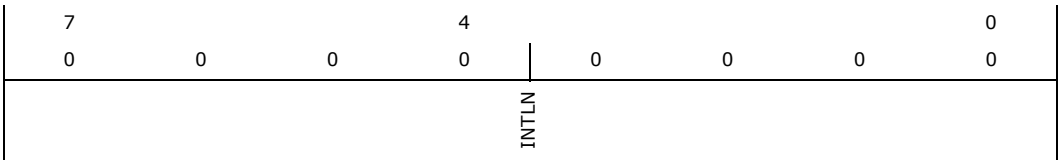
26.8.14 Interrupt Line (INTLN)—Offset 3Ch

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW	**Interrupt Line (INTLN):** This data is not used by the hardware. It is to communicate to software the interrupt line that the interrupt pin is connected to PIRQB#.

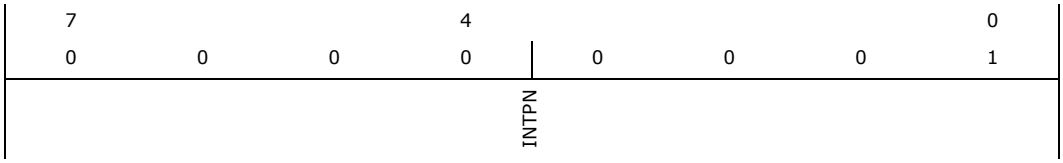
26.8.15 Interrupt Pin (INTPN)—Offset 3Dh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 1h



Bit Range	Default and Access	Field Name (ID): Description
7:0	1h RW/O	**Interrupt Pin (INTPN):** This defines the interrupt pin to be used by the SMBus controller. Bits : Pins 0h : No Interrupt 1h : INTA# 2h : INTB# 3h : INTC# 4h : INTD# 5h-Fh : Reserved

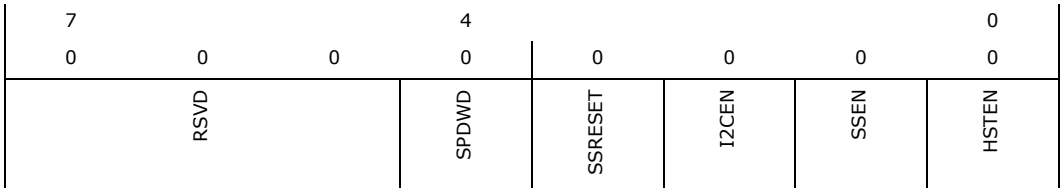
26.8.16 Host Configuration (HCFG)—Offset 40h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:5	0h RO	Reserved.
4	0h RW/O	**SPD Write Disable (SPDWD):** When this bit is set to 1, writes to SMBus addresses A0h – AEh are disabled. ***Note:*** This bit is R/WO and will be reset on PLTRST# assertion. This bit should be set by BIOS to '1'. Software can only program this bit when both the START bit and Host Busy bit are '0'; otherwise, the write may result in undefined behavior.
3	0h RW	**SSRESET (SSRESET):** Soft SMBUS Reset: When this bit is 1, the SMbus state machine and logic in PCH is reset. The HW will reset this bit to 0 when reset operation is completed.
2	0h RW	**I2C_EN (I2CEN):** When this bit is 1, the Intel PCH is enabled to communicate with I2C devices. This will change the formatting of some commands. When this bit is 0, behavior is for SMBus.

Bit Range	Default and Access	Field Name (ID): Description
1	0h RW	**SMB_SMI_EN (SSEN):** When this bit is set, any source of an SMB interrupt will instead be routed to generate an SMI#.
0	0h RW	**HST_EN (HSTEN):** When set, the SMB Host Controller interface is enabled to execute commands. The HST_INT_EN bit needs to be enabled in order for the SMB Host Controller to interrupt or SMI#. Additionally, the SMB Host Controller will not respond to any new requests until all interrupt requests have been cleared.

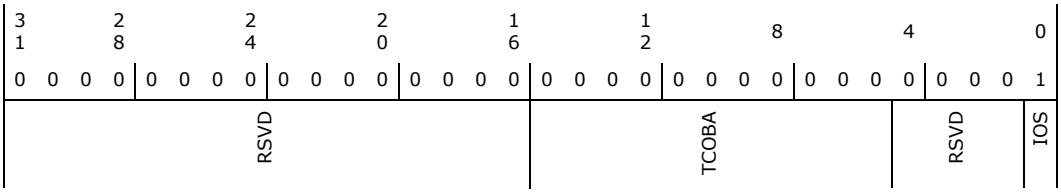
26.8.17 TCO Base Address (TCOBASE)—Offset 50h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 4

Default: 1h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:5	0h RW	**TCO Base Address (TCOBA):** Provides the 32 bytes of I/O space for TCO logic, mappable anywhere in the 64k I/O space on 32-byte boundaries.
4:1	0h RO	Reserved.
0	1h RO	**I/O Space (IOS):** Indicates an I/O Space

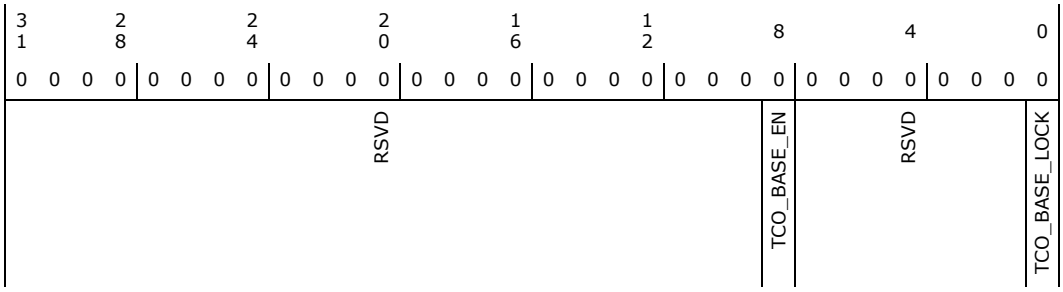
26.8.18 TCO Control (TCOCTL)—Offset 54h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	Reserved.
8	0h RW	**TCO Base Enable (TCO_BASE_EN):** When set, decode of the I/O range pointed to by the TCO base register is enabled.
7:1	0h RO	Reserved.
0	0h RW/O	**TCO Base Lock (TCO_BASE_LOCK):** If set to 1, the TCO Base Address register (TCOBASE) at offset 50 is locked. If set to 0, then the base register is writable. ***Note:*** This bit is Write Once. If you write a 0 to it, it will remain a 0 until a reset occurs. If you write a 1 to it, it will remain a 1 until a reset occurs. So the first write to this register locks this bit.

26.9 SMBus I/O and Memory Mapped I/O Registers Summary

The SMBus registers can be accessed through I/O BAR or Memory BAR registers in PCI configuration space. The offsets are the same for both I/O and Memory Mapped I/O registers.

Table 26-14. Summary of SMBus I/O and Memory Mapped I/O Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	0h	Host Status Register Address (HSTS)—Offset 0h	0h
2h	2h	Host Control Register (HCTL)—Offset 2h	0h
3h	3h	Host Command Register (HCMD)—Offset 3h	0h
4h	4h	Transmit Slave Address Register (TSA)—Offset 4h	0h
5h	5h	Data 0 Register (HD0)—Offset 5h	0h
6h	6h	Data 1 Register (HD1)—Offset 6h	0h
7h	7h	Host Block Data (HBD)—Offset 7h	0h

Table 26-14. Summary of SMBus I/O and Memory Mapped I/O Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
8h	8h	Packet Error Check Data Register (PEC)—Offset 8h	0h
9h	9h	Receive Slave Address Register (RSA)—Offset 9h	44h
Ah	Bh	Slave Data Register (SD)—Offset Ah	0h
Ch	Ch	Auxiliary Status (AUXS)—Offset Ch	0h
Dh	Dh	Auxiliary Control (AUXC)—Offset Dh	0h
Eh	Eh	SMLINK_PIN_CTL Register (SMLC)—Offset Eh	4h
Fh	Fh	SMBUS_PIN_CTL Register (SMBC)—Offset Fh	4h
10h	10h	Slave Status Register (SSTS)—Offset 10h	0h
11h	11h	Slave Command Register (SCMD)—Offset 11h	0h
14h	14h	Notify Device Address Register (NDA)—Offset 14h	0h
16h	16h	Notify Data Low Byte Register (NDLB)—Offset 16h	0h
17h	17h	Notify Data High Byte Register (NDHB)—Offset 17h	0h

26.9.1 Host Status Register Address (HSTS)—Offset 0h

All status bits are set by hardware and cleared by the software writing a one to the particular bit position. Writing a zero to any bit position has no affect.

Access Method

Type: IO Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
BDS	IUS	SMSTS	FAIL	BERR	DERR	INTR	HBSY

Bit Range	Default and Access	Field Name (ID): Description
7	0h RW/1C	**BYTE_DONE_STS (BDS):** This bit will be set to 1 when the host controller has received a byte (for Block Read commands) or if it has completed transmission of a byte (for Block Write commands) when the 32-byte buffer is not being used. Note that this bit will be set, even on the last byte of the transfer. Software clears the bit by writing a 1 to the bit position. This bit has no meaning for block transfers when the 32- byte buffer is enabled. ***Note:*** When the last byte of a block message is received, the host controller will set this bit. However, it will not immediately set the INTR bit (bit 1 in this register). When the interrupt handler clears the BYTE_DONE_STS bit, the message is considered complete, and the host controller will then set the INTR bit (and generate another interrupt). Thus, for a block message of n bytes, the Intel PCH will generate n+1 interrupts. The interrupt handler needs to be implemented to handle these cases.
6	0h RW	**In Use Status (IUS):** After a full PCI reset, a read to this bit returns a 0. After the first read, subsequent reads will return a 1. A write of a 1 to this bit will reset the next read value to 0. Writing a 0 to this bit has no effect. Software can poll this bit until it reads a 0, and will then own the usage of the host controller. This bit has no other effect on the hardware, and is only used as semaphore among various independent software threads that may need to use the Intel PCHs SMBus logic.
5	0h RW/1C	**SMBALERT_STS (SMSTS):** Intel PCH sets this bit to a 1 to indicates source of the interrupt or SMI# was the SMBAlert# signal. Software resets this bit by writing a 1 to this location. This bit should also be cleared by RSMRST# (but not PLTRST#).
4	0h RW/1C	**Failed (FAIL):** When set, this indicates that the source of the interrupt or SMI# was a failed bus transaction. This is set in response to the KILL bit being set to terminate the host transaction.
3	0h RW/1C	**Bus Error (BERR):** When set, this indicates the source of the interrupt or SMI# was a transaction collision.
2	0h RW/1C	**Device Error (DERR):** When set, this indicates that the source of the interrupt or SMI# was due one of the following: Illegal Command Field Unclaimed Cycle (host initiated) Host Device Time-out Error. CRC Error
1	0h RW/1C	**Interrupt (INTR):** When set, this indicates that the source of the interrupt or SMI# was the successful completion of its last command.
0	0h RW/1C	**Host Busy (HBSY):** A 1 indicates that the Intel PCH is running a command from the host interface. No SMB registers should be accessed while this bit is set. Exception: The BLOCK DATA REGISTER can be accessed when this bit is set ONLY when the SMB_CMD bits (in Host control register) are programmed for Block command or I2C Read command. This is necessary in order to check the DONE_STS bit.

26.9.2 Host Control Register (HCTL)—Offset 2h

Note: A read to this register will clear the pointer in the 32-byte buffer.

Access Method

Type: IO Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
PEC_EN	START	LAST_BYTE	SMB_CMD			KILL	INTREN

Bit Range	Default and Access	Field Name (ID): Description
7	0h RW	**PEC_EN (PEC_EN):** When set to 1, this bit causes the host controller to perform the SMBus transaction with the Packet Error Checking phase appended. For writes, the value of the PEC byte is transferred from the PEC Register. For reads, the PEC byte is loaded in to the PEC Register. When this bit is cleared to 0, the SMBus host controller does not perform the transaction with the PEC phase appended. This bit must be written prior to the write in which the START bit is set.
6	0h RW	**START (START):** This write-only bit is used to initiate the command described in the SMB_CMD field. All registers should be setup prior to writing a 1 to this bit position. This bit always reads zero. The HOST_BUSY bit in the Host Status register (offset 00h) can be used to identify when the Intel PCH has finished the command.
5	0h RW	**LAST_BYTE (LAST_BYTE):** This bit is used for I2C Read commands. Software sets this bit to indicate that the next byte will be the last byte to be received for the block. This causes the PCH to send a NACK (instead of an ACK) after receiving the last byte. ***Note:*** This bit may be set when the TCO timer causes the SECOND_TO_STS bit to be set. SW should clear the LAST_BYTE bit (if it is set) before starting any new command. In addition to I2C Read Commands, the LAST_BYTE bit will also cause Block Read/Write cycles to stop prematurely (at the end of the next byte).
4:2	0h RW	**SMB_CMD (SMB_CMD):** As shown by the bit encoding below, indicates which command the Intel PCH is to perform. If enabled, the Intel PCH will generate an interrupt or SMI# when the command has completed If the value is for a non-supported or reserved command, the Intel PCH will set the device error (DEV_ERR) status bit and generate an interrupt when the START bit is set. The Intel PCH will perform no command, and will not operate until DEV_ERR is cleared. Val - Command Description: 000 - Quick: The slave address and read/write value (bit 0) are stored in the tx slave address register. 001 - Byte: This command uses the transmit slave address and command registers. Bit 0 of the slave address register determines if this is a read or write command. 010 - Byte Data: This command uses the transmit slave address, command, and DATA0 registers. Bit 0 of the slave address register determines if this is a read or write command. If it is a read, the DATA0 register will contain the read data. 011 - Word Data: This command uses the transmit slave address, command, DATA0 and DATA1 registers. Bit 0 of the slave address register determines if this is a read or write command. If it is a read, after the command completes the DATA0 and DATA1 registers will contain the read data. 100 - Process Call: This command uses the transmit slave address, command, DATA0 and DATA1 registers. Bit 0 of the slave address register determines if this is a read or write command. After the command completes, the DATA0 and DATA1 registers will contain the read data. 101 - Block: This command uses the transmit slave address,command, and DATA0 registers, and the Block Data Byte register. For block write, the count is stored in the DATA0 register and indicates how many bytes of data will be transferred. For block reads, the count is received and stored in the DATA0 register. Bit 0 of the slave address register selects if this is a read or write command. For writes, data is retrieved from the first n (where n is equal to the specified count) addresses of the SRAM array. For reads, the data is stored in the Block Data Byte register. 110 - I2C Read: This command uses the transmit slave address, command, DATA0, DATA1 registers, and the Block Data Byte register. The read data is stored in the Block Data Byte register. The Intel PCH will continue reading data until the NAK is received. 111 - Block-Process: This command uses the transmit slave address, command, DATA0 and the Block Data Byte register. For block write, the count is stored in the DATA0 register and indicates how many bytes of data will be transferred. For block read, the count is received and stored in the DATA0 register. Bit 0 of the slave address register always indicate a write command. For writes, data is retrieved from the first m (where m is equal to the specified count) addresses of the SRAM array. For reads, the data is stored in the Block Data Byte register. ***Note:*** E32B bit in the Auxiliary Control Register must be set for this command to work.
1	0h RW	**KILL (KILL):** When set, kills the current host transaction taking place, sets the FAILED status bit, and asserts the interrupt (or SMI#) selected by the SMB_INTRSEL field. This bit, once set, must be cleared to allow the SMB Host Controller to function normally.
0	0h RW	**INTREN (INTREN):** Enable the generation of an interrupt or SMI# upon the completion of the command.

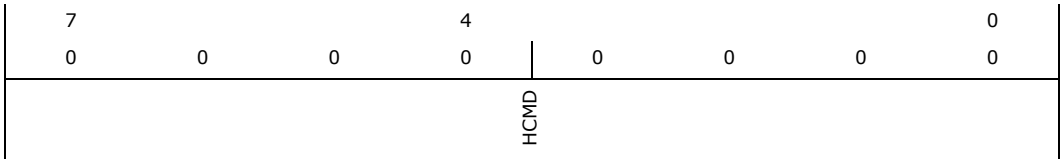
26.9.3 Host Command Register (HCMD)—Offset 3h

Access Method

Type: IO Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW	**Host Command Register (HCMD):** This eight bit field is transmitted by the host controller in the command field of the SMB protocol during the execution of any command.

26.9.4 Transmit Slave Address Register (TSA)—Offset 4h

This register is transmitted by the host controller in the slave address field of the SMB protocol. This is the address of the target.

Access Method

Type: IO Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 0h

Bit Range	Default and Access	Field Name (ID): Description
7:1	0h RW	**ADDRESS (ADDR):** 7-bit address of the targeted slave. ***Note:*** Writes to TSA values of A0h - AEh are blocked depending on the setting of the SPD write disable bit in HCFG - HostConfiguration.
0	0h RW	**RW (RW):** Direction of the host transfer. 1 = read, 0 = write

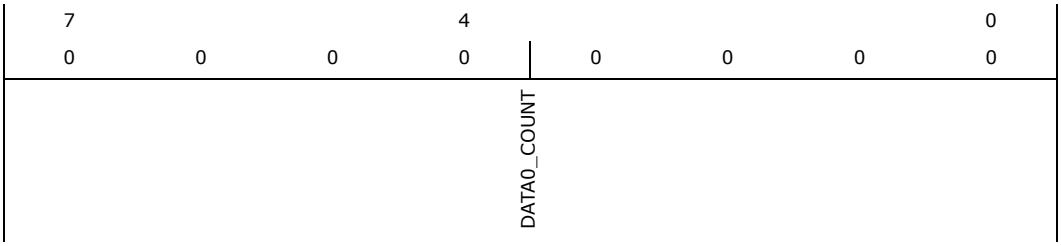
26.9.5 Data 0 Register (HD0)—Offset 5h

Access Method

Type: IO Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW	**DATA0/COUNT (DATA0_COUNT):** This field contains the eight bit data sent in the DATA0 field of the SMB protocol. For block write commands, this register reflects the number of bytes to transfer. This register should be programmed to a value between 1 and 32 for block counts. A count of 0 or a count above 32 will result in unpredictable behavior. The host controller does not check or log illegal block counts.

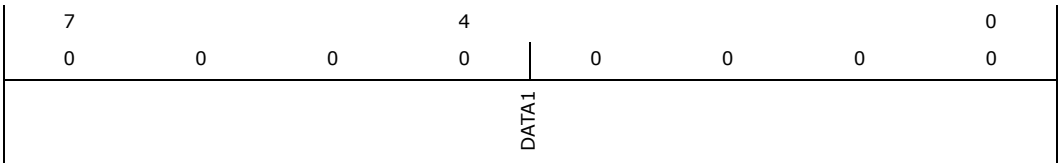
26.9.6 Data 1 Register (HD1)—Offset 6h

Access Method

Type: IO Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW	**DATA1 (DATA1):** This eight bit register is transmitted in the DATA1 field of the SMB protocol during the execution of any command.

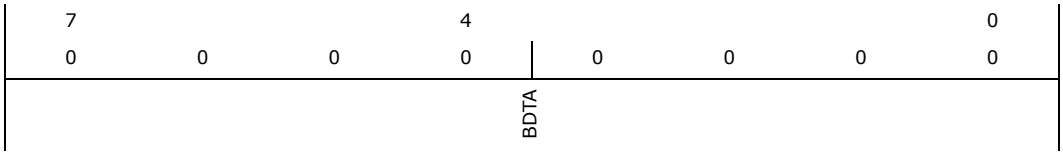
26.9.7 Host Block Data (HBD)—Offset 7h

Access Method

Type: IO Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW	**Block Data (BDTA):** This is either a register, or a pointer into a 32- byte block array, depending upon whether the E32B bit is set in the Auxiliary Control register. When the E32B bit is cleared, this is a register containing a byte of data to be sent on a block write or read from on a block read, just as it behaved on the INTEL PCH. When the E32B bit is set, reads and writes to this register are used to access the 32-byte block data storage array. An internal index pointer is used to address the array, which is reset to 0 by reading the HCTL register (offset 02h). The index pointer then increments automatically upon each access to this register. The transfer of block data into (read) or out of (write) this storage array during an SMBus transaction always starts at index address 0. When the E2B bit is set, for writes, software will write up to 32-bytes to this register as part of the setup for the command. After the Host Controller has sent the Address, Command, and Byte Count fields, it will send the bytes in the SRAM pointed to by this register. When the E2B bit is cleared for writes, software will place a single byte in this register. After the host controller has sent the address, command, and byte count fields, it will send the byte in this register. If there is more data to send, software will write the next series of bytes to the SRAM pointed to by this register and clear the DONE_STS bit. The controller will then send the next byte. During the time between the last byte being transmitted to the next byte being transmitted, the controller will insert wait-states on the interface. When the E2B bit is set for reads, after receiving the byte count into the Data0 register, the first series of data bytes go into the SRAM pointed to by this register. If the byte count has been exhausted or the 32-byte SRAM has been filled, the controller will generate an SMI# or interrupt (depending on configuration) and set the DONE_STS bit. Software will then read the data. During the time between when the last byte is read from the SRAM to when the DONE_STS bit is cleared, the controller will insert wait states on the interface.

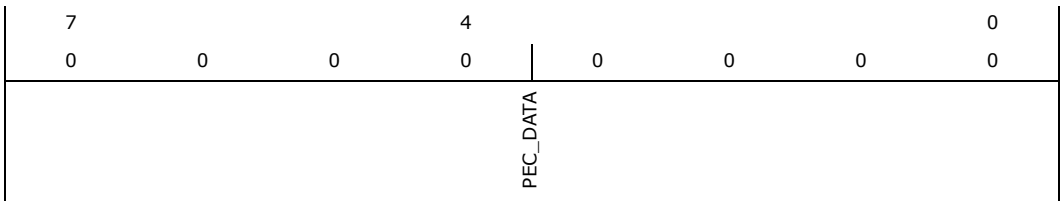
26.9.8 Packet Error Check Data Register (PEC)—Offset 8h

Access Method

Type: IO Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW	**PEC_DATA (PEC_DATA):** This 8-bit register is written with the SMBus PEC data prior to a write transaction. For read transactions, the PEC data is loaded from the SMBus into this register and is then read by software. Software must ensure that the INUSE_STS bit is properly maintained to avoid having this field over-written by a write transaction following a read transaction.

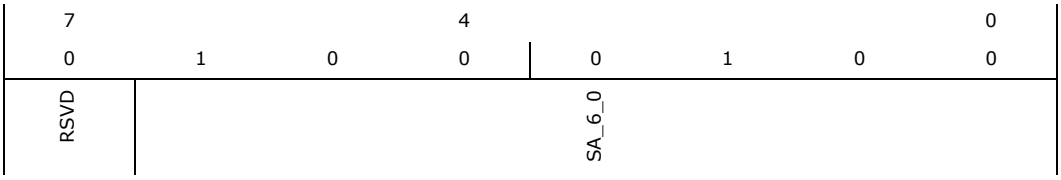
26.9.9 Receive Slave Address Register (RSA)—Offset 9h

Access Method

Type: IO Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 44h



Bit Range	Default and Access	Field Name (ID): Description
7	0h RO	Reserved.
6:0	44h RW	**SLAVE_ADDR[6:0] (SA_6_0):** This field is the slave address that the Intel PCH decodes for read and write cycles. The default is not 0 so that it can respond even before the CPU comes up (or if the CPU is dead). This register is reset by RSMRST#, but not by PLTRST#.

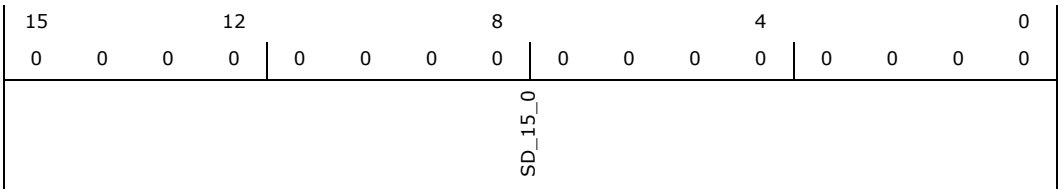
26.9.10 Slave Data Register (SD)—Offset Ah

Access Method

Type: IO Register
(Size: 16 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:0	0h RW	**SLAVE_DATA[15:0] (SD_15_0):** This field is the 16-bit data value written by the external SMBus master. The CPU can then read the value from this register. This register is reset by RSMRST#, but not by PLTRST#. SLAVE_DATA(7:0) corresponds to the Data Message Byte 0 at Slave Write Register 4 in the table. SLAVE_(15:8) corresponds to the Data Message Byte 1 at Slave Write Register 5 in the table.

26.9.11 Auxiliary Status (AUXS)—Offset Ch

All bits in this register are in the core well.

Access Method

Type: IO Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
RSVD			Reserved	Reserved	Reserved	RSVD	CRCE

Bit Range	Default and Access	Field Name (ID): Description
7:5	0h RO	Reserved.
4	0h RO	**Reserved**
3	0h RO	**Reserved**
2	0h RO	**Reserved**
1	0h RO	Reserved.
0	0h RW/1C	**CRC Error (CRCE):** This bit is set if a received message contained a CRC error. When this bit is set, the DERR bit of the host status register will also be set. This bit will be set by the controller if a software abort occurs in the middle of the CRC portion of the cycle or an abort happens after Intel PCH has received the final data bit transmitted by external slave.

26.9.12 Auxiliary Control (AUXC)—Offset Dh

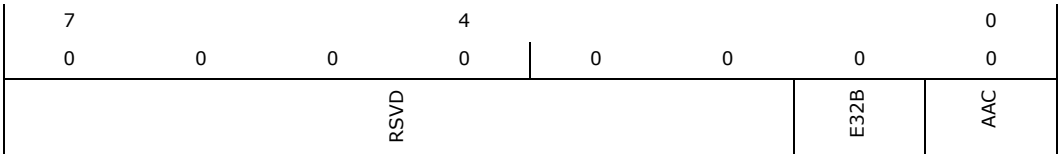
All bits in this register are in the resume well.

Access Method

Type: IO Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:2	0h RO	Reserved.
1	0h RW	**Enable 32-byte Buffer (E32B):** When set, the Host Block Data register is a pointer into a 32-byte buffer, as opposed to a single register. This enables the block commands to transfer or receive up to 32-bytes before the Intel PCH generates an interrupt.
0	0h RW	**Automatically Append CRC (AAC):** When set, the Intel PCH will automatically append the CRC. This bit must not be changed during SM Bus transactions, or undetermined behavior will result. It should be programmed only once during the lifetime of the function.

26.9.13 SMLINK_PIN_CTL Register (SMLC)—Offset Eh

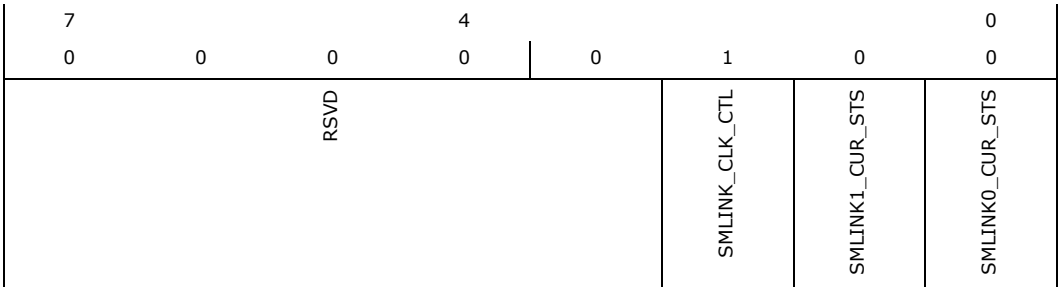
Note: This register is in the resume well and is reset by RSMRST#.

Access Method

Type: IO Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 4h



Bit Range	Default and Access	Field Name (ID): Description
7:3	0h RO	Reserved.
2	1h RW	**SMLINK_CLK_CTL (SMLINK_CLK_CTL):** 0 = Intel PCH will drive the SMLINK(0) pin low, independent of what the other SMLINK logic would otherwise indicate for the SMLINK(0) pin. 1 = The SMLINK(0) pin is Not overdriven low. The other SMLINK logic controls the state of the pin.
1	0h RO	**SMLINK[1]_CUR_STS (SMLINK1_CUR_STS):** This bit has a default value that is dependent on an external signal level. This returns the value on the SMLINK(1) pin. It will be 1 to indicate high, 0 to indicate low. This allows software to read the current state of the pin.
0	0h RO	**SMLINK[0]_CUR_STS (SMLINK0_CUR_STS):** This bit has a default value that is dependent on an external signal level. This returns the value on the SMLINK(0) pin. It will be 1 to indicate high, 0 to indicate low. This allows software to read the current state of the pin.

26.9.14 SMBUS_PIN_CTL Register (SMBC)—Offset Fh

Note: This register is in the resume well and is reset by RSMRST#

Access Method

Type: IO Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 4h

7	6	5	4	3	2	1	0
0	0	0	0	0	1	0	0
RSVD					SMBCLK_CTL	SMBDATA_CUR_STS	SMBCLK_CUR_STS

Bit Range	Default and Access	Field Name (ID): Description
7:3	0h RO	Reserved.
2	1h RW	**SMBCLK_CTL (SMBCLK_CTL):** 0 = Intel PCH will drive the SMBCLK pin low, independent of what the other SMB logic would otherwise indicate for the SMBCLK pin. 1 = The SMBCLK pin is Not overdriven low. The other SMBus logic controls the state of the pin.
1	0h RO	**SMBDATA_CUR_STS (SMBDATA_CUR_STS):** This bit has a default value that is dependent on an external signal level. This returns the value on the SMBDATA pin. It will be 1 to indicate high, 0 to indicate low. This allows software to read the current state of the pin.
0	0h RO	**SMBCLK_CUR_STS (SMBCLK_CUR_STS):** This bit has a default value that is dependent on an external signal level. This returns the value on the SMBCLK pin. It will be 1 to indicate high, 0 to indicate low. This allows software to read the current state of the pin.

26.9.15 Slave Status Register (SSTS)—Offset 10h

All bits in this register are implemented in the 64 kHz clock domain. Therefore, software must poll the register until a write takes effect before assuming that a write has completed internally.

Access Method

Type: IO Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
RSVD							HNS

Bit Range	Default and Access	Field Name (ID): Description
7:1	0h RO	Reserved.
0	0h RW/1C	**HOST_NOTIFY_STS (HNS):** The Intel PCH sets this bit to a 1 when it has completely received a successful Host Notify Command on the SMBus pins. Software reads this bit to determine that the source of the interrupt or SMI# was the reception of the Host Notify Command. Software clears this bit after reading any information needed from the Notify address and data registers by writing a 1 to this bit. Note that the Intel PCH will allow the Notify Address and Data registers to be over-written once this bit has been cleared. When this bit is 1, the Intel PCH will NACK the first byte (host address) of any new Host Notify commands on the SMBus. Writing a 0 to this bit has no effect.

26.9.16 Slave Command Register (SCMD)—Offset 11h

All bits in this register are implemented in the 64 kHz clock domain. Therefore, software must poll the register until a write takes effect before assuming that a write has completed internally. Also, software must confirm the prior written value before writing to the register again.

Access Method

Type: IO Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
RSVD					SMB_D	HNW	HNI

Bit Range	Default and Access	Field Name (ID): Description
7:3	0h RO	Reserved.
2	0h RW	**SMBALERT_DIS (SMB_D):** Software sets this bit to 1 to block the generation of the interrupt or SMI# due to the SMBALERT# source. This bit is logically inverted and ANDed with the SMBALERT_STS bit. The resulting signal is distributed to the SMI# and/or interrupt generation logic. This bit does not effect the wake logic.
1	0h RW	**HOST_NOTIFY_WKEN (HNW):** Software sets this bit to 1 to enable the reception of a Host Notify command as a wake event. When enabled this event is ORed in with the other SMBus wake events and is reflected in the SMB_WAK_STS bit of the General Purpose Event 0 Status register.
0	0h RW	**HOST_NOTIFY_INTREN (HNI):** Software sets this bit to 1 to enable the generation of interrupt or SMI# when HOST_NOTIFY_STS is 1. This enable does not affect the setting of the HOST_NOTIFY_STS bit. When the interrupt is generated, either PIRQB or SMI# is generated, depending on the value of the SMB_SMI_EN bit (D31, F3, Off40h, B1). If the HOST_NOTIFY_STS bit is set when this bit is written to a 1, then the interrupt (or SMI#) will be generated. The interrupt (or SMI#) is logically generated by ANDing the STS and INTREN bits.

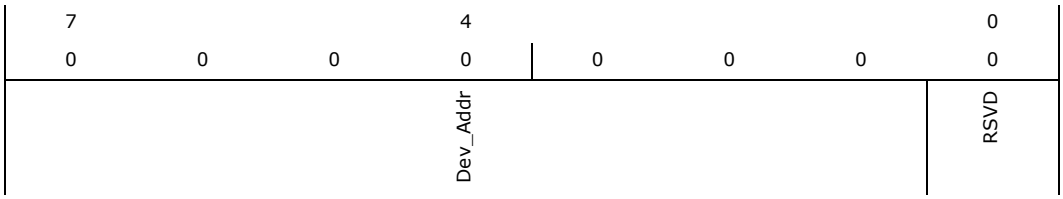
26.9.17 Notify Device Address Register (NDA)—Offset 14h

Access Method

Type: IO Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:1	0h RW	**DEVICE_ADDRESS (Dev_Addr):** This field contains the 7-bit device address received during the Host Notify protocol of the SMBus 2.0 specification. Software should only consider this field valid when the HOST_NOTIFY_STS bit is set to 1.
0	0h RO	Reserved.

26.9.18 Notify Data Low Byte Register (NDLB)—Offset 16h

Access Method

Type: IO Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
DLB							

Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO	**DATA_LOW_BYTE (DLB):** This field contains the first (low) byte of data received during the Host Notify protocol of the SMBus 2.0 specification. Software should only consider this field valid when the HOST_NOTIFY_STS bit is set to 1.

26.9.19 Notify Data High Byte Register (NDHB)—Offset 17h

Access Method

Type: IO Register
(Size: 8 bits)

Device: 31
Function: 4

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
DHB							

Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO	**DATA_HIGH_BYTE (DHB):** This field contains the second (high) byte of data received during the Host Notify protocol of the SMBus 2.0 specification. Software should only consider this field valid when the HOST_NOTIFY_STS bit is set to 1.

26.10 SMBus PCR Registers Summary

These registers are within the PCH Private Configuration Space which is accessible through the PCH Sideband Interface. They can be accessed via (SBREG_BAR + PortID + Register Offset). The offsets are DWORD aligned byte addresses.

Table 26-15. Summary of SMBus PCR Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	TCO Configuration (TCOCFG)—Offset 0h	0h
Ch	Fh	General Control (GC)—Offset Ch	0h

26.10.1 TCO Configuration (TCOCFG)—Offset 0h

Access Method

Type: MSG Register (Size: 32 bits)

Device: 31
Function: 4

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																							IE	RSVD					IS		

Bit Range	Default and Access	Field Name (ID): Description
31:8	0h RO	Reserved.
7	0h RW	**TCO IRQ Enable (IE):** When set, TCO IRQ is enabled, as selected by the TCO_IRQ_SEL field. When cleared, TCO IRQ is disabled.
6:3	0h RO	Reserved.
2:0	0h RW	**TCO IRQ Select (IS):** Specifies on which IRQ the TCO will internally appear. If not using the APIC, the TCO interrupt must be routed to IRQ9-11, and that interrupt is not sharable with the SERIRQ stream, but is shareable with other PCI interrupts. If using the APIC, the TCO interrupt can also be mapped to IRQ20-23, and can be shared with other interrupt. Bits TCO Map 000 IRQ9 (maps to 8259 and APIC) 001 IRQ10 maps to 8259 and APIC) 010 IRQ11 (maps to 8259 and APIC) 011 Reserved 100 IRQ20 (maps to APIC) 101 IRQ21 (maps to APIC) 110 IRQ22 (maps to APIC) 111 IRQ23 (maps to APIC) When setting the these bits, the IE bit should be cleared to prevent glitching. When the interrupt is mapped to APIC interrupts 9, 10 or 11, the APIC should be programmed for active-high reception. When the interrupt is mapped to APIC interrupts 20 through 23, the APIC should be programmed for active-low reception.

26.10.2 General Control (GC)—Offset Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device: 31
Function: 4

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																														NR	FD

Bit Range	Default and Access	Field Name (ID): Description
31:2	0h RO	Reserved.
1	0h RW	**No Reboot (NR):** This bit is set when the No Reboot strap is sampled high on PWROK. This bit may be set or cleared by software if the strap is sampled low but may not override the strap when it indicates No Reboot. When set, the TCO timer will count down and generate the SMI# on the first timeout, but will not reboot on the second timeout.
0	0h RW	**Function Disable (FD):** When set to one, this disables the PCI config register space for the SMBus device.

§

27 Serial Peripheral Interface

27.1 Acronyms

Acronyms	Description
MISO	Mater In Slave Out
MOSI	Mater Out Slave In
SPI	Serial Peripheral Interface
SFDP	Serial Flash Discoverable Parameter

27.2 References

None

27.3 Overview

The SPI interface, implementing three chip select signals, allows up to two flash devices and one TPM device to be connected to the PCH.

The SPI interfaces support either 1.8V or 3.3V.

27.4 Signal Description

Name	Type	Description
SPI0_CLK	O	**SPI Clock**: SPI clock signal for the common flash/TPM interface. Supports 17MHz, 30 MHz and 40 MHz.
SPI0_FLASH_ CS0_N	O	**SPI FLASH Chip Select 0**: Used to select the primary SPI Flash device. ***Note:*** This signal cannot be used for any other type of device than SPI Flash.
SPI0_FLASH_CS1_N	O	**SPI FLASH Chip Select 1**: Used to select an optional secondary SPI Flash device. ***Note:*** This signal cannot be used for any other type of device than SPI Flash.
SPI0_TPM_CS_N	O	**SPI TPM Chip Select:** Used to select the TPM device if it is connected to the SPI interface; it cannot be used for any other type of device. ***Note:*** TPM can be configured through soft straps to operate over LPC or SPI, but no more than 1 TPM is allowed in the system.
SPI0_MOSI_IO0	I/O	**SPI Master OUT Slave IN**: Defaults as a data output pin for PCH in Dual Output Fast Read mode. Can be configured with a Soft Strap as a bidirectional signal (SPI_IO0) to support the new Dual I/O Fast Read, Quad I/O Fast Read and Quad Output Fast Read modes.
SPI0_MISO_IO1	I/O	**SPI Master IN Slave OUT**: Defaults as a data input pin for PCH in Dual Output Fast Read mode. Can be configured with a Soft Strap as a bidirectional signal (SPI_IO1) to support the new Dual I/O Fast Read, Quad I/O Fast Read and Quad Output Fast Read modes.
SPI0_IO[3:2]	I/O	**SPI Data I/O:** Bidirectional signals used to support Quad I/O Fast Read and Quad Output Fast Read modes.

27.5 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
SPI0_CLK	Pull-up	15K - 40K	
SPI0_MOSI_IO0	Pull-up	15K - 40K	1
SPI0_MISO_IO1	Pull-up	15K - 40K	
SPI0_FLASH_CS0_N, SPI0_FLASH_CS1_N, SPI0_TPM_CS_N	Pull-up	15K - 40K	
SPI0_IO[2:3]	Pull-up	15K - 40K	

Notes:
1. Weak internal pull-up resistor is enabled when RSMRST# is asserted and is switched to a weak internal pull-down when RSMRST# is de-asserted.
2. Integrated pull-ups and pull-downs will not be valid until all the voltages have reached a valid level.

27.6 I/O Signal Planes and States

Signal Name	Power Plane	During Reset	Immediately after Reset	S3/S4/S5	Deep Sx
SPI0_CLK	Primary	Driven Low (See Note 1)	Driven Low	Driven Low	Off
SPI0_MOSI_IO0	Primary	Internal Pull up/ (See Note 1)	Driven Low	Driven Low	Off
SPI0_MISO_IO1	Primary	Internal Pull-up	Internal Pull-up	Internal Pull-up	Off
SPI0_FLASH_CS0_N	Primary	Driven High (See Note 1)	Driven High	Driven High	Off
SPIO_FLASH_CS1_N	Primary	Internal Pull-up (See Note 1)	Driven High	Driven High	Off
SPI0_TPM_CS_N	Primary	Driven High (See Note 1)	Driven High	Driven High	Off
SPI0_IO[2:3]	Primary	Internal Pull-up (See Note 1)	Internal Pull-up	Internal Pull-up	Off

Notes:
1. Pins are tri-stated (with weak internal pull-up) prior to RSMRST# de-assertion.
2. Integrated pull-ups and pull-downs will not be valid until all the voltages have reached a valid level.

27.7 Functional Description

27.7.1 SPI for Flash

27.7.1.1 Overview

The PCH supports up to two SPI flash devices using two separate Chip Select pins. The maximum size of flash supported is determined by the SFDP-discovered addressing capability of each device. Each component can be up to 16 MB using 3-byte addressing or 64 MB using 4-byte addressing.

The PCH SPI interface supports 17 MHz, 30 MHz, and 40 MHz SPI devices. Note that a flash device meeting 66 MHz timing is required for 40 MHz operation.

Flash devices used with the SPI controller must contain the SFDP table.

The SPI interface supports either 3.3V or 1.8V.

A SPI Flash device on Chip Select 0 (SPIO_Flash_CS0#) with a valid descriptor MUST be attached directly to the PCH.

The PCH supports fast read which consist of:

1. Dual Output Fast Read (Single Input Dual Output)
2. Dual I/O Fast Read (Dual Input Dual Output)
3. Quad Output Fast Read (Single Input Quad Output)
4. Quad I/O Fast Read (Quad Input Quad Output)

The PCH SPI has a third chip select SPI0_TPM_CS# for TPM support over SPI. TPM Bus will use SPI0_CLK, SPI0_Flash_MISO, SPI0_Flash_MOSI and SPI_Flash_CS1# SPI signals.

Notes:

1. If Boot BIOS Strap ='1', then LPC/eSPI is selected as the location for the BIOS. If the LPC bus is replaced with the eSPI bus, then the BIOS will be located off of eSPI.
2. When SPI is selected by the Boot BIOS Destination Strap and a SPI device is detected by the PCH, LPC based BIOS flash is disabled.
3. If LPC is disabled through soft straps (Soft Strap 120, Bit 13), the SPI TPM will not function as part of the SPI TPM control is done via the LPC logic. This has nothing to do with the selection of LPC/eSPI via hardstrap. It is easy to make the erroneous assumption that if eSPI is selected, then LPC should be disabled through SS 120.

27.7.1.2 SPI Supported Features

27.7.1.2.1 Descriptor Mode

Descriptor Mode is required for all SKUs of the PCH. Non-Descriptor Mode is not supported.

27.7.1.2.2 SPI Flash Regions

In Descriptor Mode the Flash is divided into multiple separate regions:

Table 27-1. SPI Flash Regions

Region	Content
0	Flash Descriptor
1	BIOS
2	Intel Management Engine
3	Gigabit Ethernet
4	Platform Data
5	Device Expansion
6	Secondary BIOS
7	RSVD
8	BMC
9	Device Expansion 2
10	Innovation Engine
11	10 GbE region #1
12	Option ROM
13-15	Reserved

Access to the SPI flash can be done by the following masters 1) TPM Device (via CS# to a TPM device on the SPI bus), 2) Host Processor, 3) Management Engine, 4) Innovation Engine, 5) Integrated 1 GbE MAC, 6) 10 GbE LAN controllers and 7) eSPI. An external BMC would get access to SPI via the eSPI interface.

The Flash Descriptor and Intel ME region are the only required regions. The Flash Descriptor has to be in region 0 and region 0 must be located in the first sector of Device 0 (Offset 0).

Regions can extend across multiple components, but must be contiguous.

Flash Region Sizes

SPI flash space requirements differ by platform and configuration. The Flash Descriptor requires one 4 KB or larger block. GbE requires two 4KB or larger blocks. The amount of flash space consumed is dependent on the erase granularity of the flash part and the platform requirements for the Intel ME and BIOS regions. The Intel ME region contains firmware to support Intel Active Management Technology and other Intel ME capabilities.

Table 27-2. Region Size Versus Erase Granularity of Flash Components

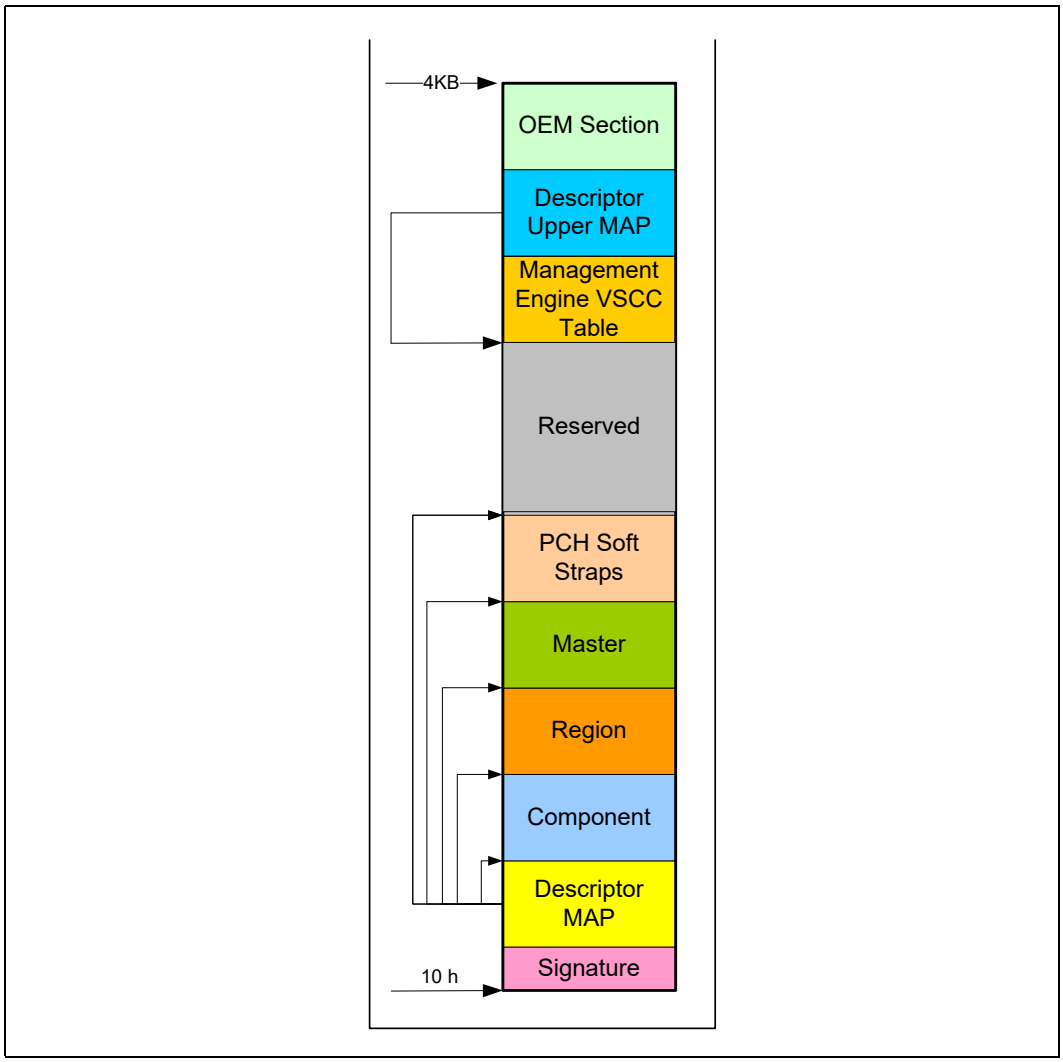
Region	Size with 4 KB Blocks	Size with 8 KB Blocks	Size with 64 KB Blocks
Descriptor	4 KB	8 KB	64 KB
Legacy GbE	8 KB	16 KB	128 KB
BIOS	Varies by Platform	Varies by Platform	Varies by Platform
Intel ME	Varies by Platform	Varies by Platform	Varies by Platform
BMC	Varies by Platform	Varies by Platform	Varies by Platform
10/1 GbE	Varies by Platform	Varies by Platform	Varies by Platform

27.7.1.3 Flash Descriptor

The bottom sector of the flash component 0 contains the Flash Descriptor. The maximum size of the Flash Descriptor is 4 KB. If the block/sector size of the SPI flash device is greater than 4 KB, the flash descriptor will only use the first 4 KB of the first block. The flash descriptor requires its own block at the bottom of memory (00h). The information stored in the Flash Descriptor can only be written during the manufacturing process as its read/write permissions must be set to read only when the computer leaves the manufacturing floor.

The Flash Descriptor is made up of eleven sections as shown in Figure 27-1.

Figure 27-1. Flash Descriptor Regions



- The Flash signature selects Descriptor Mode as well as verifies if the flash is programmed and functioning. The data at the bottom of the flash (offset 10h) must be 0FF0A55Ah in order to be in Descriptor mode.
- The Descriptor map has pointers to the other five descriptor sections as well as the size of each.
- The component section has information about the SPI flash in the system including: the number of components, density of each, invalid instructions (such as chip erase), and frequencies for read, fast read and write/erase instructions.
- The Region section points to the three other regions as well as the size of each region.
- The master region contains the security settings for the flash, granting read/write permissions for each region and identifying each master by a requestor ID.
- The processor and PCH Soft Strap sections contain processor and PCH configurable parameters.
- The Reserved region between the top of the processor strap section and the bottom of the OEM Section is reserved for future chipset usages.

- The Descriptor Upper MAP determines the length and base address of the Management Engine VSCC Table.
- The Management Engine VSCC Table holds the JEDEC ID and the VSCC information of the entire SPI Flash supported by the NVM image.
- OEM Section is 256 Bytes reserved at the top of the Flash Descriptor for use by OEM.

27.7.1.3.1 Descriptor Master Region

The master region defines read and write access setting for each region of the SPI device. The master region recognizes four masters: BIOS, Gigabit Ethernet, Management Engine, and EC. Each master is only allowed to do direct reads of its primary regions.

Note: The 10 GbE Region #2 is not used by the Intel® C620 Series Chipset.

Table 27-3. Region Access Control Table (Sheet 1 of 2)

Master Read/Write Access						
Region	**Processor and BIOS**	**Intel ME**	**1 GbE Controller**	**EVA**	**EC/BMC**	**Innovation Engine**
Descriptor	N/A	N/A	N/A	N/A	N/A	N/A
BIOS	Processor and BIOS can always read from and write to BIOS Region	Read/Write	Read/Write	Read/Write	Read/Write	Read/Write
Intel Management Engine (Intel ME)	Read/Write	Intel ME can always read from and write to Intel ME Region	Read/Write	Read/Write	Read/Write	Read/Write
1 Gigabit Ethernet	Read/Write	Read/Write	GbE software can always read from and write to GbE region	Read/Write	Read/Write	Read/Write
Platform Data Region	Read/Write	read/write	N/A	Read/write	Read/write	Read/write
Device Expansion Region	Read/Write	Read/Write	N/A	EVA can always read from and write to the DER	Read/write	Read/Write
Secondary BIOS	Processor and BIOS can always read from and write to BIOS Region	Read/Write	Read/Write	Read/Write	Read/write	Read/Write
RSVD	RO	N/A	N/A	N/A	N/A	N/A
BMC	Read/Write	Read/Write	Read/Write	Read/Write	EC can always read from and write to EC region	Read/Write
Device Expansion Region #2	Read/Write	Read/Write	Read/Write	EVA can always read from and write to the DER	Read/write	Read/Write
Innovation Engine	Read/Write	Read/Write		Read/Write	Read/write	I/E can always read from and write to the DER
10 GbE Region #1	Read/Write	Read/Write	Read/Write	Read/Write	Read/Write	

Table 27-3. Region Access Control Table (Sheet 2 of 2)

Master Read/Write Access						
Region	**Processor and BIOS**	**Intel ME**	**1 GbE Controller**	**EVA**	**EC/BMC**	**Innovation Engine**
Option ROM	Read/Write	Read/Write	Read/Write	Read/Write	Read/Write Read/Write	Read/Write

27.7.1.4 Flash Access

There are two types of accesses: Direct Access and Program Register Accesses.

27.7.1.4.1 Direct Access

- Masters are allowed to do direct read only of their primary region.
  - Gigabit Ethernet region can only be directly accessed by the Gigabit Ethernet controller. Gigabit Ethernet software must use Program Registers to access the Gigabit Ethernet region.
- Master's Host or Management Engine virtual read address is converted into the SPI Flash Linear Address (FLA) using the Flash Descriptor Region Base/Limit registers.

Direct Access Security

- Requester ID of the device must match that of the primary Requester ID in the Master Section
- Calculated Flash Linear Address must fall between primary region base/limit
- Direct Write not allowed
- Direct Read Cache contents are reset to 0's on a read from a different master.
  - Supports the same cache flush mechanism in ICH7 which includes Program Register Writes.

27.7.1.4.2 Program Register Access

- Program Register Accesses are not allowed to cross a 4-KB boundary and can not issue a command that might extend across two components.
- Software programs the FLA corresponding to the region desired
  - Software must read the devices Primary Region Base/Limit address to create a FLA.

Register Access Security

- Only primary region masters can access the registers

Note: Processor running Gigabit Ethernet software can access Gigabit Ethernet registers

- Masters are only allowed to read or write those regions they have read/write permission.
- Using the Flash Region Access Permissions, one master can give another master read/write permissions to their area.
- Using the five Protected Range registers, each master can add separate read/write protection above that granted in the Flash Descriptor for their own accesses.
  - Example: The BIOS may want to protect different regions of BIOS from being erased.
  - Ranges can extend across region boundaries.

27.7.2 SPI Support for TPM

The PCH's SPI flash controller supports a discrete TPM on the platform via its dedicated SPI0_TPM_CS# signal. The platform must have no more than 1 TPM.

SPI controller supports accesses to SPI TPM at approximately 17 MHz, 30 MHz or 40 MHz depending on the PCH soft strap. 17 MHz is the reset default; a valid PCH soft strap setting overrides the requirement for the 17 MHz.

TPM requires the support for the interrupt routing. However, the TPM's interrupt pin is routed to the PCH's PIRQ pin. Thus, TPM interrupt is completely independent from the SPI controller.

Note: The SPI controller is configurable to prevent TPM access when the descriptor is invalid (or no flash is attached).

27.8 SPI Configuration Registers Summary

Table 27-4. Summary of SPI Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	Device ID and Vendor ID (BIOS_SPI_DID_VID)—Offset 0h	9D248086h
4h	7h	Status and Command (BIOS_SPI_STS_CMD)—Offset 4h	400h
8h	Bh	Revision ID (BIOS_SPI_CC_RID)—Offset 8h	0h
Ch	Fh	BIST, Header Type, Latency Timer, Cache Line Size (BIOS_SPI_BIST_HTYPE_LT_CLS)—Offset Ch	0h
10h	13h	SPI BAR0 MMIO (BIOS_SPI_BAR0)—Offset 10h	0h
D0h	D3h	SPI Unsupported Request Status (BIOS_SPI_UR_STS_CTL)—Offset D0h	0h
D8h	DBh	BIOS Decode Enable (BIOS_SPI_BDE)—Offset D8h	FFCFh
DCh	DFh	BIOS Control (BIOS_SPI_BC)—Offset DCh	28h

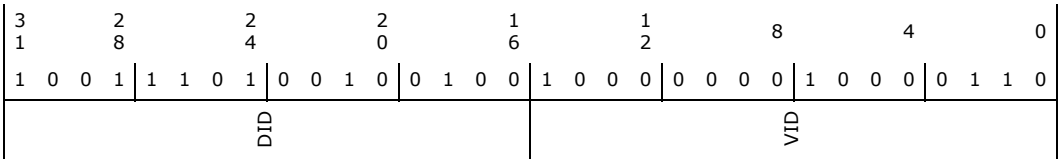
27.8.1 Device ID and Vendor ID (BIOS_SPI_DID_VID)—Offset 0h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 9D248086h



Bit Range	Default and Access	Field Name (ID): Description
31:16	9D24h RO/V	**Device Identification (DID):** Identifier for the SPI Flash Controller in Host Root Space.
15:0	8086h RO	**Vendor Identification (VID):** This field identifies the manufacturer of the device. 0x8086 indicates Intel

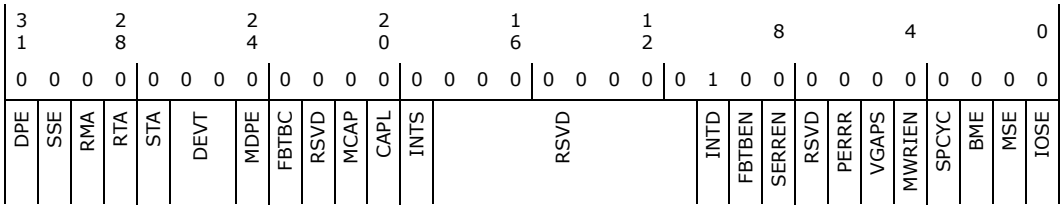
27.8.2 Status and Command (BIOS_SPI_STS_CMD)—Offset 4h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 400h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RW/1C/V	**Detected Parity Error (DPE):** Detected Parity Error (DPE)
30	0h RW/1C/V	**Signaled System Error (SSE):** Signaled System Error (SSE)
29	0h RO	**Received Master Abort (RMA):** Received Master Abort (RMA)
28	0h RO	**Received Target Abort (RTA):** Received Target Abort (RTA)
27	0h RW/1C/V	**Signaled Target Abort (STA):** Signaled Target Abort (STA)
26:25	0h RO	**Devsel Timing (DEVT):** Devsel Timing (DEVT)
24	0h RO	**Master Data Parity Error (MDPE):** Master Data Parity Error (MDPE)
23	0h RO	**Fast Back to Back Capable (FBTBC):** Fast Back to Back Capable (FBTBC)
22	0h RO	Reserved.
21	0h RO	**66 MHz Capable (MCAP):** Not 66MHz Capable Device
20	0h RO	**Capabilities List (CAPL):** Capabilities List
19	0h RO	**Interrupt Status (INTS):** Interrupt Status

Bit Range	Default and Access	Field Name (ID): Description
18:11	0h RO	Reserved.
10	1h RO	**Interrupt Disable (INTD):** Interrupt Disable
9	0h RO	**Fast Back to Back Enable (FBTBEN):** Fast Back to Back Enable
8	0h RW	**System Error Enable (SERREN):** System Error Enable
7	0h RO	Reserved.
6	0h RW	**Parity Error Response (PERRR):** Parity Error Response
5	0h RO	**VGA Palette Snoop (VGAPS):** VGA Palette Snoop
4	0h RO	**Memory Write and Invalidate Enable (MWRIEN):** Memory Write and Invalidate Enable.
3	0h RO	**Special Cycles (SPCYC):** Special Cycles
2	0h RW	**Bus Master Enable (BME):** Bus Master Enable
1	0h RW	**Memory Space Enable (MSE):** Memory Space Enable
0	0h RO	**IO Space Enable (IOSE):** IO Space Enable

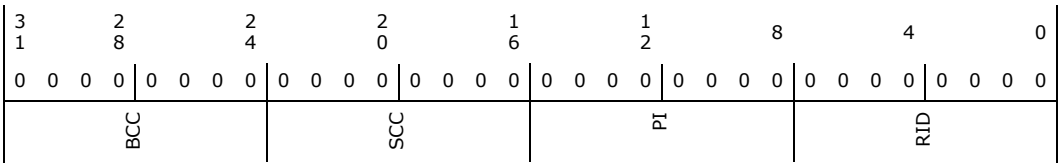
27.8.3 Revision ID (BIOS_SPI_CC_RID)—Offset 8h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	**Base Class Code (BCC):** Base Class Code

Bit Range	Default and Access	Field Name (ID): Description
23:16	0h RO	**Sub-Class Code (SCC):** Sub-Class Code
15:8	0h RO	**Programming Interface (PI):** Programming Interface
7:0	0h RO/V	**Revision ID (RID):** Revision ID (RID): Indicates the part revision.

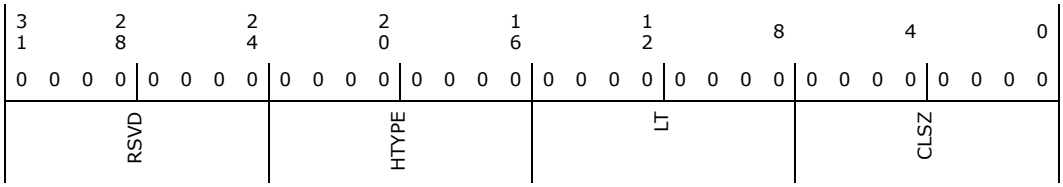
27.8.4 BIST, Header Type, Latency Timer, Cache Line Size (BIOS_SPI_BIST_HTYPE_LT_CLS)—Offset Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23:16	0h RO	**Header Type (HTYPE):** Header Type
15:8	0h RO	**Latency Timer (LT):** Latency Timer
7:0	0h RO	**Cacheline Size (CLSZ):** Cacheline Size

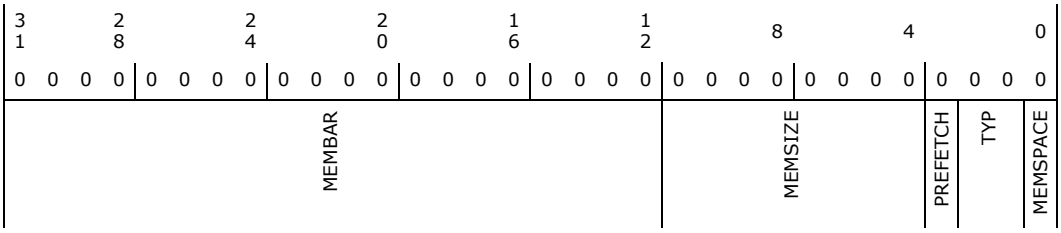
27.8.5 SPI BAR0 MMIO (BIOS_SPI_BAR0)—Offset 10h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:12	0h RW	**Memory BAR (MEMBAR):** Software programs this register with the base address of the device's memory region. The Host/BIOS MMIO registers in the flash controller are offset from this BAR.
11:4	0h RO	**Memory Size (MEMSIZE):** Hardwired to 0 to indicate 4 KB of memory space.
3	1h RO	**Prefetchable (PREFETCH):** A device can mark a range as prefetchable if there are no side effects on reads, the device returns all bytes on reads regardless of the byte enables. Hardwired to 1 to indicate the device's memory space as prefetchable.
2:1	0h RO	**Type (TYP):** Hardwired to 0 to indicate that Base register is 32 bits wide and mapping can be done anywhere in the 32-bit memory space.
0	0h RO	**Memory Space Indicator (MEMSPACE):** Hardwired to 0 to identify a Memory BAR.

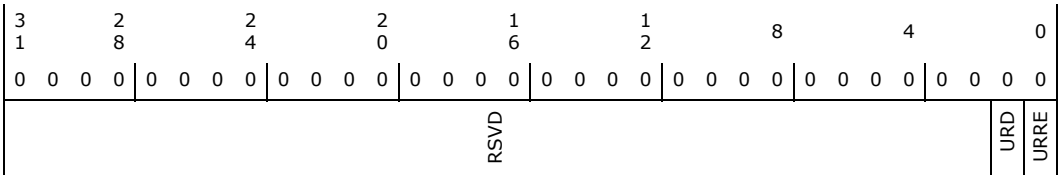
27.8.6 SPI Unsupported Request Status (BIOS_SPI_UR_STS_CTL)—Offset D0h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:2	0h RO	Reserved.
1	0h RW/1C/V	**Unsupported Request Detected (URD):** Set to 1 by hardware upon detecting an Unsupported Request that is not considered an Advisory Non-Fatal error. Cleared to 0 when software writes a 1 to this register.
0	0h RW	**Unsupported Request Reporting Enabled (URRE):** If set to 1 by software, the flash controller generates a DoSERR sideband message when the URD bit transitions from 0 to 1.

27.8.7 BIOS Decode Enable (BIOS_SPI_BDE)—Offset D8h

This register only effects BIOS decode if BIOS is resident on SPI.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 5

Default: FFCFh

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	0	0	1	1	1	1
RSVD																EF8	EF0	EE8	EE0	ED8	ED0	EC8	EC0	LFE	LEE	RSVD		E70	E60	E50	E40

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15	1h RO	**F8-FF Enable (EF8):** F8-FF Enable (EF8): Enables decoding of 512K of the following BIOS range: Data space: FFF80000h - FFFFFFFFh
14	1h RW	**F0-F8 Enable (EF0):** F0-F8 Enable (EF0): Enables decoding of 512K of the following BIOS range: Data space: FFF00000h - FFF7FFFFh
13	1h RW	**E8-EF Enable (EE8):** E8-EF Enable (EE8): Enables decoding of 512K of the following BIOS range: Data space: FFE80000h - FFEFFFFFh
12	1h RW	**E0-E8 Enable (EE0):** E0-E8 Enable (EE0): Enables decoding of 512K of the following BIOS range: Data space: FFE00000h - FFE7FFFFh
11	1h RW	**D8-DF Enable (ED8):** D8-DF Enable (ED8): Enables decoding of 512K of the following BIOS range: Data space: FFD80000h - FFDFFFFFh
10	1h RW	**D0-D7 Enable (ED0):** D0-D7 Enable (ED0): Enables decoding of 512K of the following BIOS range: Data space: FFD00000h - FFD7FFFFh
9	1h RW	**C8-CF Enable (EC8):** C8-CF Enable (EC8): Enables decoding of 512K of the following BIOS range: Data space: FFC80000h - FFCFFFFFh

Bit Range	Default and Access	Field Name (ID): Description
8	1h RW	**C0-C7 Enable (EC0):** C0-C7 Enable (EC0): Enables decoding of 512K of the following BIOS range: Data space: FFC00000h - FFC7FFFFh
7	1h RW	**Legacy F Segment Enable (LFE):** Legacy F Segment Enable (LFE): This enables the decoding of the legacy 64KB range at F0000h - FFFFFh
6	1h RW	**Legacy E Segment Enable (LEE):** Legacy E Segment Enable (LFE): This enables the decoding of the legacy 64KB range at E0000h - EFFFFh
5:4	0h RO	Reserved.
3	1h RW	**70-7F Enable (E70):** 70-7F Enable (E70): Enables decoding of 1MB of the following BIOS range: Data space: FF700000h - FF7FFFFFh
2	1h RW	**60-6F Enable (E60):** 60-6F Enable (E60): Enables decoding of 1MB of the following BIOS range: Data space: FF600000h - FF6FFFFFh
1	1h RW	**50-5F Enable (E50):** 50-5F Enable (E50): Enables decoding of 1MB of the following BIOS range: Data space: FF500000h - FF5FFFFFh
0	1h RW	**40-4F Enable (E40):** 40-4F Enable (E40): Enables decoding of 1MB of the following BIOS range: Data space: FF400000h - FF4FFFFFh

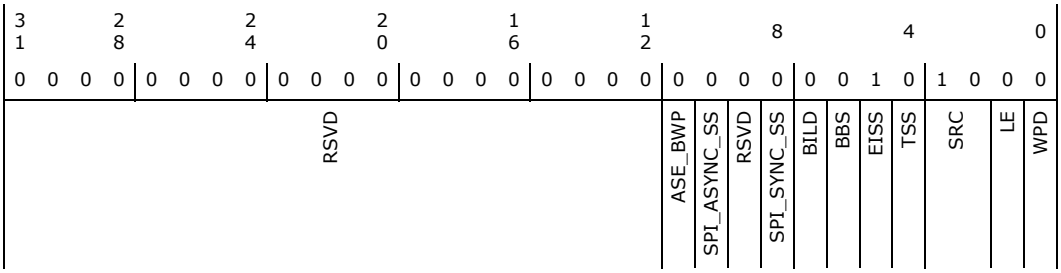
27.8.8 BIOS Control (BIOS_SPI_BC)—Offset DCh

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 28h



Bit Range	Default and Access	Field Name (ID): Description
31:12	0h RO	Reserved.
11	0h RW/L	**Async SMI Enable for BIOS Write Protection (ASE_BWP):** When set to '1' the flash controller will generate an SMI when it blocks a BIOS write or erase due to WPD=0. The value in this field can be written by software as long as the BIOS Interface Lock-Down (BILD) is not set.

Bit Range	Default and Access	Field Name (ID): Description
10	0h RO/V	**Asynchronous SMI Status (SPI_ASYNC_SS):** Status indication that the SPI Flash Controller has asserted an asynchronous SMI. Hardware clears the bit when it sends the De-assert SMI message. 0: default state 1: SPI flash controller asserted asynchronous SMI
9	0h RO	Reserved.
8	0h RO/V	**Synchronous SMI Status (SPI_SYNC_SS):** Status indication that the SPI Flash Controller has asserted a synchronous SMI. Hardware clears the bit when it sends the De-assert Synchronous SMI message. 0: default state 1: SPI flash controller asserted Synchronous SMI
7	0h RW/L	**BIOS Interface Lock-Down (BILD):** BIOS Interface Lock-Down (BILD): When set, prevents TS and BBS from being changed. This bit can only be written from 0 to 1 once.
6	0h RW/V/L	**Boot BIOS Strap (BBS):** Boot BIOS Strap (BBS): This field determines the destination of accesses to the BIOS memory range. Bit Description 0 SPI 1 LPC When SPI or LPC is selected, the range that is decoded is further qualified by BIOS Decode Enable. The value in this field can be overwritten by software as long as the BIOS Interface Lock-Down (BILD) is not set
5	1h RW/L	**Enable InSMM.STS (EISS):** Enable InSMM.STS (EISS): When this bit is set, the BIOS region is not writable until the CPU sets the InSMM.STS bit. If this bit [5] is set, then WPD must be a '1' and InSMM.STS (0xFED3_0880[0]) must be '1' also in order to write to BIOS region of SPI Flash. If this bit [5] is clear, then the InSMM.STS is a don't care. This bit is locked by LE
4	0h RO/V	**Top Swap Status (TSS):** Top Swap Status (TSS): This bit provides a read-only path to view the state of the Top Swap bit. It is duplicated here to be consistent with the LPC version of the BC register.
3:2	2h RW	**SPI Read Configuration (SRC):** SPI Read Configuration (SRC): These bits are located in PCI Config space to allow them to be set early in the boot flow. This 2-bit field controls two policies related to BIOS reads on the SPI interface: Bit 3- Prefetch Enable Bit 2- Cache Disable Settings are summarized below: Bits 3:2 Description 00 No prefetching, but caching enabled. Direct Memory reads load the read buffer cache with [quote]valid[/quote] data, allowing repeated reads to the same range to complete quickly 01 No prefetching and no caching. One-to-one correspondence of host BIOS reads to SPI cycles. This value can be used to invalidate the cache. 10 Prefetching and Caching enabled. This mode is used for long sequences of short reads to consecutive addresses (i.e., shadowing) 11 Illegal. Caching must be enabled when Prefetching is enabled. This eliminates the need for a complex prefetch-flushing mechanism.
1	0h RW/L	**Lock Enable (LE):** When set, setting the WPD bit will cause SMI. When cleared, setting the WPD bit will not cause SMI. Once set, this bit can only be cleared by a PLTRST#. When this bit is set, EISS - bit [5] of this register is locked down.
0	0h RW	**Write Protect Disable (WPD):** When set, access to the BIOS space is enabled for both read and write cycles to BIOS. When cleared, only read cycles are permitted to the FWH or SPI flash. When this bit is written from a '0' to a '1' and the LE bit is also set, an SMI# is generated. This ensures that only SMM code can update BIOS.

27.9 SPI Memory Mapped Registers Summary

The SPI memory mapped registers are accessed based upon offsets from SPI_BAR0 (in PCI config SPI_BAR0 register).

Table 27-5. Summary of SPI Memory Mapped Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	SPI BIOS MMIO PRI (BIOS_BFPREG)—Offset 0h	0h
4h	7h	Hardware Sequencing Flash Status and Control (BIOS_HSFSTS_CTL)—Offset 4h	2000h
8h	Bh	Flash Address (BIOS_FADDR)—Offset 8h	0h
Ch	Fh	Discrete Lock Bits (BIOS_DLOCK)—Offset Ch	0h
10h	13h	Flash Data 0 (BIOS_FDATA0)—Offset 10h	0h
14h	17h	Flash Data 1 (BIOS_FDATA1)—Offset 14h	0h
18h	1Bh	Flash Data 2 (BIOS_FDATA2)—Offset 18h	0h
1Ch	1Fh	Flash Data 3 (BIOS_FDATA3)—Offset 1Ch	0h
20h	23h	Flash Data 4 (BIOS_FDATA4)—Offset 20h	0h
24h	27h	Flash Data 5 (BIOS_FDATA5)—Offset 24h	0h
28h	2Bh	Flash Data 6 (BIOS_FDATA6)—Offset 28h	0h
2Ch	2Fh	Flash Data 7 (BIOS_FDATA7)—Offset 2Ch	0h
30h	33h	Flash Data 8 (BIOS_FDATA8)—Offset 30h	0h
34h	37h	Flash Data 9 (BIOS_FDATA9)—Offset 34h	0h
38h	3Bh	Flash Data 10 (BIOS_FDATA10)—Offset 38h	0h
3Ch	3Fh	Flash Data 11 (BIOS_FDATA11)—Offset 3Ch	0h
40h	43h	Flash Data 12 (BIOS_FDATA12)—Offset 40h	0h
44h	47h	Flash Data 13 (BIOS_FDATA13)—Offset 44h	0h
48h	4Bh	Flash Data 14 (BIOS_FDATA14)—Offset 48h	0h
4Ch	4Fh	Flash Data 15 (BIOS_FDATA15)—Offset 4Ch	0h
50h	53h	Flash Region Access Permissions (BIOS_FRACC)—Offset 50h	42C2h
54h	57h	Flash Region 0 (BIOS_FREG0)—Offset 54h	0h
58h	5Bh	Flash Region 1 (BIOS_FREG1)—Offset 58h	0h
5Ch	5Fh	Flash Region 2 (BIOS_FREG2)—Offset 5Ch	0h
60h	63h	Flash Region 3 (BIOS_FREG3)—Offset 60h	0h
64h	67h	Flash Region 4 (BIOS_FREG4)—Offset 64h	0h
68h	6Bh	Flash Region 5 (BIOS_FREG5)—Offset 68h	0h
6Ch	6Fh	Flash Region 6(BIOS_FREG6)—Offset 6Ch	0h
70h	73h	Flash Region 7(BIOS_FREG7)—Offset 70h	0h
74h	77h	Flash Region 8(BIOS_FREG8)—Offset 74h	0h
78h	7Bh	Flash Region 9(BIOS_FREG9)—Offset 78h	0h
7Ch	7Fh	Flash Region 10(BIOS_FREG10)—Offset 7Ch	0h
80h	84h	Flash Region 11(BIOS_FREG11)—Offset 80h	0h
84h	87h	Flash Protected Range 0 (BIOS_FPR0)—Offset 84h	0h
88h	8Bh	Flash Protected Range 1 (BIOS_FPR1)—Offset 88h	0h
8Ch	8Fh	Flash Protected Range 2 (BIOS_FPR2)—Offset 8Ch	0h

Table 27-5. Summary of SPI Memory Mapped Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
90h	93h	Flash Protected Range 3 (BIOS_FPR3)—Offset 90h	0h
94h	97h	Flash Protected Range 4 (BIOS_FPR4)—Offset 94h	0h
98h	9Bh	Global Protected Range 0 (BIOS_GPR0)—Offset 98h	0h
A0h	A3h	Software Sequencing Flash Status and Control (BIOS_SSFSTS_CTL)—Offset A0h	6000000h
A4h	A7h	Prefix Opcode and Opcode Type Configuration (BIOS_PREOP_OPTYPE)—Offset A4h	0h
A8h	ABh	Opcode Menu0 Configuration (BIOS_OPMENU0)—Offset A8h	0h
ACh	AFh	Opcode Menu1 Configuration (BIOS_OPMENU1)—Offset ACh	0h
B0h	B3h	Secondary Flash Region Access Permissions (BIOS_SFRACC)—Offset B0h	0h
B4h	B7h	Flash Descriptor Observability Control (BIOS_FDOC)—Offset B4h	0h
B8h	BBh	Flash Descriptor Observability Data (BIOS_FDOD)—Offset B8h	0h
C0h	C3h	Additional Flash Control (BIOS_AFC)—Offset C0h	0h
C4h	C7h	Vendor Specific Component Capabilities for Component 0 (BIOS_SFDP0_VSCC0)—Offset C4h	2000h
C8h	CBh	Vendor Specific Component Capabilities for Component 1 (BIOS_SFDP1_VSCC1)—Offset C8h	2000h
CCh	CFh	Parameter Table Index (BIOS_PTINX)—Offset CCh	0h
D0h	D3h	Parameter Table Data (BIOS_PTDATA)—Offset D0h	0h
D4h	D7h	SPI Bus Requester Status (BIOS_SBRS)—Offset D4h	0h

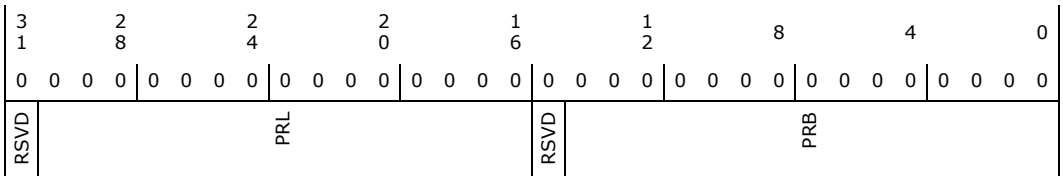
27.9.1 SPI BIOS MMIO PRI (BIOS_BFPREG)—Offset 0h

Access Method

Type: MEM Register (Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RO	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
30:16	0h RO/V	**BIOS Flash Primary Region Limit (PRL):** This specifies address bits 26:12 for the Primary Region Limit. The value in this register is loaded from the contents in the Flash Descriptor.FLREG1.Region Limit, or the Flash Descriptor.FLREG6.Region Limit depending on the BFPREG.SBRS bit.
15	0h RO	Reserved.
14:0	0h RO/V	**BIOS Flash Primary Region Base (PRB):** This specifies address bits 26:12 for the Primary Region Base. The value in this register is loaded from the contents in the Flash Descriptor.FLREG1.Region Base, or the Flash Descriptor.FLREG6.Region Base depending on the BFPREG.SBRS bit.

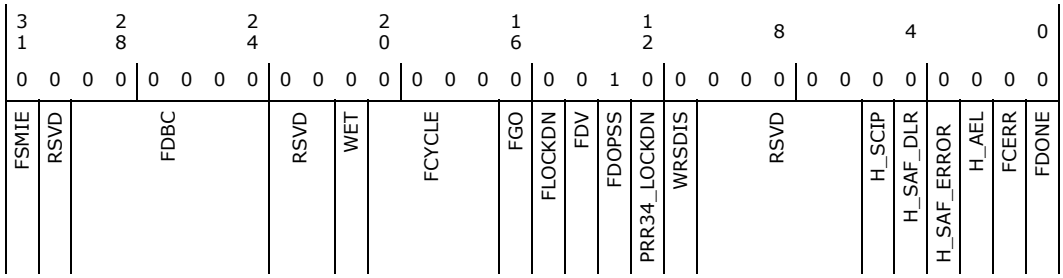
27.9.2 Hardware Sequencing Flash Status and Control (BIOS_HSFSTS_CTL)—Offset 4h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 2000h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RW	**Flash SPI SMI# Enable (FSMIE):** When set to 1, the SPI asserts an SMI# request whenever the Flash Cycle Done bit is 1.
30	0h RO	Reserved.
29:24	0h RW	**Flash Data Byte Count (FDBC):** This field specifies the number of bytes to shift in or out during the data portion of the SPI cycle. The contents of this register are 0s based with 0b representing 1 byte and 3Fh representing 64 bytes. The number of bytes transferred is the value of this field plus 1. This field is ignored for the Block Erase command.
23:22	0h RO	Reserved.
21	0h RW	**Write Enable Type (WET):** Write Enable Type (WET) 0 Use 06h as the write enable instruction 1 Use 50h as the write enable instruction

Bit Range	Default and Access	Field Name (ID): Description
20:17	0h RW	**FLASH Cycle (FCYCLE):** This field defines the Flash SPI cycle type generated to the FLASH when the FGO bit is set as defined below: 0 Read (1 up to 64 bytes by setting FDBC) 1 Reserved 2 Write (1 up to 64 bytes by setting FDBC) 3 4k Block Erase 4 64k Sector erase 5 Read SFDP 6 Read JEDEC ID 7 write status 8 read status 9 RPMC Op1 A RPMC Op2 Flash controller hardware automatically inserts a write enable opcode prior to Write and erase operations. Hardware automatically polls for device not-busy using read status after write and erase operations. Flash controller hardware automatically polls for device not-busy using read status and RPMC Op2 after RPMC Op1 operations. Read status is repeated until the device is not-busy, then RPMC Op2 is repeated until the device is not-busy. If the device does not support 64k erase size (or if it doesn't support SFDP) then only 4k is allowed. ***Note:*** If reserved 1 is programmed to this field, flash controller will handle it as if it is 0 (Read)
16	0h RW/1S/V	**Flash Cycle Go (FGO):** A write to this register with a 1 in this bit initiates a request to the Flash SPI Arbiter to start a cycle. This register is cleared by hardware when the cycle is granted by the SPI arbiter to run the cycle on the SPI bus. When the cycle is complete, the FDONE bit is set. Software is forbidden to write to any register in the HSFLCTL register between the FGO bit getting set and the FDONE bit being cleared. Any attempt to violate this rule will be ignored by hardware. Hardware allows other bits in this register to be programmed for the same transaction when writing this bit to 1. This saves an additional memory write. This bit always returns 0 on reads.
15	0h RW/L	**Flash Configuration Lock-Down (FLOCKDN):** When set to 1, those Flash Program Registers that are locked down by this FLOCKDN bit cannot be written. Once set to 1, this bit can only be cleared by a hardware reset.
14	0h RO/V	**Flash Descriptor Valid (FDV):** This bit is set to a 1 if the Flash Controller read the correct Flash Descriptor Signature. If the Flash Descriptor Valid bit is not 1, software cannot use the Hardware Sequencing registers, but must use the software sequencing registers. Any attempt to use the Hardware Sequencing registers will result in the FCERR bit being set
13	1h RO/V	**Flash Descriptor Override Pin-Strap Status (FDOPSS):** This register reflects the value the Flash Descriptor Override Pin-Strap. '1': No override '0': The Flash Descriptor Override strap is set
12	0h RW/L	**PRR3 PRR4 Lock-Down (PRR34_LOCKDN):** When set to 1, the BIOS PRR3 and PRR4 registers cannot be written. Once set to 1, this bit can only be cleared by a hardware reset.
11	0h RW/L	**Flash Configuration Lock-Down (WRSDIS):** '0': write status operation may be issued using Hardware Sequencing '1': write status is not allowed as a Hardware Sequencing operation. The flash controller will block the operation and set the FCERR bit when software sets the 'go' bit.
10:6	0h RO	Reserved.
5	0h RO/V	**SPI Cycle In Progress (H_SCIP):** Hardware sets this bit when software sets the Flash Cycle Go (FGO) bit in the Hardware Sequencing Flash Control register. This bit remains set until the cycle completes on the SPI interface. Hardware automatically sets and clears this bit so that software can determine when read data is valid and/or when it is safe to begin programming the next command. Software must only program the next command when this bit is 0.
4	0h RW/1C/V	**SAF Data Length Error (H_SAF_DLE)**: Hardware sets this bit to 1 when a transaction is returned from the eSPI channel with an incorrect data length.
3	0h RW/1C/V	**SAF Error (H_SAF_ERROR)**: Hardware sets this bit to 1 when a transaction that is requested that is not supported by slave attached flash, e.g. read status.

Bit Range	Default and Access	Field Name (ID): Description
2	0h RW/1C/V	**Access Error Log (H_AEL):** Hardware sets this bit to a 1 when an attempt was made to access the BIOS region using the direct access method or an access to the BIOS Program Registers that violated the security restrictions. This bit is simply a log of an access security violation. This bit is cleared by software writing a '1'.
1	0h RW/1C/V	**Flash Cycle Error (FCERR):** Hardware sets this bit to 1 when a program register access is blocked to the FLASH due to one of the protection policies or when any of the programmed cycle registers is written while a programmed access is already in progress. This bit remains asserted until cleared by software writing a 1 or until hardware reset occurs. Software must clear this bit before setting the FLASH Cycle GO bit in this register.
0	0h RW/1C/V	**Flash Cycle Done (FDONE):** The PCH sets this bit to 1 when the SPI Cycle completes after software previously set the FGO bit. This bit remains asserted until cleared by software writing a 1 or hardware reset. When this bit is set and the SPI SMI# Enable bit is set, an internal signal is asserted to the SMI# generation block. Software must make sure this bit is cleared prior to enabling the SPI SMI# assertion for a new programmed access.

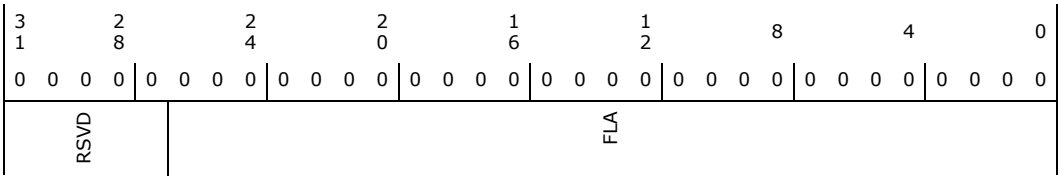
27.9.3 Flash Address (BIOS_FADDR)—Offset 8h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:27	0h RO	Reserved.
26:0	0h RW	**Flash Linear Address (FLA):** The FLA is the starting byte linear address of a SPI Read or Write cycle or an address within a Block for the Block Erase command. The Flash Linear Address must fall within a region for which BIOS has access permissions. Hardware must convert the FLA into a Flash Physical Address (FPA) before running this cycle on the SPI bus.

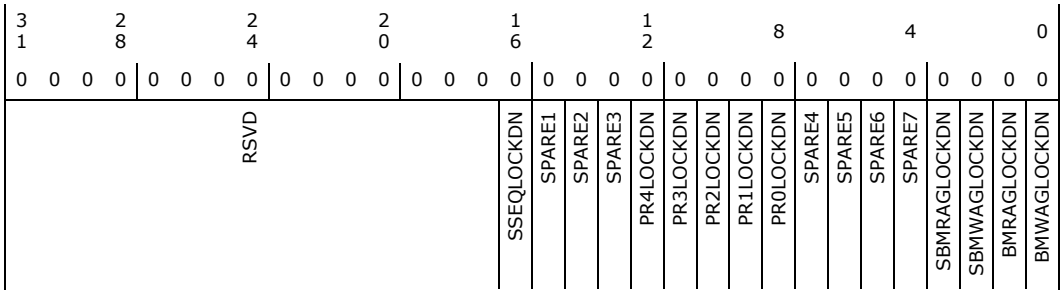
27.9.4 Discrete Lock Bits (BIOS_DLOCK)—Offset Ch

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:17	0h RO	Reserved.
16	0h RW/L	**SSEQ Lock-Down (SSEQLOCKDN)**
15	0h RW/L	**Spare1 (SPARE1)**
14	0h RW/L	**Spare2 (SPARE2)**
13	0h RW/L	**Spare3 (SPARE3)**
12	0h RW/L	**PR4 Lock-Down (PR4LOCKDN)**
11	0h RW/L	**PR3 Lock-Down (PR3LOCKDN)**
10	0h RW/L	**PR2 Lock-Down (PR2LOCKDN)**
9	0h RW/L	**PR1 Lock-Down (PR1LOCKDN)**
8	0h RW/L	**PR0 Lock-Down (PR0LOCKDN)**
7	0h RW/L	**Spare4 (SPARE4)**
6	0h RW/L	**Spare5 (SPARE5)**
5	0h RW/L	**Spare6 (SPARE6)**
4	0h RW/L	**Spare7 (SPARE7)**
3	0h RW/L	**SBMRAG Lock-Down (SBMRAGLOCKDN)**

Bit Range	Default and Access	Field Name (ID): Description
2	0h RW/L	**SBMWAG Lock-Down (SBMWAGLOCKDN)**
1	0h RW/L	**BMRAG Lock-Down (BMRAGLOCKDN)**
0	0h RW/L	**BMWAG Lock-Down (BMWAGLOCKDN)**

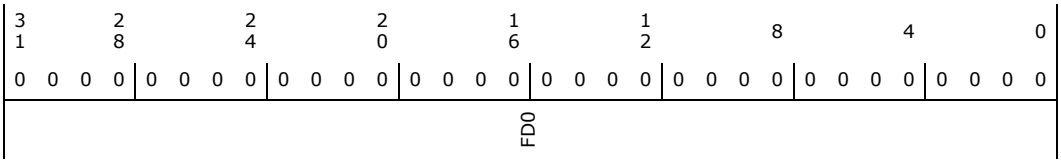
27.9.5 Flash Data 0 (BIOS_FDATA0)—Offset 10h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/V	**Flash Data 0 (FD0):** This field is shifted out as the SPI Data on the Master-Out Slave-In Data pin during the data portion of the SPI cycle. This register also shifts in the data from the Master-In Slave-Out pin into this register during the data portion of the SPI cycle. The data is always shifted starting with the least significant byte, msb to lsb, followed by the next least significant byte, msb to lsb, etc. Specifically, the shift order on SPI in terms of bits within this register is: 7-6-5-4-3-2-1-0-15-14-13-...8-23-22-...16-31...24. Bit 24 is the last bit shifted out/in. There are no alignment assumptions; byte 0 always represents the value specified by the cycle address. Note that the data in this register may be modified by the hardware during any programmed SPI transaction. Direct Memory Reads do not modify the contents of this register.

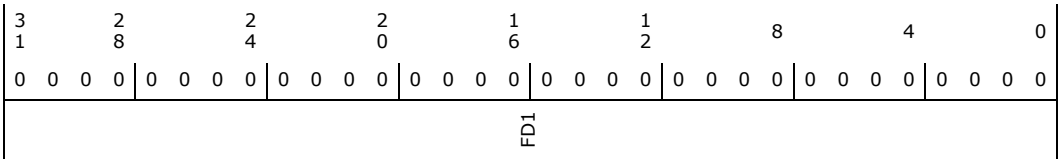
27.9.6 Flash Data 1 (BIOS_FDATA1)—Offset 14h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/V	**Flash Data 1 (FD1):** Flash Data 1 (FD1): Similar definition as Flash Data 0. However, this register does not begin shifting until FD0 has completely shifted in/out.

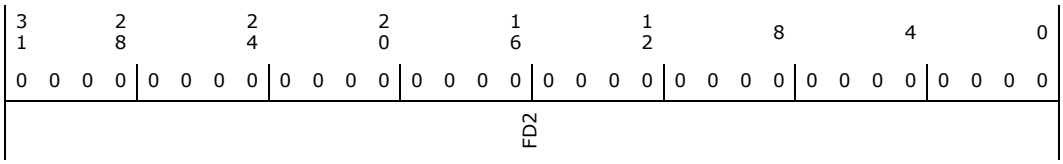
27.9.7 Flash Data 2 (BIOS_FDATA2)—Offset 18h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/V	**Flash Data 2 (FD2):** Flash Data 2 (FD2): Similar definition as Flash Data 0. However, this register does not begin shifting until FD1 has completely shifted in/out.

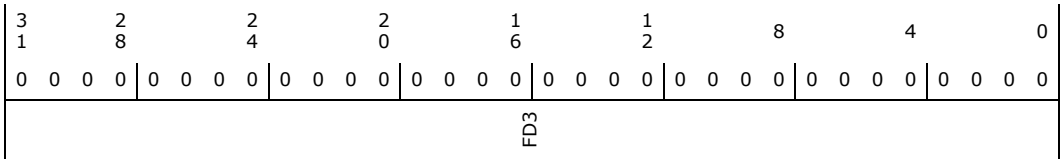
27.9.8 Flash Data 3 (BIOS_FDATA3)—Offset 1Ch

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/V	**Flash Data 3 (FD3):** Flash Data 3 (FD3): Similar definition as Flash Data 0. However, this register does not begin shifting until FD2 has completely shifted in/out.

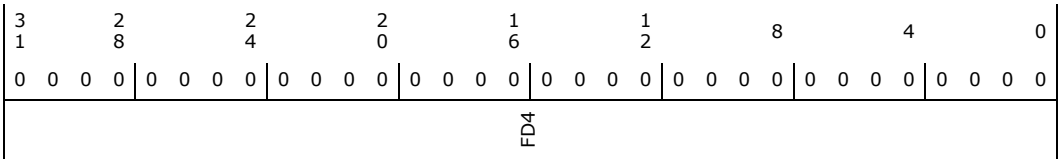
27.9.9 Flash Data 4 (BIOS_FDATA4)—Offset 20h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/V	**Flash Data 4 (FD4):** Flash Data 4 (FD4): Similar definition as Flash Data 0. However, this register does not begin shifting until FD3 has completely shifted in/out.

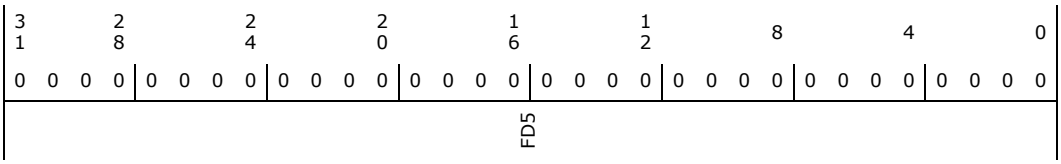
27.9.10 Flash Data 5 (BIOS_FDATA5)—Offset 24h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/V	**Flash Data 5 (FD5):** Flash Data 5 (FD5): Similar definition as Flash Data 0. However, this register does not begin shifting until FD4 has completely shifted in/out.

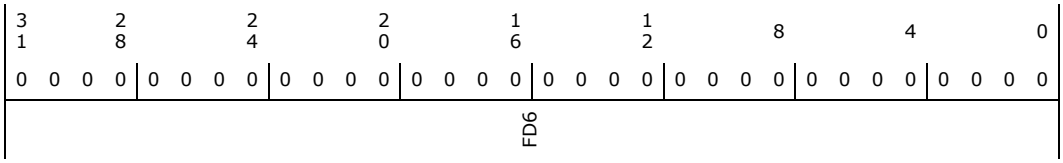
27.9.11 Flash Data 6 (BIOS_FDATA6)—Offset 28h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/V	**Flash Data 6 (FD6):** Flash Data 6 (FD6): Similar definition as Flash Data 0. However, this register does not begin shifting until FD5 has completely shifted in/out.

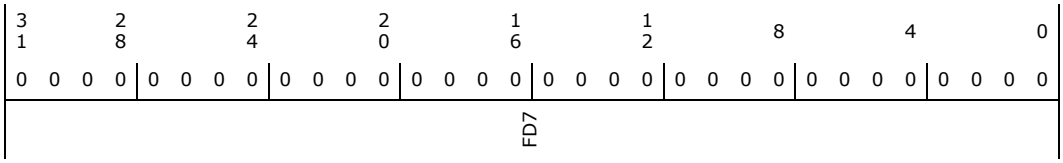
27.9.12 Flash Data 7 (BIOS_FDATA7)—Offset 2Ch

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/V	**Flash Data 7 (FD7):** Flash Data 7 (FD7): Similar definition as Flash Data 0. However, this register does not begin shifting until FD6 has completely shifted in/out.

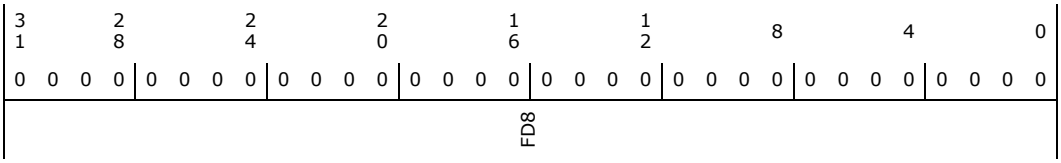
27.9.13 Flash Data 8 (BIOS_FDATA8)—Offset 30h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/V	**Flash Data 8 (FD8):** Flash Data 8 (FD8): Similar definition as Flash Data 0. However, this register does not begin shifting until FD7 has completely shifted in/out.

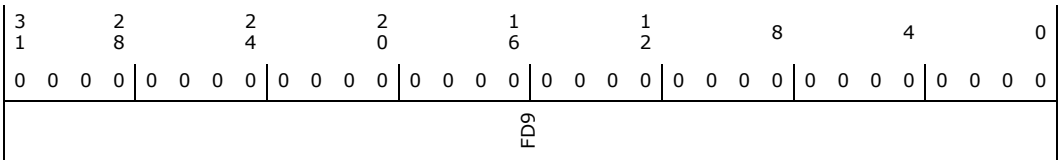
27.9.14 Flash Data 9 (BIOS_FDATA9)—Offset 34h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/V	**Flash Data 9 (FD9):** Flash Data 9 (FD9): Similar definition as Flash Data 0. However, this register does not begin shifting until FD8 has completely shifted in/out.

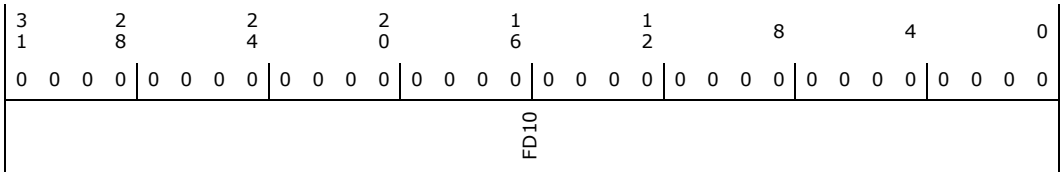
27.9.15 Flash Data 10 (BIOS_FDATA10)—Offset 38h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/V	**Flash Data 10 (FD10):** Flash Data 10 (FD10): Similar definition as Flash Data 0. However, this register does not begin shifting until FD9 has completely shifted in/out.

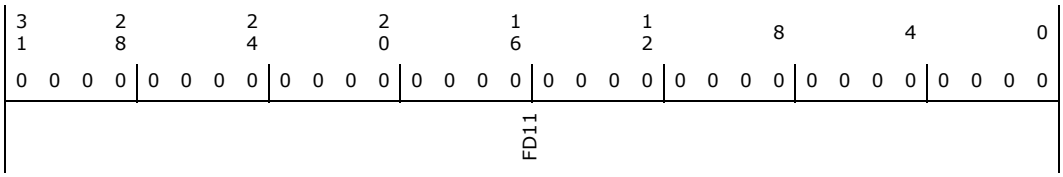
27.9.16 Flash Data 11 (BIOS_FDATA11)—Offset 3Ch

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/V	**Flash Data 11 (FD11):** Flash Data 11 (FD11): Similar definition as Flash Data 0. However, this register does not begin shifting until FD10 has completely shifted in/out.

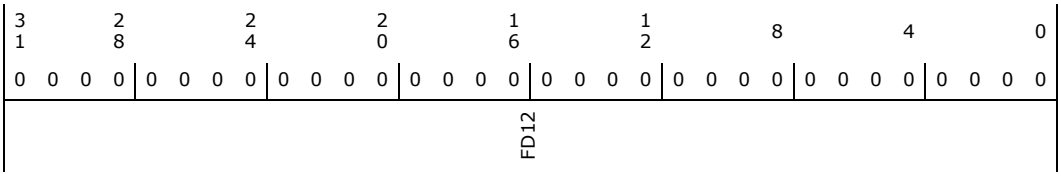
27.9.17 Flash Data 12 (BIOS_FDATA12)—Offset 40h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/V	**Flash Data 12 (FD12):** Flash Data 12 (FD12): Similar definition as Flash Data 0. However, this register does not begin shifting until FD11 has completely shifted in/out.

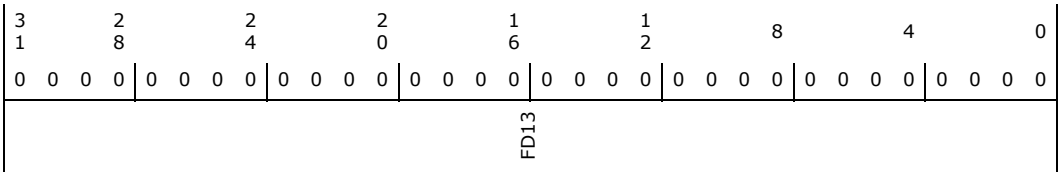
27.9.18 Flash Data 13 (BIOS_FDATA13)—Offset 44h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/V	**Flash Data 13 (FD13):** Flash Data 13 (FD13): Similar definition as Flash Data 0. However, this register does not begin shifting until FD12 has completely shifted in/out.

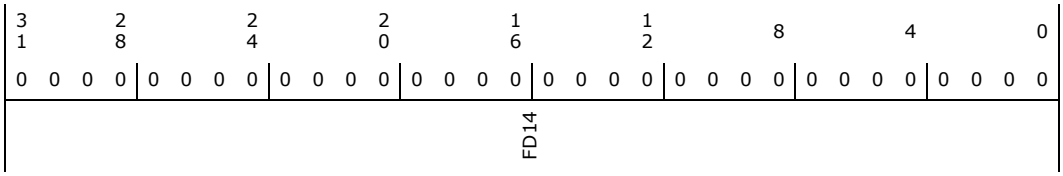
27.9.19 Flash Data 14 (BIOS_FDATA14)—Offset 48h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/V	**Flash Data 14 (FD14):** Flash Data 14 (FD14): Similar definition as Flash Data 0. However, this register does not begin shifting until FD13 has completely shifted in/out.

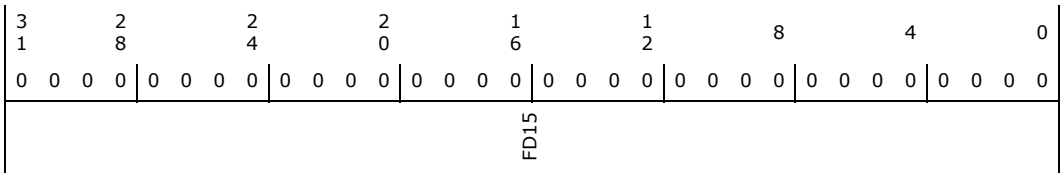
27.9.20 Flash Data 15 (BIOS_FDATA15)—Offset 4Ch

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/V	**Flash Data 15 (FD15):** Flash Data 15 (FD15): Similar definition as Flash Data 0. However, this register does not begin shifting until FD14 has completely shifted in/out.

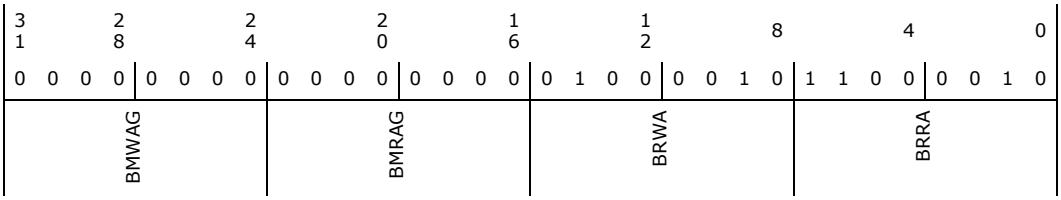
27.9.21 Flash Region Access Permissions (BIOS_FRACC)—Offset 50h

Access Method

Type: MEM Register (Size: 32 bits)

Device: 31
Function: 5

Default: 42C2h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RW/L	**BIOS Master Write Access Grant (BMWAG):** BIOS Master Write Access Grant (BMWAG): Each bit [31:24] corresponds to Master[7:0]. BIOS can grant one or more masters write access to the BIOS region 1 overriding the permissions in the Flash Descriptor. The contents of this register are locked by the FLOCKDN bit.
23:16	0h RW/L	**BIOS Master Read Access Grant (BMRAG):** BIOS Master Read Access Grant (BMRAG): Each bit [23:16] corresponds to Master[7:0]. BIOS can grant one or more masters read access to the BIOS region 1 overriding the read permissions in the Flash Descriptor. The contents of this register are locked by the FLOCKDN bit
15:8	42h RO/V	**BIOS Region Write Access (BRWA):** BIOS Region Write Access (BRWA): Each bit [15:8] corresponds to Regions [7:0]. If the bit is set, this master can erase and write that particular region through register accesses. The contents of this register are that of the Flash Descriptor.Flash Master 1.Master Region Write Access OR a particular master has granted BIOS write permissions in their Master Write Access Grant register OR the Flash Descriptor Security Override strap is set. BIOS always have the write access to its own Region 1 and Region 6 by default.
7:0	C2h RO/V	**BIOS Region Read Access (BRRA):** BIOS Region Read Access (BRRA): Each bit [7:0] corresponds to Regions [7:0]. If the bit is set, this master can read that particular region through register accesses. The contents of this register are that of the Flash Descriptor.Flash Master 1.Master Region Write Access OR a particular master has granted BIOS read permissions in their Master Read Access Grant register OR the Flash Descriptor Security Override strap is set. BIOS always have the read access to its own Region 1 and Region 6 by default.

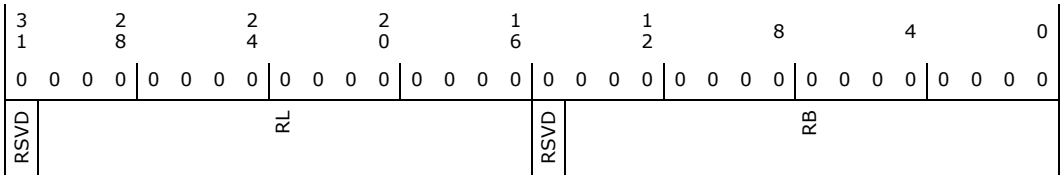
27.9.22 Flash Region 0 (BIOS_FREG0)—Offset 54h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 00000000h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RO	Reserved.
30:16	0h RO/V	**Region Limit (RL):** Region Limit (RL): This specifies address bits 26:12 for the Region 0 Limit. The value in this register is loaded from the contents in the Flash Descriptor.FLREG0.Region Limit
15	0h RO	Reserved.
14:0	0h RO/V	**Region Base (RB):** Region Base (RB): This specifies address bits 26:12 for the Region 0 Base. The value in this register is loaded from the contents in the Flash Descriptor.FLREG0.Region Base

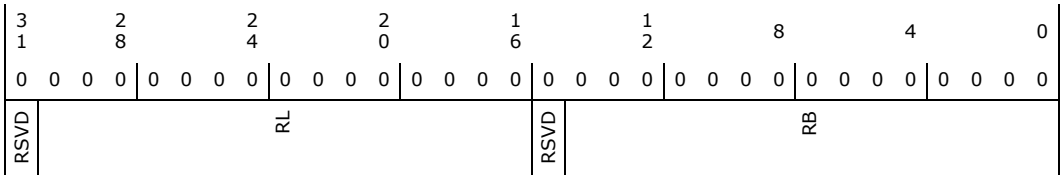
27.9.23 Flash Region 1 (BIOS_FREG1)—Offset 58h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 00000000h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RO	Reserved.
30:16	0h RO/V	**Region Limit (RL):** Region Limit (RL): This specifies address bits 26:12 for the Region 1 Limit. The value in this register is loaded from the contents in the Flash Descriptor.FLREG1.Region Limit
15	0h RO	Reserved.
14:0	0h RO/V	**Region Base (RB):** Region Base (RB): This specifies address bits 26:12 for the Region 1 Base. The value in this register is loaded from the contents in the Flash Descriptor.FLREG1.Region Base

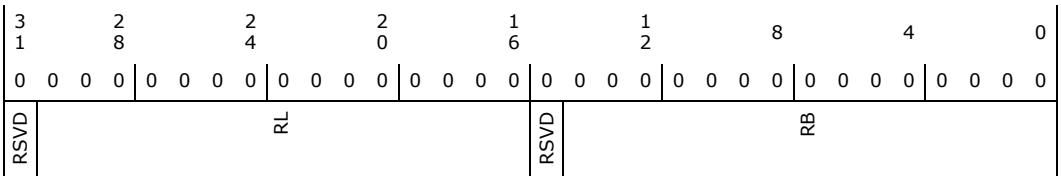
27.9.24 Flash Region 2 (BIOS_FREG2)—Offset 5Ch

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 00000000h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RO	Reserved.
30:16	0h RO/V	**Region Limit (RL):** Region Limit (RL): This specifies address bits 26:12 for the Region 2 Limit. The value in this register is loaded from the contents in the Flash Descriptor.FLREG2.Region Limit
15	0h RO	Reserved.
14:0	0h RO/V	**Region Base (RB):** Region Base (RB): This specifies address bits 26:12 for the Region 2 Base. The value in this register is loaded from the contents in the Flash Descriptor.FLREG2.Region Base

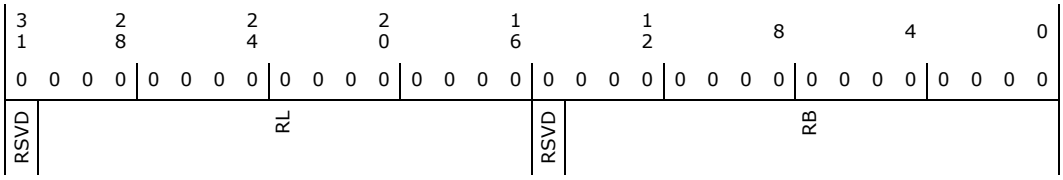
27.9.25 Flash Region 3 (BIOS_FREG3)—Offset 60h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 00000000h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RO	Reserved.
30:16	0h RO/V	**Region Limit (RL):** Region Limit (RL): This specifies address bits 26:12 for the Region 3 Limit. The value in this register is loaded from the contents in the Flash Descriptor.FLREG3.Region Limit
15	0h RO	Reserved.
14:0	0h RO/V	**Region Base (RB):** Region Base (RB): This specifies address bits 26:12 for the Region 3 Base. The value in this register is loaded from the contents in the Flash Descriptor.FLREG3.Region Base

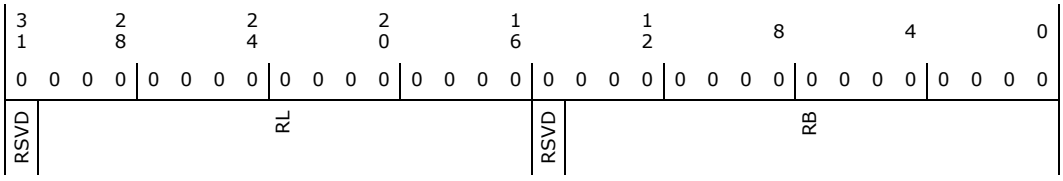
27.9.26 Flash Region 4 (BIOS_FREG4)—Offset 64h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 00000000h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RO	Reserved.
30:16	0h RO/V	**Region Limit (RL):** Region Limit (RL): This specifies address bits 26:12 for the Region 4 Limit. The value in this register is loaded from the contents in the Flash Descriptor.FLREG4.Region Limit
15	0h RO	Reserved.
14:0	0h RO/V	**Region Base (RB):** Region Base (RB): This specifies address bits 26:12 for the Region 4 Base. The value in this register is loaded from the contents in the Flash Descriptor.FLREG4.Region Base

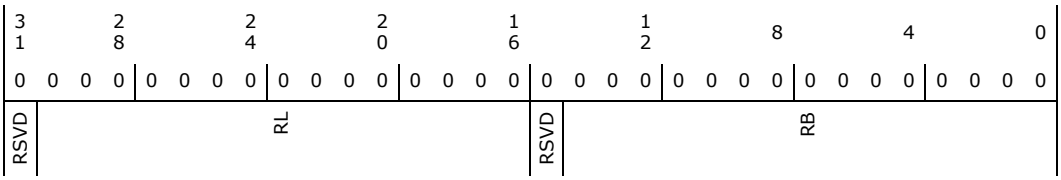
27.9.27 Flash Region 5 (BIOS_FREG5)—Offset 68h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 00000000h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RO	Reserved.
30:16	0h RO/V	**Region Limit (RL):** Region Limit (RL): This specifies address bits 26:12 for the Region 5 Limit. The value in this register is loaded from the contents in the Flash Descriptor.FLREG5.Region Limit
15	0h RO	Reserved.
14:0	0000h RO/V	**Region Base (RB):** Region Base (RB): This specifies address bits 26:12 for the Region 5 Base. The value in this register is loaded from the contents in the Flash Descriptor.FLREG5.Region Base

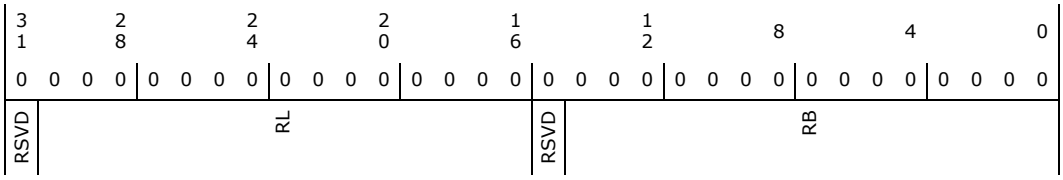
27.9.28 Flash Region 6 (BIOS_FREG6)—Offset 6Ch

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 00000000h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RO	Reserved.
30:16	0h RO/V	**Region Limit (RL):** Region Limit (RL): This specifies address bits [26:12] for the Region 5 Limit. The value in this register is loaded from the contents in the Flash Descriptor.FLREG5.Region Limit
15	0h RO	Reserved.
14:0	0h RO/V	**Region Base (RB):** Region Base (RB): This specifies address bits [26:12] for the Region 5 Base. The value in this register is loaded from the contents in the Flash Descriptor.FLREG5.Region Base

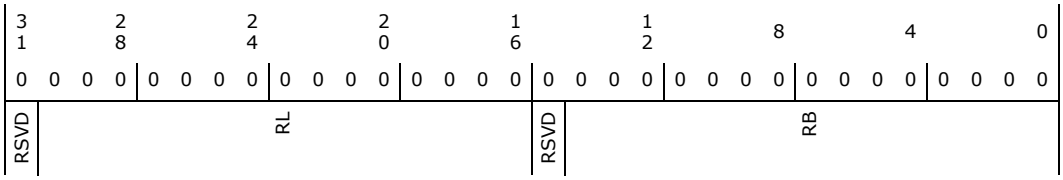
27.9.29 Flash Region 7(BIOS_FREG7)—Offset 70h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 00000000h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RO	Reserved.
30:16	0h RO/V	**Region Limit (RL):** Region Limit (RL): This specifies address bits [26:12] for the Region 7Limit. The value in this register is loaded from the contents in the Flash Descriptor.FLREG7.Region Limit
15	0h RO	Reserved.
14:0	0h RO/V	**Region Base (RB):** Region Base (RB): This specifies address bits [26:12] for the Region 7Base. The value in this register is loaded from the contents in the Flash Descriptor.FLREG7.Region Base

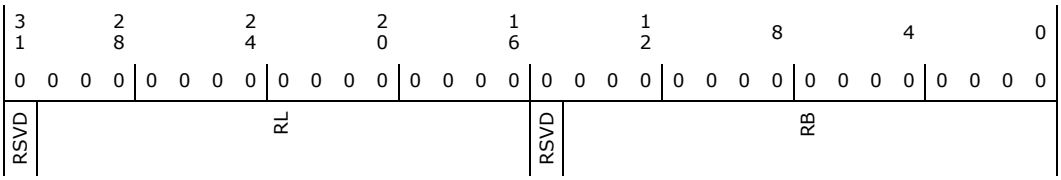
27.9.30 Flash Region 8(BIOS_FREG8)—Offset 74h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 00000000h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RO	Reserved.
30:16	0h RO/V	**Region Limit (RL):** Region Limit (RL): This specifies address bits 26:12 for the Region 8Limit. The value in this register is loaded from the contents in the Flash Descriptor.FLREG8.Region Limit
15	0h RO	Reserved.
14:0	0h RO/V	**Region Base (RB):** Region Base (RB): This specifies address bits 26:12 for the Region 8Base. The value in this register is loaded from the contents in the Flash Descriptor.FLREG8.Region Base

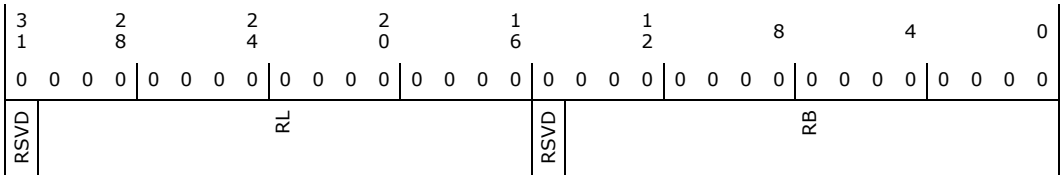
27.9.31 Flash Region 9(BIOS_FREG9)—Offset 78h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 00000000h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RO	Reserved.
30:16	0h RO/V	**Region Limit (RL):** Region Limit (RL): This specifies address bits [26:12] for the Region 9Limit. The value in this register is loaded from the contents in the Flash Descriptor.FLREG9.Region Limit
15	0h RO	Reserved.
14:0	0h RO/V	**Region Base (RB):** Region Base (RB): This specifies address bits [26:12] for the Region 9Base. The value in this register is loaded from the contents in the Flash Descriptor.FLREG9.Region Base

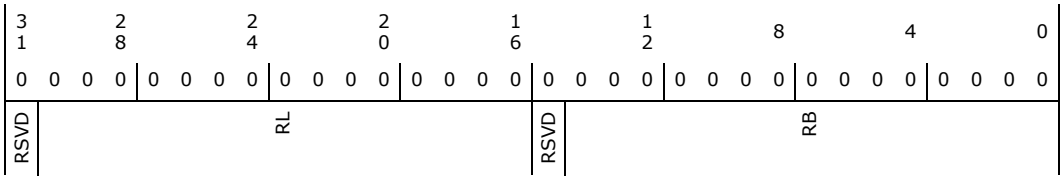
27.9.32 Flash Region 10(BIOS_FREG10)—Offset 7Ch

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 00000000h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RO	Reserved.
30:16	0h RO/V	**Region Limit (RL):** Region Limit (RL): This specifies address bits [26:12] for the Region 10Limit. The value in this register is loaded from the contents in the Flash Descriptor.FLREG510Region Limit
15	0h RO	Reserved.
14:0	0h RO/V	**Region Base (RB):** Region Base (RB): This specifies address bits [26:12] for the Region 10Base. The value in this register is loaded from the contents in the Flash Descriptor.FLREG510Region Base

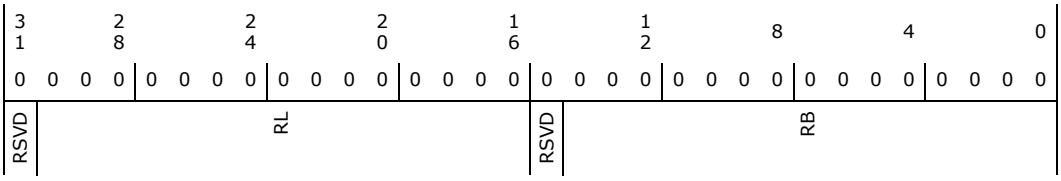
27.9.33 Flash Region 11 (BIOS_FREG11)—Offset 80h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 00000000h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RO	Reserved.
30:16	0h RO/V	**Region Limit (RL):** Region Limit (RL): This specifies address bits [26:12] for the Region11 Limit. The value in this register is loaded from the contents in the Flash Descriptor.FLREG11.Region Limit
15	0h RO	Reserved.
14:0	0h RO/V	**Region Base (RB):** Region Base (RB): This specifies address bits [26:12] for the Region 11Base. The value in this register is loaded from the contents in the Flash Descriptor.FLREG11.Region Base

27.9.34 Flash Protected Range 0 (BIOS_FPR0)—Offset 84h

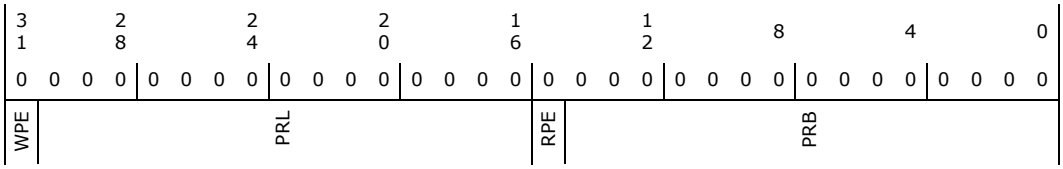
This register cannot be written when the FLOCKDN bit is set to 1.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RW/L	**Write Protection Enable (WPE):** Write Protection Enable: When set, this bit indicates that the Base and Limit fields in this register are valid and that writes and erases directed to addresses between them (inclusive) must be blocked by hardware. The base and limit fields are ignored when this bit is cleared.
30:16	0h RW/L	**Protected Range Limit (PRL):** Protected Range Limit: This field corresponds to FLA address bits 26:12 and specifies the upper limit of the protected range. Address bits 11:0 are assumed to be FFFh for the limit comparison. Any address greater than the value programmed in this field is unaffected by this protected range.
15	0h RW/L	**Read Protection Enable (RPE):** Read Protection Enable: When set, this bit indicates that the Base and Limit fields in this register are valid and that read directed to addresses between them (inclusive) must be blocked by hardware. The base and limit fields are ignored when this bit is cleared.
14:0	0h RW/L	**Protected Range Base (PRB):** Protected Range Base: This field corresponds to FLA address bits 26:12 and specifies the lower base of the protected range. Address bits 11:0 are assumed to be 000h for the base comparison. Any address less than the value programmed in this field is unaffected by this protected range.

27.9.35 Flash Protected Range 1 (BIOS_FPR1)—Offset 88h

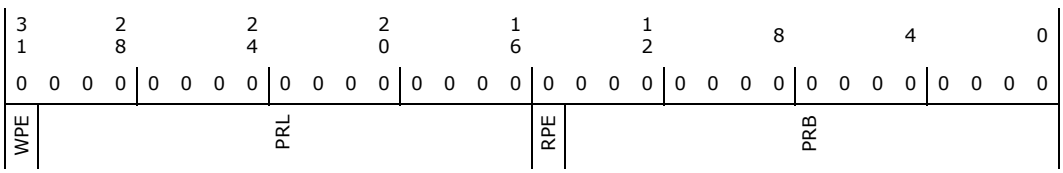
This register cannot be written when the FLOCKDN bit is set to 1.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RW/L	**Write Protection Enable (WPE):** Write Protection Enable: When set, this bit indicates that the Base and Limit fields in this register are valid and that writes and erases directed to addresses between them (inclusive) must be blocked by hardware. The base and limit fields are ignored when this bit is cleared.
30:16	0h RW/L	**Protected Range Limit (PRL):** Protected Range Limit: This field corresponds to FLA address bits 26:12 and specifies the upper limit of the protected range. Address bits 11:0 are assumed to be FFFh for the limit comparison. Any address greater than the value programmed in this field is unaffected by this protected range.
15	0h RW/L	**Read Protection Enable (RPE):** Read Protection Enable: When set, this bit indicates that the Base and Limit fields in this register are valid and that read directed to addresses between them (inclusive) must be blocked by hardware. The base and limit fields are ignored when this bit is cleared.
14:0	0h RW/L	**Protected Range Base (PRB):** Protected Range Base: This field corresponds to FLA address bits 26:12 and specifies the lower base of the protected range. Address bits 11:0 are assumed to be 000h for the base comparison. Any address less than the value programmed in this field is unaffected by this protected range.

27.9.36 Flash Protected Range 2 (BIOS_FPR2)—Offset 8Ch

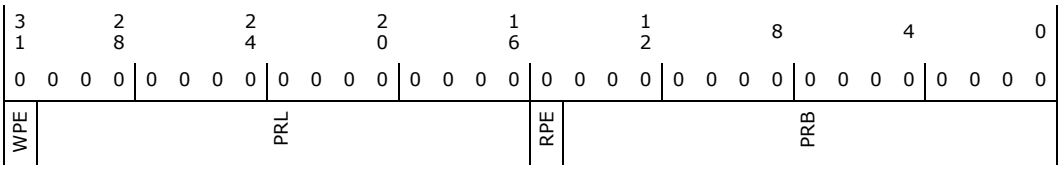
This register cannot be written when the FLOCKDN bit is set to 1.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RW/L	**Write Protection Enable (WPE):** Write Protection Enable: When set, this bit indicates that the Base and Limit fields in this register are valid and that writes and erases directed to addresses between them (inclusive) must be blocked by hardware. The base and limit fields are ignored when this bit is cleared.
30:16	0h RW/L	**Protected Range Limit (PRL):** Protected Range Limit: This field corresponds to FLA address bits 26:12 and specifies the upper limit of the protected range. Address bits 11:0 are assumed to be FFFh for the limit comparison. Any address greater than the value programmed in this field is unaffected by this protected range.
15	0h RW/L	**Read Protection Enable (RPE):** Read Protection Enable: When set, this bit indicates that the Base and Limit fields in this register are valid and that read directed to addresses between them (inclusive) must be blocked by hardware. The base and limit fields are ignored when this bit is cleared.

Bit Range	Default and Access	Field Name (ID): Description
14:0	0h RW/L	**Protected Range Base (PRB):** Protected Range Base: This field corresponds to FLA address bits 26:12 and specifies the lower base of the protected range. Address bits 11:0 are assumed to be 000h for the base comparison. Any address less than the value programmed in this field is unaffected by this protected range.

27.9.37 Flash Protected Range 3 (BIOS_FPR3)—Offset 90h

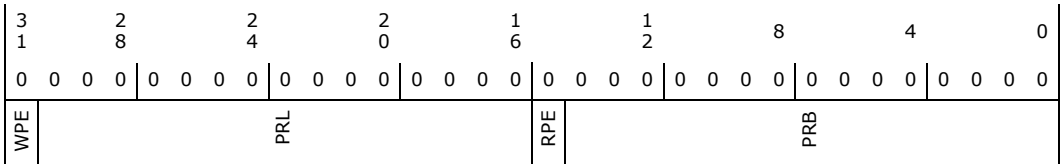
This register cannot be written when the PRR34_LOCKDN bit is set to 1.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RW/L	**Write Protection Enable (WPE):** Write Protection Enable: When set, this bit indicates that the Base and Limit fields in this register are valid and that writes and erases directed to addresses between them (inclusive) must be blocked by hardware. The base and limit fields are ignored when this bit is cleared.
30:16	0h RW/L	**Protected Range Limit (PRL):** Protected Range Limit: This field corresponds to FLA address bits 26:12 and specifies the upper limit of the protected range. Address bits 11:0 are assumed to be FFFh for the limit comparison. Any address greater than the value programmed in this field is unaffected by this protected range.
15	0h RW/L	**Read Protection Enable (RPE):** Read Protection Enable: When set, this bit indicates that the Base and Limit fields in this register are valid and that read directed to addresses between them (inclusive) must be blocked by hardware. The base and limit fields are ignored when this bit is cleared.
14:0	0h RW/L	**Protected Range Base (PRB):** Protected Range Base: This field corresponds to FLA address bits 26:12 and specifies the lower base of the protected range. Address bits 11:0 are assumed to be 000h for the base comparison. Any address less than the value programmed in this field is unaffected by this protected range.

27.9.38 Flash Protected Range 4 (BIOS_FPR4)—Offset 94h

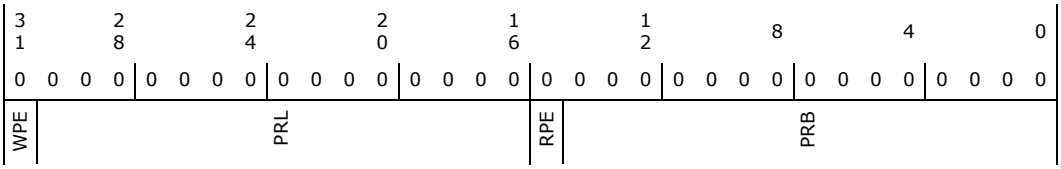
This register cannot be written when the PRR34_LOCKDN bit is set to 1.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RW/L	**Write Protection Enable (WPE):** Write Protection Enable: When set, this bit indicates that the Base and Limit fields in this register are valid and that writes and erases directed to addresses between them (inclusive) must be blocked by hardware. The base and limit fields are ignored when this bit is cleared.
30:16	0h RW/L	**Protected Range Limit (PRL):** Protected Range Limit: This field corresponds to FLA address bits [26:12] and specifies the upper limit of the protected range. Address bits [11:0] are assumed to be FFFh for the limit comparison. Any address greater than the value programmed in this field is unaffected by this protected range.
15	0h RW/L	**Read Protection Enable (RPE):** Read Protection Enable: When set, this bit indicates that the Base and Limit fields in this register are valid and that read directed to addresses between them (inclusive) must be blocked by hardware. The base and limit fields are ignored when this bit is cleared.
14:0	0h RW/L	**Protected Range Base (PRB):** Protected Range Base: This field corresponds to FLA address bits [26:12] and specifies the lower base of the protected range. Address bits 11:0 are assumed to be 000h for the base comparison. Any address less than the value programmed in this field is unaffected by this protected range.

27.9.39 Global Protected Range 0 (BIOS_GPR0)—Offset 98h

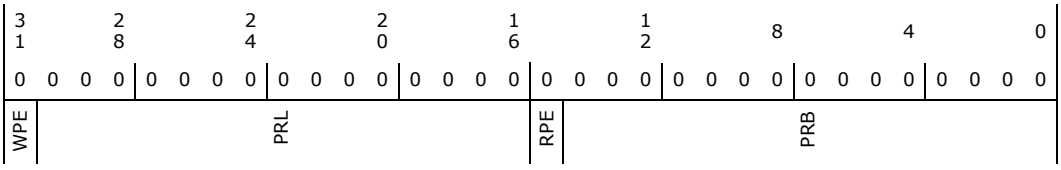
This register is initialized via softstraps. This protected range applies globally to all masters / flash requesters.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RO/V	**Write Protection Enable (WPE):** Write Protection Enable: When set, this bit indicates that the Base and Limit fields in this register are valid and that writes and erases directed to addresses between them (inclusive) must be blocked by hardware. The base and limit fields are ignored when this bit is cleared.
30:16	0h RO/V	**Protected Range Limit (PRL):** Protected Range Limit: This field corresponds to FLA address bits [26:12] and specifies the upper limit of the protected range. Address bits 11:0 are assumed to be FFFh for the limit comparison. Any address greater than the value programmed in this field is unaffected by this protected range.
15	0h RO/V	**Read Protection Enable (RPE):** Read Protection Enable: When set, this bit indicates that the Base and Limit fields in this register are valid and that read directed to addresses between them (inclusive) must be blocked by hardware. The base and limit fields are ignored when this bit is cleared.
14:0	0h RO/V	**Protected Range Base (PRB):** Protected Range Base: This field corresponds to FLA address bits [26:12] and specifies the lower base of the protected range. Address bits 11:0 are assumed to be 000h for the base comparison. Any address less than the value programmed in this field is unaffected by this protected range.

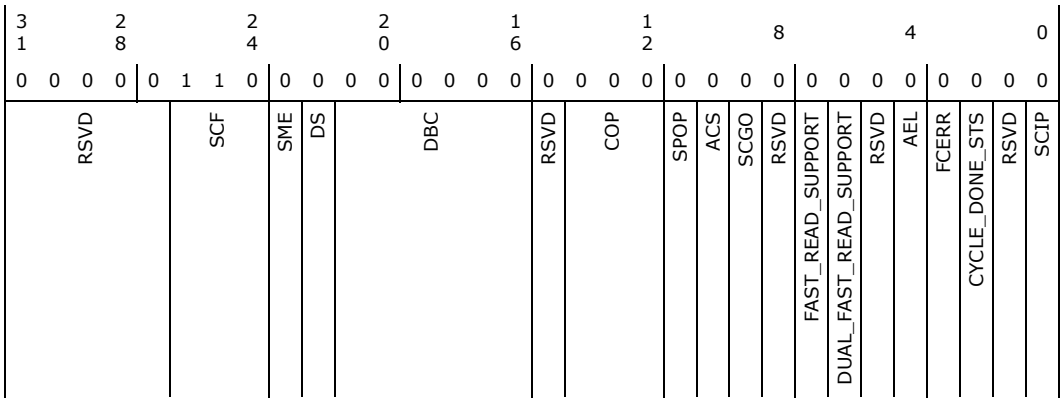
27.9.40 Software Sequencing Flash Status and Control (BIOS_SSFSTS_CTL)—Offset A0h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 6000000h



Bit Range	Default and Access	Field Name (ID): Description
31:27	0h RO	Reserved.
26:24	6h RW/L	**SPI Cycle Frequency (SCF):** SPI Cycle Frequency (SCF): The listed frequencies are approximate. See 31.11. 000 : 120 MHz (not supported in the Intel® C620 Series Chipset) 001 : 60 MHz (not supported in the Intel® C620 Series Chipset) 010 : 48 MHz not supported in the Intel® C620 Series Chipset 011 : 40 MHz 100 : 30 MHz 101 : 24 MHz (not supported in the Intel® C620 Series Chipset) 110 : 17 MHz All Others: Reserved This register sets frequency to use for all SPI Software Sequencing cycles (write, erase, fast read, read status, etc.) except for the Read cycle which always run at 20 MHz. This register is locked when the SPI Configuration Lock-Down (FLOCKDN) bit is set. ***Note:*** Fast Reset Test mode overrides the value programmed into this register.
23	0h RW	**SPI SMI# Enable (SME):** SPI SMI# Enable (SME): When set to 1, the SPI asserts an SMI# request whenever the Cycle Done Status bit is 1.
22	0h RW	**Data Cycle (DS):** Data Cycle (DS): When set to 1, there is data that corresponds to this transaction. When 0, no data is delivered for this cycle, and the DBC and data fields themselves are don t cares.
21:16	0h RW	**Data Byte Count (DBC):** Data Byte Count (DBC): This field specifies the number of bytes to shift in or out during the data portion of the SPI cycle. The valid settings (in decimal) are any value from 0 to 63. The number of bytes transferred is the value of this field plus 1. Note that when this field is 00_0000b, then there is 1 byte to transfer and that 11_1111b means there are 64 bytes to transfer.
15	0h RO	Reserved.
14:12	0h RW	**Cycle Opcode Pointer (COP):** Cycle Opcode Pointer (COP): This field selects one of the programmed opcodes in the Opcode Menu to be used as the SPI Command/ Opcode. In the case of an Atomic Cycle Sequence, this determines the second command.
11	0h RW	**Sequence Prefix Opcode Pointer (SPOP):** Sequence Prefix Opcode Pointer (SPOP): This field selects one of the two programmed prefix opcodes for use when performing an Atomic Cycle Sequence. A value of 0 points to the opcode in the least significant byte of the Prefix Opcodes register. By making this programmable, the ICH supports flash devices that have different opcodes for enabling writes to the data space vs. status register.
10	0h RW	**Atomic Cycle Sequence (ACS):** Atomic Cycle Sequence (ACS): When set to 1 along with the SCGO assertion, the PCH will execute a sequence of commands on the SPI interface without allowing another master to arbitrate and interleave cycles. The sequence is composed of: - Atomic Sequence Prefix Command (8-bit opcode only) - Primary Command specified below by software (can include address and data) - Polling the Flash Status Register (opcode 05h) until bit 0 becomes 0b. The SPI Cycle in Progress bit remains set and the Cycle Done Status bit remains unset until the Busy bit in the Flash Status Register returns 0.
9	0h RW/1S/V	**SPI Cycle Go (SCGO):** SPI Cycle Go (SCGO): This bit always returns 0 on reads. However, a write to this register with a '1' in this bit starts the SPI cycle defined by the other bits of this register. The [quote]SPI Cycle in Progress[/quote] (SCIP) bit gets set by this action. Hardware must ignore writes to this bit while the Cycle In Progress bit is set. Hardware allows other bits in this register to be programmed for the same transaction when writing this bit to 1. This saves an additional memory write.
8	0h RO	Reserved.
7	0h RO/V	**Fast Read Supported (FAST_READ_SUPPORT):** Fast Read Supported: This bit reflects the value of the Fast Read Support bit in the Flash Descriptor Component Section.
6	0h RO/V	**Dual Output Fast Read Supported (DUAL_FAST_READ_SUPPORT):** Dual Output Fast Read Supported: This bit reflects the value of the Dual Output Fast Read Support bit Anthe Flash Descriptor Component Section.
5	0h RO	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
4	0h RO/V	**Access Error Log (AEL):** Access Error Log (AEL): This bit reflects the value of the Hardware Sequencing Status.AEL register.
3	0h RW/1C/V	**Flash Cycle Error (FCERR):** Flash Cycle Error (FCERR): Hardware sets this bit to 1 when a programmed access is blocked from running on the SPI interface due to one of the protection policies or when any of the programmed cycle registers is written while a programmed access is already in progress. This bit remains asserted until cleared by software writing a 1 or hardware reset.
2	0h RW/1C/V	**Cycle Done Status (CYCLE_DONE_STS):** Cycle Done Status: The PCH sets this bit to 1 when the SPI Cycle completes (i.e., SCIP bit is 0) after software sets the GO bit. This bit remains asserted until cleared by software writing a 1 or hardware reset. When this bit is set and the SPI SMI# Enable bit is set, an internal signal is asserted to the SMI# generation block. Software must make sure this bit is cleared prior to enabling the SPI SMI# assertion for a new programmed access.
1	0h RO	Reserved.
0	0h RO/V	**SPI Cycle In Progress (SCIP):** SPI Cycle In Progress (SCIP): Hardware sets this bit when software sets the SPI Cycle Go bit in the Command register. This bit remains set until the cycle completes on the SPI interface. Hardware automatically sets and clears this bit so that software can determine when read data is valid and/or when it is safe to begin programming the next command. Software must only program the next command when this bit is 0.

27.9.41 Prefix Opcode and Opcode Type Configuration (BIOS_PREOP_OPTYPE)—Offset A4h

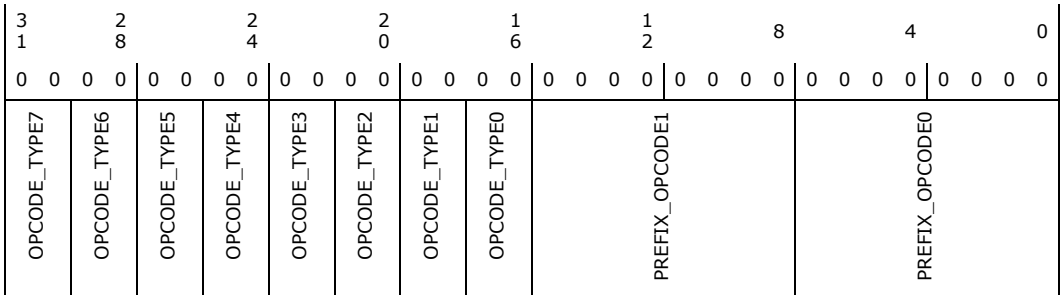
This register is not writable when the Flash Configuration Lock-Down (FLOCKDN) bit is set. Entries in this register correspond to the entries in the Opcode Menu Configuration register.
Note that the definition below only provides write protection for opcodes that have addresses associated with them. Therefore, any erase or write opcodes that do not use an address should be avoided.

Access Method

Type: MEM Register (Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:30	0h RW/L	**Opcode Type 7 (OPCODE_TYPE7):** Same description as OPCODE_TYPE0.
29:28	0h RW/L	**Opcode Type 6 (OPCODE_TYPE6):** Same description as OPCODE_TYPE0.
27:26	0h RW/L	**Opcode Type 5 (OPCODE_TYPE5):** Same description as OPCODE_TYPE0.
25:24	0h RW/L	**Opcode Type 4 (OPCODE_TYPE4):** Same description as OPCODE_TYPE0.
23:22	0h RW/L	**Opcode Type 3 (OPCODE_TYPE3):** Same description as OPCODE_TYPE0.
21:20	0h RW/L	**Opcode Type 2 (OPCODE_TYPE2):** Same description as OPCODE_TYPE0.
19:18	0h RW/L	**Opcode Type 1 (OPCODE_TYPE1):** Same description as OPCODE_TYPE0.
17:16	0h RW/L	**Opcode Type 0 (OPCODE_TYPE0):** This field specifies information about the corresponding Opcode 0. This information allows the hardware to 1) know whether to use the address field and 2) provide BIOS and Shared Flash protection capabilities. The hardware implementation also uses the read vs. write information for modifying the behavior of the SPI interface logic. The encoding of the two bits is: 00 = No Address associated with this Opcode and Read Cycle type 01 = No Address associated with this Opcode and Write Cycle type 10 = Address required; Read cycle type 11 = Address required; Write cycle type
15:8	0h RW/L	**Prefix Opcode 1 (PREFIX_OPCODE1):** Software programs an SPI opcode into this field that is permitted to run as the first command in an atomic cycle sequence.
7:0	0h RW/L	**Prefix Opcode 0 (PREFIX_OPCODE0):** Software programs an SPI opcode into this field that is permitted to run as the first command in an atomic cycle sequence.

27.9.42 Opcode Menu0 Configuration (BIOS_OPMENU0)—Offset A8h

This register is not writable when the SPI Configuration Lock-Down bit (FLOCKDN) is set. Eight entries are available in this register to give BIOS a sufficient set of commands for communicating with the flash device, while also restricting what malicious software can do. This keeps the hardware flexible enough to operate with a wide variety of SPI devices.

It is recommended that BIOS avoid programming Write Enable opcodes in this menu. Malicious software could then perform writes and erases to the SPI flash without using the atomic cycle mechanism. This could cause functional failures in a shared flash environment.

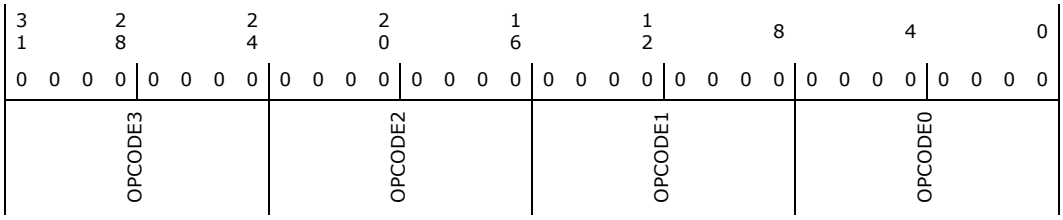
Write Enable opcodes should only be programmed in the Prefix Opcodes.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RW/L	**Allowable Opcode 3 (OPCODE3):** See the description for bits 7:0
23:16	0h RW/L	**Allowable Opcode 2 (OPCODE2):** See the description for bits 7:0
15:8	0h RW/L	**Allowable Opcode 1 (OPCODE1):** See the description for bits 7:0
7:0	0h RW/L	**Allowable Opcode 0 (OPCODE0):** Software programs an SPI opcode into this field for use when initiating SPI commands through the Control Register.

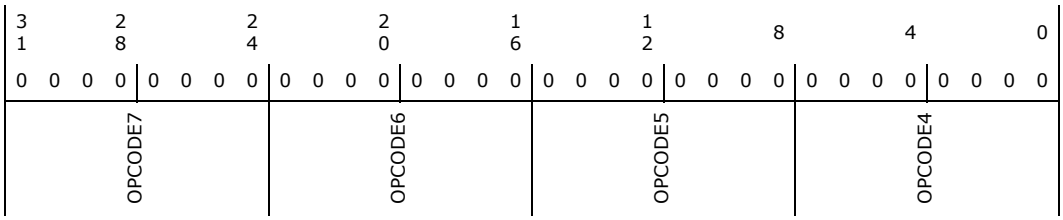
27.9.43 Opcode Menu1 Configuration (BIOS_OPMENU1)—Offset ACh

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RW/L	**Allowable Opcode 7 (OPCODE7):** See the description for bits 7:0 in BIOS_OPMENU0 Register.
23:16	0h RW/L	**Allowable Opcode 6 (OPCODE6):** See the description for bits 7:0 in BIOS_OPMENU0 Register.
15:8	0h RW/L	**Allowable Opcode 5 (OPCODE5):** See the description for bits 7:0 in BIOS_OPMENU0 Register.
7:0	0h RW/L	**Allowable Opcode 4 (OPCODE4):** See the description for bits 7:0 in BIOS_OPMENU0 Register.

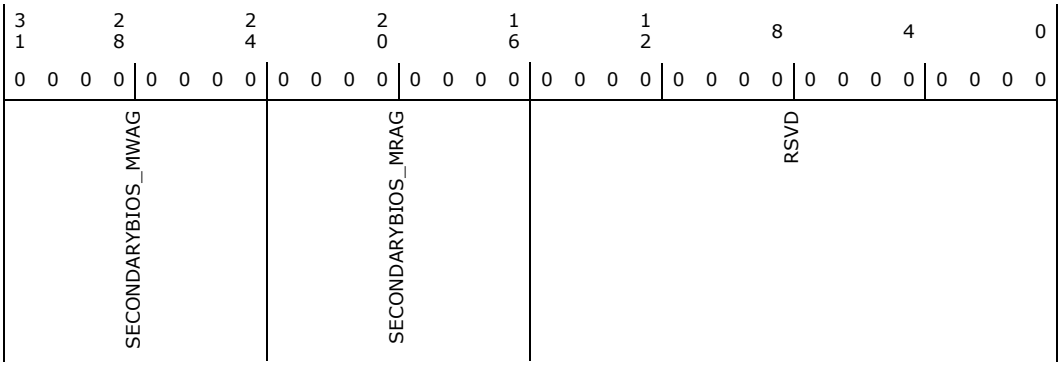
27.9.44 Secondary Flash Region Access Permissions (BIOS_SFRACC)—Offset B0h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RW/L	**Secondary BIOS Master Write Access Grant (SECONDARYBIOS_MWAG):** Each bit 31:29 corresponds to Master7:0. BIOS can grant one or more masters write access to the Secondary BIOS region 6 overriding the permissions in the Flash Descriptor. Bits for unassigned masters are reserved. The contents of this register are locked by the FLOCKDN bit.
23:16	0h RW/L	**Secondary BIOS Master Read Access Grant (SECONDARYBIOS_MRAG):** Each bit 28:16 corresponds to Master7:0. BIOS can grant one or more masters read access to the Secondary BIOS region 6 overriding the read permissions in the Flash Descriptor. Bits for unassigned masters are reserved. The contents of this register are locked by the FLOCKDN bit.
15:0	0h RO	Reserved.

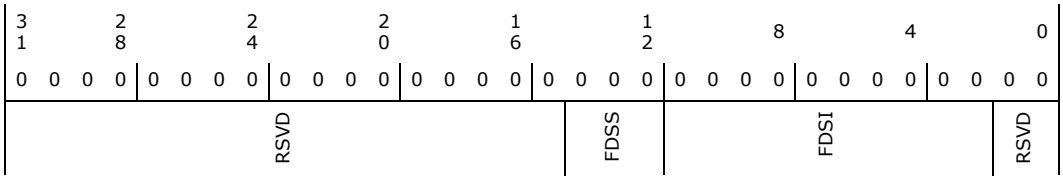
27.9.45 Flash Descriptor Observability Control (BIOS_FDOC)—Offset B4h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:15	0h RO	Reserved.
14:12	0h RW	**Flash Descriptor Section Select (FDSS):** Selects which section within the loaded Flash Descriptor to observe. 000: Flash Signature and Descriptor Map 001: Component 010: Region 011: Master Others: Reserved
11:2	0h RW	**Flash Descriptor Section Index (FDSI):** Selects the DW offset within the Flash Descriptor Section to observe.
1:0	0h RO	Reserved.

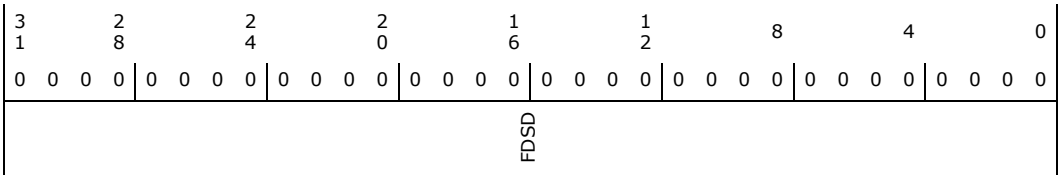
27.9.46 Flash Descriptor Observability Data (BIOS_FDOD)—Offset B8h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RO/V	**Flash Descriptor Section Data (FDSD):** Returns the DW of data to observe as selected in the Flash Descriptor Observability Control.

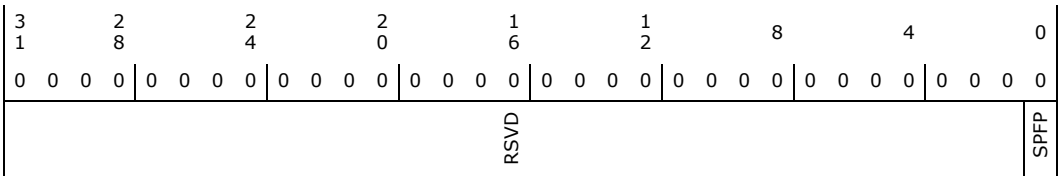
27.9.47 Additional Flash Control (BIOS_AFC)—Offset C0h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h RW/V/P	**Stop Prefetch on Flush Pending (SPFP):** When set to 1, the in progress of a prefetch will be ended if subsequence access from the master of the same interface is detected to be a cache-miss and read cache will be flushed. When set to 0, the prefetch will be allowed to complete prior to flushing.

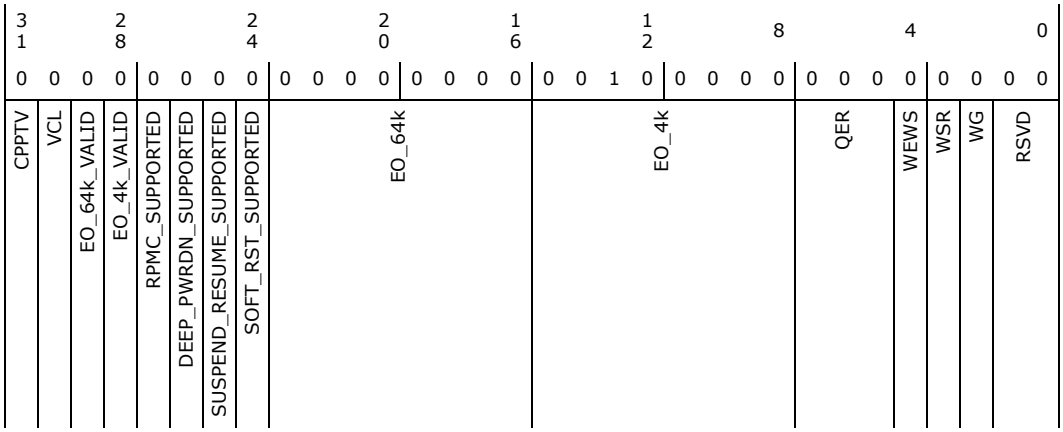
27.9.48 Vendor Specific Component Capabilities for Component 0 (BIOS_SFDP0_VSCC0)—Offset C4h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 2000h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RO/V	**Component Property Parameter Table Valid (CPPTV):** This bit is set to a 1 if the Flash Controller detects a valid SFDP Component Property Parameter Table in Component 0. ***Note:*** If this bit is set software must not overwrite bits that were initialized by hardware via SFDP discovery.
30	0h RW/L	**Vendor Component Lock (VCL):** 0: The lock bit is not set 1: The Vendor Component Lock bit is set. This register locks itself when set.
29	0h RW/V/L	**64k Erase Valid (EO_64k_VALID):** 0: The EO_64k opcode is not valid. 1: The EO_64k opcode is valid.
28	0h RW/V/L	**4k Erase Valid (EO_4k_VALID):** 0: The EO_4k opcode is not valid. 1: The EO_4k opcode is valid.
27	0h RW/L	**RPMC Supported (RPMC_SUPPORTED):** 0 the device does not support RPMC 1 the device supports RPMC
26	0h RW/V/L	**Deep Power Down Supported (DEEP_PWRDN_SUPPORTED):** 0 the device does not support Deep Power down. 1 the device supports Deep Power down.
25	0h RW/V/L	**Suspend/Resume Supported (SUSPEND_RESUME_SUPPORTED):** 0 the device does not support Suspend/Resume 1 the device supports Suspend/Resume
24	0h RW/V/L	**Soft Reset Supported (SOFT_RST_SUPPORTED):** 0 the device does not support Soft Reset 1 the device supports Soft Reset
23:16	0h RW/V/L	**64k Erase Opcode (EO_64k):** This register is programmed with the Flash 64k sector erase instruction opcode for component 0. This register is locked by the Vendor Component Lock (VCL) bit or the CPPTV bit.

Bit Range	Default and Access	Field Name (ID): Description
15:8	20h RW/V/L	**4k Erase Opcode (EO_4k):** This register is programmed with the Flash 64k sector erase instruction opcode for component 0. This register is locked by the Vendor Component Lock (VCL) bit or the CPPTV bit.
7:5	0h RW/V/L	**Quad Enable Requirements (QER):** 000 = Part does not require a Quad Enable bit to be set, either because Quad is not supported or because the manufacturer somehow permanently enables Quad capability. 001 = Part requires bit 9 in status register 2 to be set to enable quad IO. Writing one byte to status register clears all bits in register 2, therefore status register writes MUST be two bytes. 010 = Part requires bit 6 of status register 1 to be set to enable quad IO. 011 = Part requires bit 7 of the configuration register to be set to enable Quad. 100 = Part requires bit 9 in status register 2 to be set to enable quad IO. Writing one byte to the status register does not clear the second byte. This register is locked by the Vendor Component Lock (VCL) bit. If the SFDP table contains this information, the flash controller loads these bits from the table. The flash controller uses this information to prevent clearing the QE bit in the flash device's status register when WSR=1.
4	0h RW/V/L	**Write Enable on Write Status (WEWS):** 0 = 50h is the opcode to enable a status register write 1 : 06h is the opcode to enable a status register write This register is locked by the Vendor Component Lock (VCL) bit or the CPPTV bit. ***Note:*** Hardware ignores the state of this bit.
3	0h RW/V/L	**Write Status Required (WSR):** 0 = No requirement to write to the Write Status Register prior to a write 1 = A write is required to the Write Status Register prior to write and erase to remove any protection. This is required for SST components. This register is locked by the Vendor Component Lock (VCL) bit or the CPPTV bit. ***Note:*** Hardware ignores the state of this bit.
2	0h RW/V/L	**Write Granularity (WG):** 0: Reserved 1: 64 Byte This register is locked by the Vendor Component Lock (VCL) bit or the CPPTV bit. ***Note:*** Hardware ignores the state of this bit.
1:0	0h RO	Reserved.

27.9.49 Vendor Specific Component Capabilities for Component 1 (BIOS_SFDP1_VSCC1)—Offset C8h

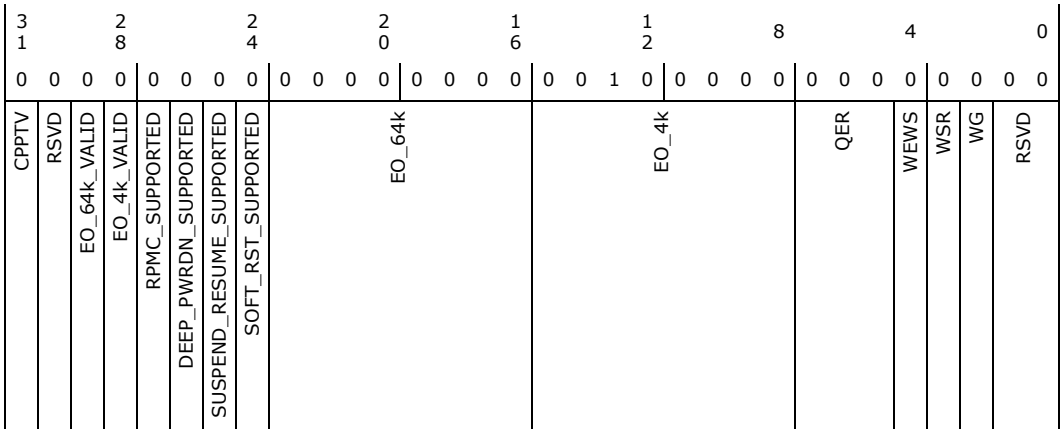
This register pertain to cycles targeting addresses outside of Component 0. The lockable bits in this register are locked when either CPPTV is set to 1 by hardware or when VCL is 1.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 2000h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RO/V	**Component Property Parameter Table Valid (CPPTV):** This bit is set to a 1 if the Flash Controller detects a valid SFDP Component Property Parameter Table in Component 1. ***Note:*** If this bit is set software must not overwrite bits that were initialized by hardware via SFDP discovery RO
30	0h RO	Reserved.
29	0h RW/V/L	**64k Erase Valid (EO_64k_VALID):** 0: The EO_64k opcode is not valid. 1: The EO_64k opcode is valid.
28	0h RW/V/L	**4k Erase Valid (EO_4k_VALID):** 0: The EO_4k opcode is not valid. 1: The EO_4k opcode is valid.
27	0h RW/L	**RPMC Supported (RPMC_SUPPORTED):** 0 the device does not support RPMC 1 the device supports RPMC
26	0h RW/V/L	**Deep Power Down Supported (DEEP_PWRDN_SUPPORTED):** 0 the device does not support Deep Power Down 1 the device supports Deep Power Down
25	0h RW/V/L	**Suspend/Resume Supported (SUSPEND_RESUME_SUPPORTED):** 1 the device supports Suspend/Resume
24	0h RW/V/L	**Soft Reset Supported (SOFT_RST_SUPPORTED):** 0 the device does not support Soft Reset 1 the device supports Soft Reset
23:16	0h RW/V/L	**64k Erase Opcode (EO_64k):** This register is programmed with the Flash 64k sector erase instruction opcode for component 1. This register is locked by the Vendor Component Lock (VCL) bit or the CPPTV bit.
15:8	20h RW/V/L	**4k Erase Opcode (EO_4k):** This register is programmed with the Flash 4k subsector erase instruction opcode for component 1. Software must program this register if the SFDP table for this component does not show 4 KB erase capability. This register is locked by the Vendor Component Lock (VCL) bit or the CPPTV bit.

Bit Range	Default and Access	Field Name (ID): Description
7:5	0h RW/V/L	**Quad Enable Requirements (QER):** 000 = Part does not require a Quad Enable bit to be set, either because Quad is not supported or because the manufacturer somehow permanently enables Quad capability. 001 = Part requires bit 9 in status register 2 to be set to enable quad IO. Writing one byte to status register clears all bits in register 2, therefore status register writes MUST be two bytes. 010 = Part requires bit 6 of status register 1 to be set to enable quad IO. 011 = Part requires bit 7 of the configuration register to be set to enable Quad. 100 = Part requires bit 9 in status register 2 to be set to enable quad IO. Writing one byte to the status register does not clear the second byte. This register is locked by the Vendor Component Lock (VCL) bit. If the SFDP table contains this information, the flash controller loads these bits from the table. The flash controller uses this information to prevent clearing the QE bit in the flash device's status register when WSR=1.
4	0h RW/V/L	**Write Enable on Write Status (WEWS):** 0: 50h is the opcode to enable a status register write 1: 06h is the opcode to enable a status register write This register is locked by the Vendor Component Lock (VCL) bit or the CPPTV bit.
3	0h RW/V/L	**Write Status Required (WSR):** 0: No requirement to write to the Write Status Register prior to a write 1: A write is required to the Write Status Register prior to write and erase to remove any protection. This is required for SST components. This register is locked by the Vendor Component Lock (VCL) bit or the CPPTV bit.
2	0h RW/V/L	**Write Granularity (WG):** 0: Reserved 1: 64 Byte This register is locked by the Vendor Component Lock (VCL) bit or the CPPTV bit. ***Note:*** Hardware ignores the state of this bit.
1:0	0h RO	Reserved.

27.9.50 Parameter Table Index (BIOS_PTINX)—Offset CCh

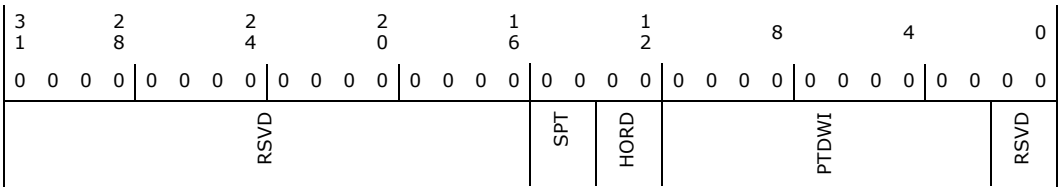
Observability control for Component Property Tables

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:14	0h RW	**Supported Parameter Table (SPT):** Selects which supported parameter table to observe. 00 : Component 0 Property Parameter Table 01 : Component 1 Property Parameter Table 10 - 11 : Reserved
13:12	0h RW	**Header or Data (HORD):** Select parameter table header DW vs. Data DW. 00 : SFDP Header 01 : Parameter Table Header 10 : Data 11 : Reserved
11:2	0h RW	**Parameter Table DW Index (PTDWI):** Selects the DW offset within the parameter table to observe.
1:0	0h RO	Reserved.

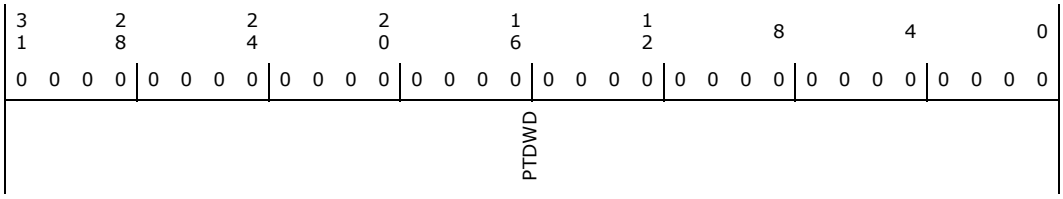
27.9.51 Parameter Table Data (BIOS_PTDATA)—Offset D0h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RO/V	**Parameter Table DW Data (PTDWD):** Returns the DW of data to observe as selected in the Parameter Table Index register. ***Note:*** The returned value of reserved fields in the SFDP header or table must be ignored by software. The flash controller may return either 0 or 1 for these fields.

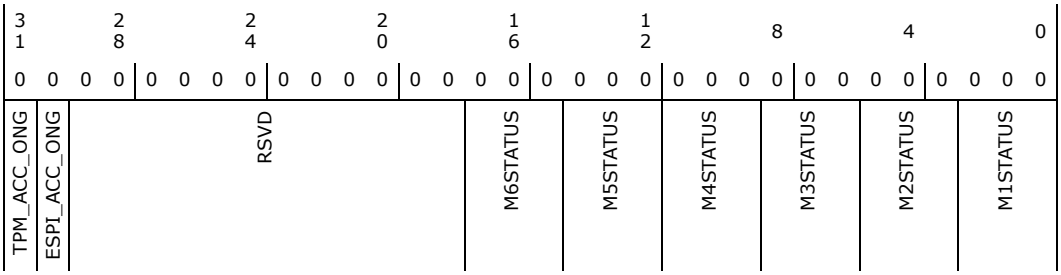
27.9.52 SPI Bus Requester Status (BIOS_SBRS)—Offset D4h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RO/V	**TPM access ongoing (TPM_ACC_ONG)**
30	0h RO/V	**eSPI access ongoing (ESPI_ACC_ONG)**
29:18	0h RO	Reserved.
17:15	0h RO/V	**Master 6 Status (M6STATUS):** Master 6 Status (M6STATUS): See description under M1STATUS
14:12	0h RO/V	**Master 5 Status (M5STATUS):** Master 5 Status (M5STATUS): See description under M1STATUS
11:9	0h RO/V	**Master 4 Status (M4STATUS):** Master 4 Status (M4STATUS): See description under M1STATUS
8:6	0h RO/V	**Master 3 Status (M3STATUS):** Master 3 Status (M3STATUS): See description under M1STATUS
5:3	0h RO/V	**Master 2 Status (M2STATUS):** Master 2 Status (M2STATUS): See description under M1STATUS
2:0	0h RO/V	**Master 1 Status (M1STATUS):** Indicates whether this master has an outstanding transaction enqueued or in flight and the transaction type. 0xx : no transaction 100 : flash read transaction 101 : flash write transaction 110 : flash erase transaction 111 : flash RPMC transaction

27.10 BIOS Flash Program Registers Summary

Table 27-6. Summary of BIOS Flash Program Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
F0h	F3h	Set Strap Msg Lock (SSML)—Offset F0h	0h
F4h	F7h	Set Strap Msg Control (SSMC)—Offset F4h	0h
F8h	FBh	Set Strap Msg Data (SSMD)—Offset F8h	0h

27.10.1 Set Strap Msg Lock (SSML)—Offset F0h

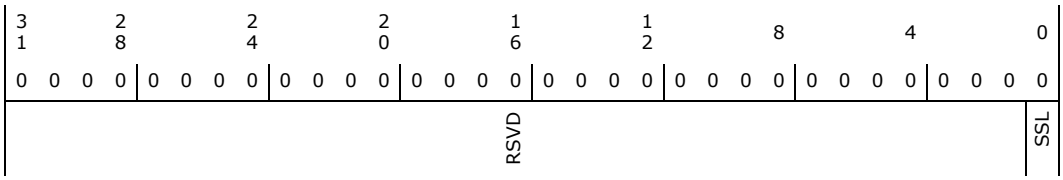
Lockable: Yes
Power Well: Resume
reset_type= wrsmrst#

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h RW/L	**Set_Strap Lock (SSL):** When set to '1', all of SSML, SSMC and SSMD is locked, including this Lock bit. Note that this bit is reset on CF9 resets.

27.10.2 Set Strap Msg Control (SSMC)—Offset F4h

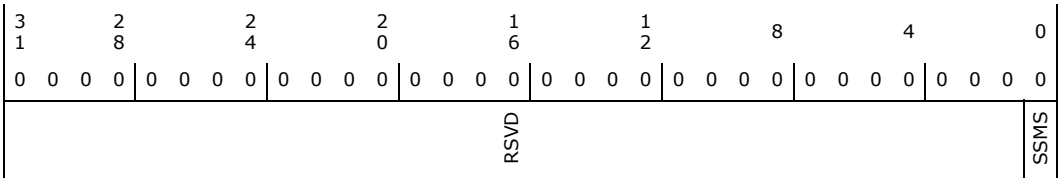
Lockable: Yes
Power Well: DSW
reset_type= dsw_pok
The DSW bits are all cleared by dsw_pok, but must not be cleared by CF9h resets.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h RW/L	**Set_Strap Mux Select (SSMS):** When set to '1', the Set strap message bits [47:32] come from the Set_Strap Msg Data register. This bit is reset by the RSMRST# pin only. When '0', the Set-Strap data continues to come from the soft straps themselves. This register field is locked by the Set Strap Lock (SSML.SSL) bit.

27.10.3 Set Strap Msg Data (SSMD)—Offset F8h

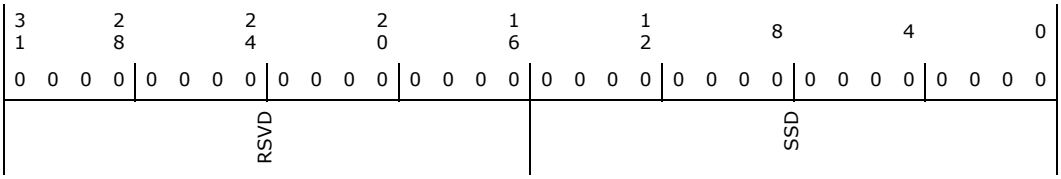
Lockable: Yes
Power Well: DSW
Reset Type: dsw_pok
This register is used to provide a BIOS programmable sticky register which contains data that will be used in the Set-Strap type 1 msg on subsequent resets. These bits are in the resume well, so only reset on G3.
The usage model is that on each reset BIOS will check the state of the CPU. If the state is correct then BIOS continues. If not, then BIOS writes the SSMD and SSMC registers and does a CF9 reset.
On the reset the value of what was written to SSMD takes effect.
Note that some mobile platforms force G3 on S5 requests. For those platforms, if the user/BIOS wants to have these bits set, there will be 2 resets on every power-on. If the platform accepts the default of '0' for these controls, then there is only one reset.
The bits are in DSW and not RTC well because this allows a user upgrade, assuming the user unplugged the system before doing the upgrade, to revert to a setting of 0. This should reduce any interoperability concerns regarding user upgrades.
The DSW bits are all cleared dsw_pok, and must not be cleared by CF9h resets.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 31
Function: 5

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:0	0h RW/L	**Set_Strap Data (SSD):** When SSMS is '1', then this data is sent in the Set-Strap msg Type 1 upon reset. This data is sent in the 2nd DW of data, bits [15:0]. This register field is locked by the Set Strap Lock (SSML.SSL) bit.

§

28 Testability

28.1 Acronyms

Acronyms	Description
BSDL	Boundary Scan Description Language
IEEE	Institute of Electrical and Electronics Engineers
I/O	Input/Output
I/OD	Input/Output Open Drain
JTAG	Joint Test Action Group
TAP	Test Access Port

28.2 References

Specification	Location
IEEE Standard Test Access Port and Boundary Scan Architecture*	http://standards.ieee.org/findstds/standard/1149.1-2013.html

28.3 Overview

This section contains information regarding the PCH testability signals that provides access to JTAG, run control, system control, and observation resources. PCH JTAG TAP ports are compatible with the IEEE Standard Test Access Port and Boundary Scan Architecture 1149.1 and 1149.6 Specification, as detailed per device in each BSDL file. JTAG Pin definitions are from IEEE Standard Test Access Port and Boundary-Scan. Architecture (IEEE Std. 1149.1-2001).

28.4 Signal Description

Table 28-1. Testability Signals (Sheet 1 of 2)

Name	Type	Description
JTAG_TCK	I/O	**Test Clock Input (TCK):** The test clock input provides the clock for the JTAG test logic.
JTAG_TMS	I/OD	**Test Mode Select (TMS):** The signal is decoded by the Test Access Port (TAP) controller to control test operations.
JTAG_TDI	I/OD	**Test Data Input (TDI):** Serial test instructions and data are received by the test logic at TDI.
JTAG_TDO	I/OD	**Test Data Output (TDO):** TDO is the serial output for test instructions and data from the test logic defined in this standard.
JTAGx	I/O	This pin is used to support merged debug port topologies.
ITP_PMODE	O	This signal is used to transmit CPU and PCH power/reset information to the ITP Debugger.
TRIGGER_IN	I	From the CPU, for cross die triggering for debug trace
TRIGGER_OUT	O	To the CPU IOT for cross die triggering
PRDY_N	I/OD	Acknowledge from CPU for Run control.
PREQ_N	I/OD	From the PCH to the CPU for run control by DCI for closed chassis testing.

Table 28-1. Testability Signals (Sheet 2 of 2)

Name	Type	Description
CPU_TRST_N	O	JTAG output from DCI to CPU
POWER_DEBUG_N	OD	Connect to PWR_DEBUG_N pin on CPU.

28.5 I/O Signal Planes and States

Table 28-2. Power Planes and States for PCH Testability Signals

Signal Name	Power Plane	During Reset	Immediately after Reset	S3/S4/S5	Deep Sx
JTAG_TCK	Primary	Internal PD	Internal PD	Internal PD	Off
JTAG_TMS	Primary	Internal PU	Internal PU	Internal PU	Off
JTAG_TDI	Primary	Internal PU	Internal PU	Internal PU	Off
JTAG_TDO	Primary	Undriven	Undriven	Undriven	Off
JTAGx[1]	Primary	Internal PU (as TDO)/Internal PD (as TCK)	Internal PU/ Internal PD	Internal PU/ Internal PD	Off
ITP_PMODE	Primary	Internal PU	Internal PU	Internal PU	Off
TRIGGER_IN	Primary	Internal PD	Internal PD	Undriven	Off
TRIGGER_OUT	Primary	Internal PD	Internal PD	Undriven	Off
PRDY_N	Primary	External PU	External PU	Undriven	Off
PREQ_N	Primary	Internal PU	Internal PU	Undriven	Off
CPU_TRST_N	Primary	Internal PD	Internal PD	Internal PD	Off

Notes: This signal is used in common JTAG topology to take in last device's TDO to DCI. The only planned supported topology is the Shared Topology, so this pin will operate as TCK mode.
Notes: Internal pull-ups and pull-downs will not be valid until all the voltages are valid.

28.6 Intel® Trace Hub (Intel® TH)

28.6.1 Overview

Intel Trace Hub is a debug architecture that unifies hardware and software system visibility. Intel Trace Hub is not merely intended for hardware debug or software debug, but full system debug. This includes debugging hardware and software as they interact and produce complex system behavior. Intel Trace Hub defines new features and also leverages some existing debug technologies to provide a complete framework for hardware and software co-debug, software development and tuning, as well as overall system performance optimization.

Intel Trace Hub is a set of silicon features with supported software API. The primary purpose is to collect trace data from different sources in the system and combine them into a single output stream with time-correlated to each other. Intel Trace Hub uses common hardware interface for collecting time-correlated system traces through standard destinations. Intel Trace Hub adopts industry standard (MIPI* STPv2) debug methodology for system debug and software development.

There are multiple destinations to receive the trace data from Intel Trace Hub:

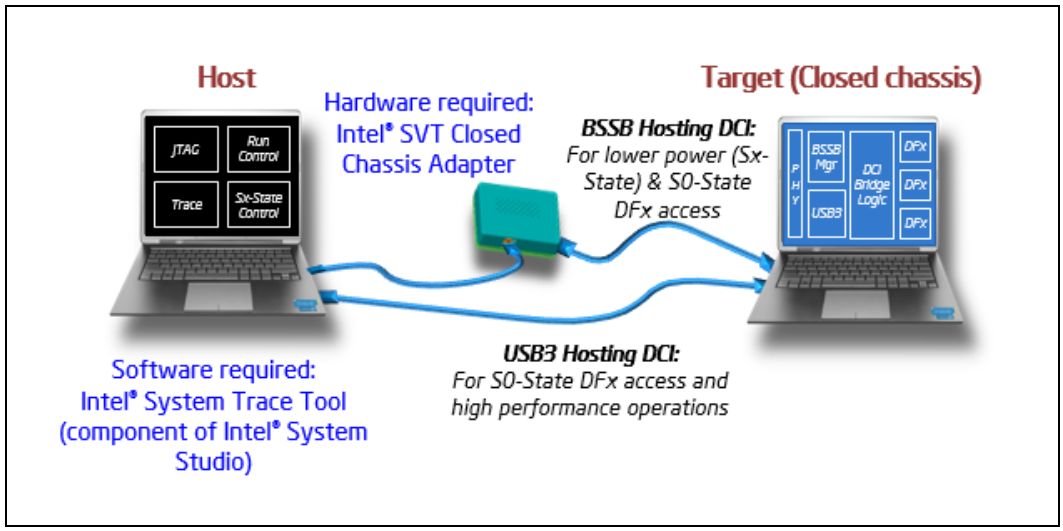
- Direct Connect Interface (DCI)
    - BSSB Hosting DCI
    - USB3 Hosting DCI
- System Memory

There are multiple trace sources planned to be supported:

- BIOS
- Intel ME
- AET (Architecture Event Trace)
- PCH Power Management Event Trace
- IE (IE FS add source debug message)
- Windows* ETW (for driver or application)

28.6.2 Platform Setup

Figure 28-1. Platform Setup with Intel Trace Hub



28.7 Direct Connect Interface (DCI)

Direct Connect Interface (DCI) is a new debug transport technology to enable closed chassis debug through any of USB3 ports out from Intel silicon. Some bridging logic is embedded in the silicon to "bridge" the gap between standard IO ports and the debug interfaces including JTAG, probe mode, hooks, trace infrastructure, and etc. To control the operation of this embedded logic, a DCI packet based protocol is invented which controls and data can be sent or received. This protocol can operate over a few different physical transport paths to the target which known as "hosting interfaces".

There are two types of DCI hosting interfaces:

- BSSB Hosting DCI
- USB3 Hosting DCI

Supported capabilities in DCI are:

- Closed Chassis Debug at S0 and Sx State

- JTAG Access and Run Control (Probe Mode)
- System Tracing with Intel Trace Hub

Debug host software that support DCI are:

- Intel® In-Target Probe (Intel® ITP) II Platform Debug Toolkit (PDT)
- Intel® System Studio (ISS)

28.7.1 BSSB Hosting DCI

BSSB stands for Boundary Scan Side Band. It was developed to provide an alternate path to convey controls and data to or from the embedded logic by connecting physically to the target through a USB3 port. BSSB provides an alternate side band path around the USB3 controller, so that the embedded logic can be accessed, even when the USB controller is not alive (such as in low power states), or is malfunctioning. This path does not rely on USB protocol, link layer, or physical layer, because the xHCI functions are generally not available in such conditions. Instead, this path relies on a special adapter that developed by Intel called Intel® Silicon View Technology (Intel® SVT) Closed Chassis Adapter (CCA). It is a simple data transformation device. This adapter generates a BSSB signaling protocol operating at up to 400 MHz and serializes data flowing through it. This adapter works together with debug host software and the embedded logic, contain a back-pressure scheme that makes both sides tolerant of overflow and starvation conditions, which is the moral equivalent of the USB link layer. This path also uses native DCI packet protocol instead of USB protocol.

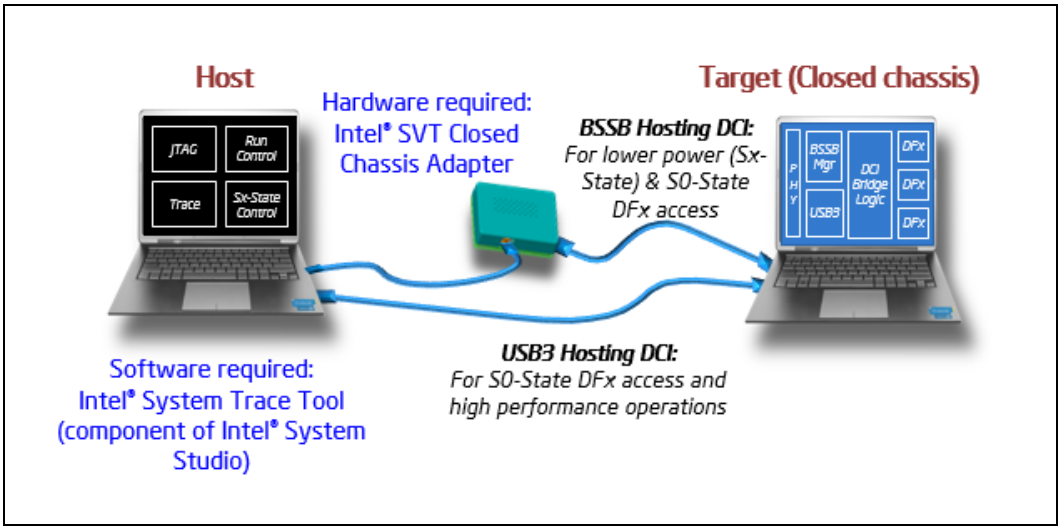
Intel SVT CCA (MM#:921521) can be purchased through Intel® Design-In Tools Store at https://designintools.intel.com/product_p/itpxdpsvt.htm.

28.7.2 USB3 Hosting DCI

It relies on Debug Class Devices (DbC) which is comprised of a set of logic that is bolted to the side of the xHCI host controller and enable the target to act the role of a USB device for debug purpose. This path uses the USB packet protocol layer, USB link layer flow control and USB3 physical layer at 5 GHz.

28.7.3 Platform Setup

Figure 28-2. Platform Setup with DCI Connection



28.8 Intel Trace Huh Configuration Registers Summary

Table 28-3. Summary of Intel Trace Huh Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	Vendor and Device Identification (VID)—Offset 0h	X_8086h
4h	7h	Command and Status Register (CMD)—Offset 4h	100000h
8h	Bh	Revision ID (RID)—Offset 8h	1300h
Ch	Fh	Header Type (HT)—Offset Ch	0h
10h	13h	MTB Low BAR (MTB_LBAR)—Offset 10h	4h
14h	17h	MTB Upper BAR (MTB_UBAR)—Offset 14h	0h
18h	1Bh	SW Low BAR (SW_LBAR)—Offset 18h	4h
1Ch	1Fh	SW Upper BAR (SW_UBAR)—Offset 1Ch	0h
20h	23h	RTIT Low BAR (RTIT_LBAR)—Offset 20h	4h
24h	27h	RTIT Upper BAR (RTIT_UBAR)—Offset 24h	0h
34h	37h	Capabilities Pointer (CAP)—Offset 34h	40h
3Ch	3Fh	Interrupt Line and Interrupt Pin (INTL)—Offset 3Ch	1FFh
40h	43h	MSI Capability (MSICID)—Offset 40h	2800005h
44h	47h	MSI Lower Message Address (MSILMA)—Offset 44h	0h
48h	4Bh	MSI Upper Message Address (MSIUMA)—Offset 48h	0h
4Ch	4Fh	MSI Message Data (MSIMD)—Offset 4Ch	0h
80h	83h	Device Specific Control and Device Specific Status (NPKDSC)—Offset 80h	1h

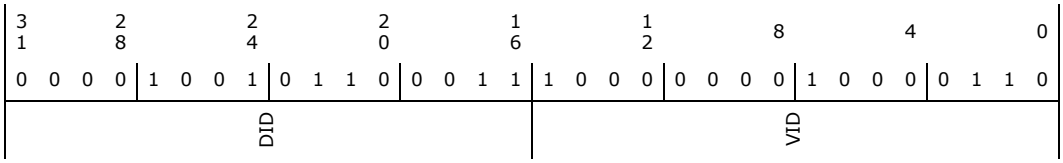
28.8.1 Vendor and Device Identification (VID)—Offset 0h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 7

Default: 9638086h



Bit Range	Default and Access	Field Name (ID): Description
31:16	-- RO	**Device ID (DID):** The value that uniquely identifies the Intel® Trace Hub from all other PCI devices. See the Device and Version ID Table in Volume 1 of the datasheet for the default value.
15:0	8086h RO	**Vendor ID (VID):** 8086 is Intel Vendor Identification code

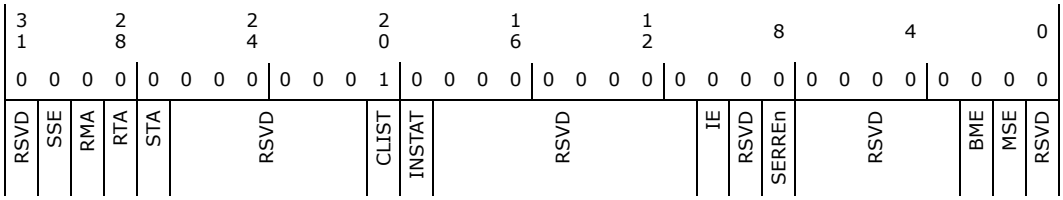
28.8.2 Command and Status Register (CMD)—Offset 4h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 7

Default: 100000h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RO	Reserved.
30	0h RW/1C	**SSE:** This bit is set when the device has detected an un-correctable error and reported it via SERR message over sideband. This requires SERR Enable bit to be set in Command register.
29	0h RW/1C	**RMA:** This bit is set when device receives a Completion transaction with Unsupported Request completion status. No error will be reported
28	0h RW/1C	**RTA:** This bit is set when device receives a Completion transaction with Completer Abort completion status. No error will be reported

Bit Range	Default and Access	Field Name (ID): Description
27	0h RW/1C	**STA:** Set by the device when aborting a request that violates the device programming model. When SERR Enable is set SERR message will be send over sideband
26:21	0h RO	Reserved.
20	1h RO	**CLIST:** Indicates the controller contains a capabilities pointer list and the capability pointer register is implemented at offset 0x40 in the configuration space
19	0h RO	**INSTAT:** Interrupt Status: Reflects the state of the interrupt pin at the input of the enable/disable circuit. When the interrupt is asserted, and cleared when the interrupt is cleared (independent of the state of Interrupt Disable bit in command register. This bit is only associated with the INTx messages and has no meaning if the device is using MSI
18:11	0h RO	Reserved.
10	0h RW	**Interrupt Disable (IE):** Disables the function to generate INTx interrupt. A value of 0 enables the function to generate INTA messages on IOSF sideband. ***Note:*** This bit has no effect on MSI generation.
9	0h RO	Reserved.
8	0h RW	**System Error Enable (SERREn):** Setting this bit enables the generation of System Error message.
7:3	0h RO	Reserved.
2	0h RW	**Bus Master Enable (BME):** When set enables the ability to issue Memory or IO requests, including MSI.
1	0h RW	**Memory Space Enable (MSE):** When set, Memory Space Decoding is enabled and memory transactions targeting the device are accepted. ***Note:*** The MSE has to be set to accept any memory transaction on the primary interface targeting any of the Trace Hub's BARs
0	0h RO	Reserved.

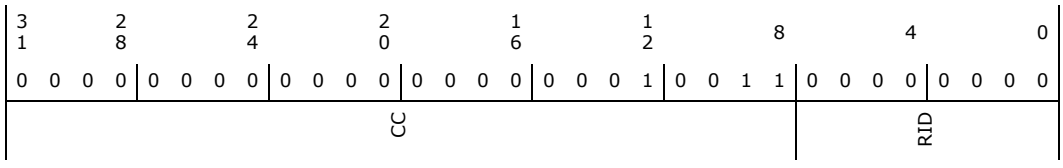
28.8.3 Revision ID (RID)—Offset 8h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 7

Default: 1300h



Bit Range	Default and Access	Field Name (ID): Description
31:8	13h RO	**Class Code (CC)**
7:0	0h RO	**Revision ID (RID):** Indicates stepping of the host controller. Refer to Device and Revision ID table in Vol1 of the datasheet for specific value.

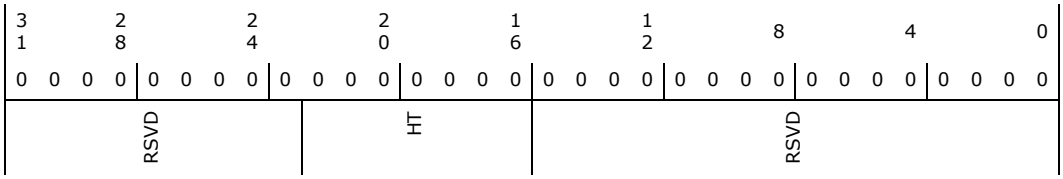
28.8.4 Header Type (HT)—Offset Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 7

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:23	0h RO	Reserved.
22:16	0h RO	**Header Type (HT):** Implements a Type 0 configuration header
15:0	0h RO	Reserved.

28.8.5 MTB Low BAR (MTB_LBAR)—Offset 10h

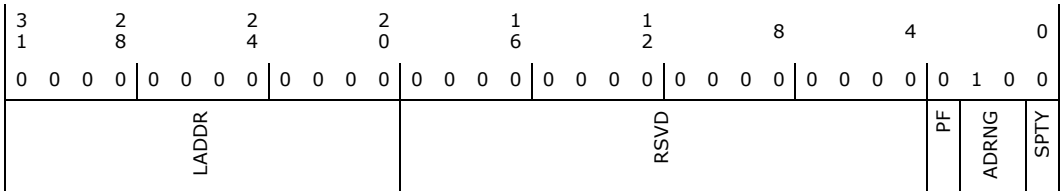
This register sets the lower BAR value for all memory mapped CSRs and the MTB.MSC Trace Buffer.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 7

Default: 4h



Bit Range	Default and Access	Field Name (ID): Description
31:20	0h RW	**Lower Base Address (LADDR):** Lower programmable Base Address.
19:4	0h RO	Reserved.
3	0h RO	**Prefetchable (PF):** Value of 0 indicates the BAR cannot be prefetched
2:1	2h RO	**Address Range (ADRNG):** Value of 0x2 indicates that the BAR is located anywhere system memory space (i.e., 64-bit addressing)
0	0h RO	**Space Type (SPTY):** Value of 0 indicates the BAR is located in memory space.

28.8.6 MTB Upper BAR (MTB_UBAR)—Offset 14h

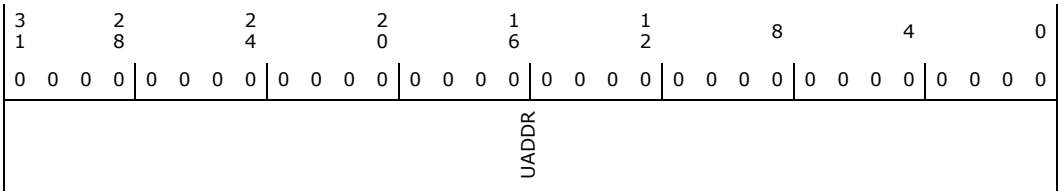
This register sets the upper BAR value for all memory mapped CSRs and the MTB.MSC Trace Buffer

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 7

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Upper Base Address (UADDR):** Upper programmable Base Address

28.8.7 SW Low BAR (SW_LBAR)—Offset 18h

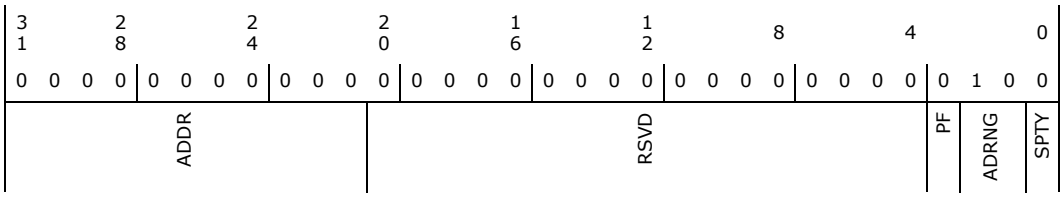
This register sets the lower 32 bits of the BAR value for the STMR STH target.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 7

Default: 4h



Bit Range	Default and Access	Field Name (ID): Description
31:21	0h RW	**Lower Base Address (ADDR):** Lower programmable Base Address
20:4	0h RO	Reserved.
3	0h RO	**Prefetchable (PF):** Value of 0 indicates the BAR cannot be prefetched
2:1	2h RO	**Address Range (ADRNG):** Value of 0x2 indicates that the BAR is located anywhere system memory space (i.e., 64-bit addressing)
0	0h RO	**Space Type (SPTY):** Value of 0 indicates the BAR is located in memory space

28.8.8 SW Upper BAR (SW_UBAR)—Offset 1Ch

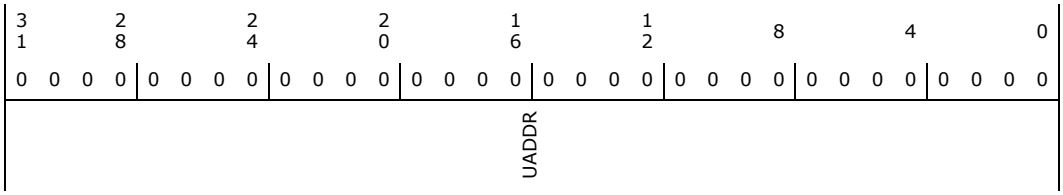
This register sets the upper 32 bits of the BAR value for the STMR STH target.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 7

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Upper Base Address (UADDR):** Upper programmable Base Address

28.8.9 RTIT Low BAR (RTIT_LBAR)—Offset 20h

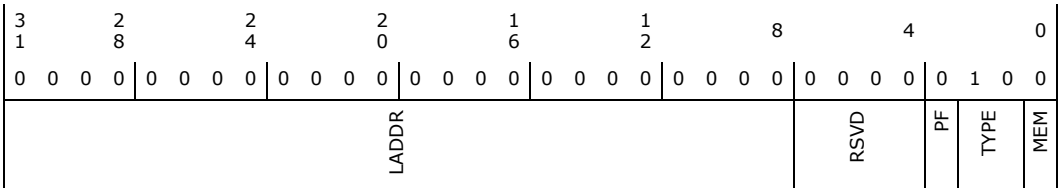
This register sets the lower BAR value for the STH RTIT trace sources. It is to be noted that if the RTIT_PRESENT parameter is not set, this register will become read-only and will return all zeros if read.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 7

Default: 4h



Bit Range	Default and Access	Field Name (ID): Description
31:8	0h RW	**Lower Base Address (LADDR):** Lower programmable Base Address
7:4	0h RO	Reserved.
3	0h RO	**Prefetchable (PF):** Value of 0 indicates the BAR cannot be prefetched
2:1	2h RO	**Address Range (TYPE):** Value of 0x2 indicates that the BAR is located anywhere system memory space (i.e., 64-bit addressing)
0	0h RO	**Space Type (MEM):** Value of 0 indicates the BAR is located in memory space

28.8.10 RTIT Upper BAR (RTIT_UBAR)—Offset 24h

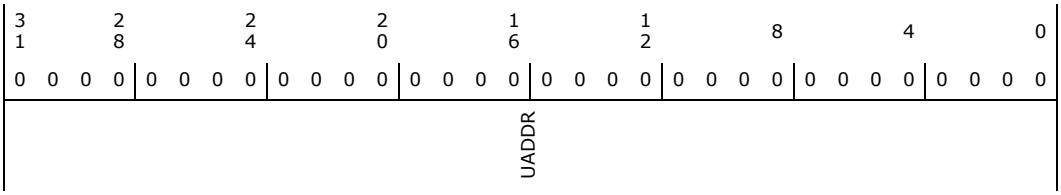
This register sets the upper BAR value for the Software Trace Hub RTIT trace sources. It is to be noted that if the RTIT_PRESENT parameter is not set, this register will become read-only and will return all zeros if read.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 7

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Upper Base Address (UADDR):** Upper programmable Base Address

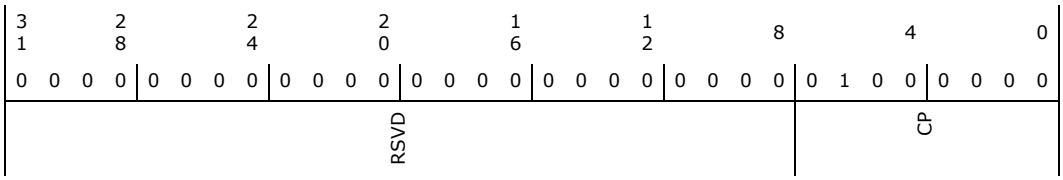
28.8.11 Capabilities Pointer (CAP)—Offset 34h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 7

Default: 40h



Bit Range	Default and Access	Field Name (ID): Description
31:8	0h RO	Reserved.
7:0	40h RO	**Capability Pointer (CP):** Pointer to first capability structure at 40h

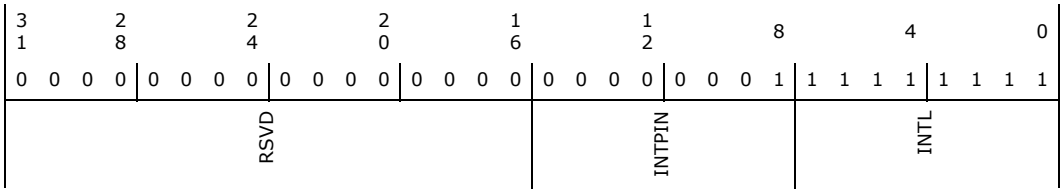
28.8.12 Interrupt Line and Interrupt Pin (INTL)—Offset 3Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 7

Default: 1FFh



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:8	1h RO	**Interrupt Pin (INTPIN):** The Trace Hub uses a single INTx interrupt bonded to INTA.
7:0	FFh RW	**Interrupt Line (INTL):** Hardware does not use this field. Rather it is programmed by system software and device drivers to communicate interrupt line routing information.

28.8.13 MSI Capability (MSICID)—Offset 40h

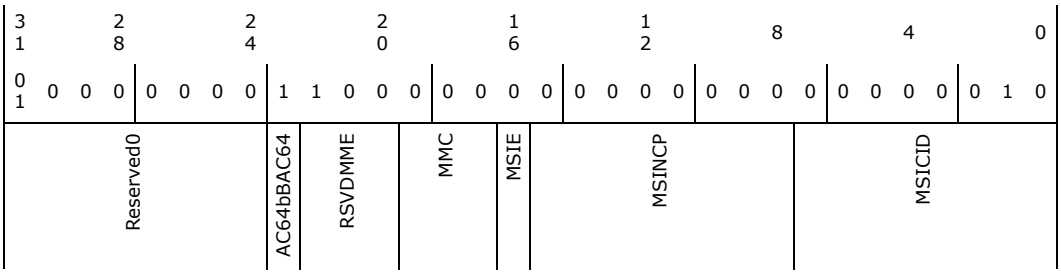
MSI Capability ID, MSI Next Capability Pointer, MSI Message Control Register

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 7

Default: 2800005h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	**Reserved0**
23	1h RO	**AC64b**
22:24	0h RO	Reserved.
23	1h RO	**64-bit Address Capable (BAC64):** Trace Hub is capable of generating 64-bit memory addresses.
22:20	0h RW	**Multiple Message Enable (MME):** Indicates the number of messages allocated to the device.
19:17	0h RO	**Multiple Message Capable (MMC):** Value of 0 indicates the device only support single interrupt message.
16	0h RW	**MSI Enable (MSIE):** If set, MSI is enabled and the legacy interrupts messages will not be generated.
15:8	0h RO	**Multiple Message Enable (MSINCP):** Indicates the number of messages allocated to the device.
7:0	5h RO	**MSI Capability ID (MSICID):** A value of 05h indicating the presence of the MSI capability register set.

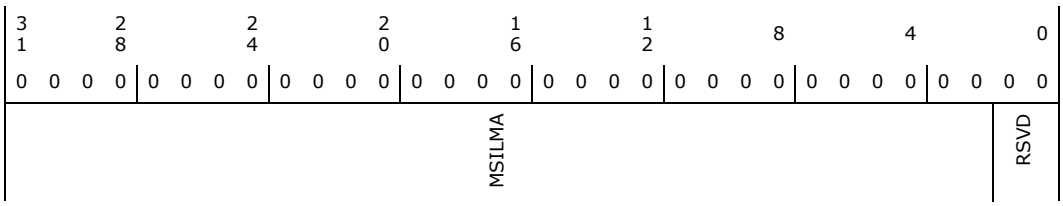
28.8.14 MSI Lower Message Address (MSILMA)—Offset 44h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 7

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:2	0h RW	**MSI Message Lower Address (MSILMA):** Lower 32-bits of system software assigned message address to the device with bits [1:0] always cleared indicating message address has to always be DW aligned.
1:0	0h RO	Reserved.

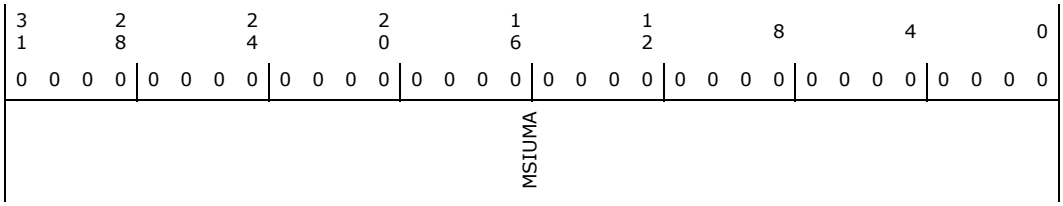
28.8.15 MSI Upper Message Address (MSIUMA)—Offset 48h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 7

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**MSI Message Upper Address (MSIUMA):** Upper 32-bits of system software assigned message address to the device

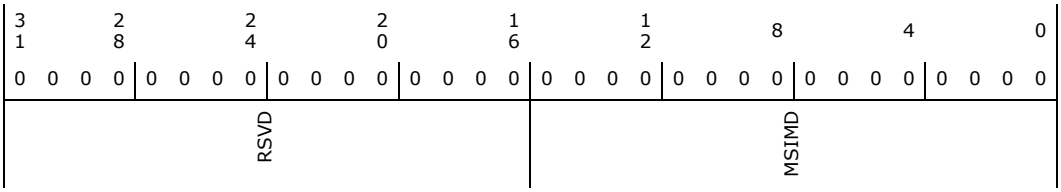
28.8.16 MSI Message Data (MSIMD)—Offset 4Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 7

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:0	0h RW	**MSI Message Data (MSIMD):** 16-bit message data pattern assigned by the system software to the device. When MSI is generated the actual data is 32-bit and the upper 16 bits are always 0.

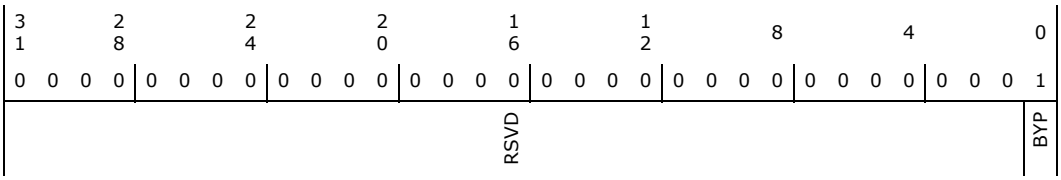
28.8.17 Device Specific Control and Device Specific Status (NPKDSC)—Offset 80h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 7

Default: 1h



Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	1h RW	**Trace Hub Bypass (BYP):** If set, the Trace Hub logic will be bypassed.

§

29 Universal Serial Bus

29.1 Acronyms

Acronyms	Description
USB	Universal Serial Bus
xHCI	eXtensible Host Controller Interface
UASP	USB Attached SCIS Protocol

29.2 References

Specification	Location
USB 3.0 Specification	www.usb.org
USB 2.0 Specification	www.usb.org

29.3 Overview

The PCH implements an xHCI USB controller which provides support for up to 14 USB 2.0 signal pairs and 10 SuperSpeed USB 3.0 signal pairs.

Use of USB devices certified by the USB-IF (http://usb.org) is recommended. Devices not certified by the USB-IF may have lower electrical margin and may not function reliably

Note: Each walk-up USB 3.0 capable port must have USB 3.0 signaling and USB 2.0 signaling.

Note: EHCI is no longer supported in PCH.

29.4 Signal Description

Name	Type	Description
USB3_1_RXN **USB3_1_RXP**	I	**USB 3.0 Differential Receive Pair 1:** These are USB 3.0-based high-speed differential signals for Port #1 and the xHCI Controller. It should map to a USB connector with one of the OC (overcurrent) pins.
USB3_1_TXN **USB3_1_TXP**	O	**USB 3.0 Differential Transmit Pair 1:** These are USB 3.0-based high-speed differential signals for Port #1 and the xHCI Controller. It should map to a USB connector with one of the OC (overcurrent) pins.
USB3_2_RXN **USB3_2_RXP**	I	**USB 3.0 Differential Receive Pair 2:** These are USB 3.0-based high-speed differential signals for Port #2 and the xHCI Controller. It should map to a USB connector with one of the OC (overcurrent) pins.
USB3_2_TXN **USB3_2_TXP**	O	**USB 3.0 Differential Transmit Pair 2:** These are USB 3.0-based high-speed differential signals for Port #2 and the xHCI Controller. It should map to a USB connector with one of the OC (overcurrent) pins.
USB3_3_RXN **USB3_3_RXP**	I	**USB 3.0 Differential Receive Pair 3:** These are USB 3.0-based high-speed differential signals for Port #3 and the xHCI Controller. It should map to a USB connector with one of the OC (overcurrent) pins.
USB3_3_TXN **USB3_3_TXP**	O	**USB 3.0 Differential Transmit Pair 3:** These are USB 3.0-based high-speed differential signals for Port #3 and the xHCI Controller. It should map to a USB connector with one of the OC (overcurrent) pins.

Name	Type	Description
USB3_4_RXN, USB3_4_RXP	I	**USB 3.0 Differential Receive Pair 4:** These are USB 3.0-based high-speed differential signals for Port #4 and the xHCI Controller. It should map to a USB connector with one of the OC (overcurrent) pins.
USB3_4_TXN, USB3_4_TXP	O	**USB 3.0 Differential Transmit Pair 4:** These are USB 3.0-based high-speed differential signals for Port #4 and the xHCI Controller. It should map to a USB connector with one of the OC (overcurrent) pins.
USB3_5_RXN USB3_5_RXP	I	**USB 3.0 Differential Receive Pair 5:** These are USB 3.0-based high-speed differential signals for Port #5 and the xHCI Controller. It should map to a USB connector with one of the OC (overcurrent) pins.
USB3_5_TXN USB3_5_TXP	O	**USB 3.0 Differential Transmit Pair 5:** These are USB 3.0-based high-speed differential signals for Port #5 and the xHCI Controller. It should map to a USB connector with one of the OC (overcurrent) pins.
USB3_6_RXN USB3_6_RXP	I	**USB 3.0 Differential Receive Pair 6:** These are USB 3.0-based high-speed differential signals for Port #6 and the xHCI Controller. It should map to a USB connector with one of the OC (overcurrent) pins.
USB3_6_TXN USB3_6_TXP	O	**USB 3.0 Differential Transmit Pair 6:** These are USB 3.0-based high-speed differential signals for Port #6 and the xHCI Controller. It should map to a USB connector with one of the OC (overcurrent) pins.
USB3_7_PCIE0_RXN USB3_7_PCIE0_RXP	I	**USB 3.0 Differential Receiver Pair 7**: These are USB 3.0-based high-speed differential signals for Port #7 and the xHCI Controller. It should map to a USB connector with one of the OC (overcurrent) pins. The pins also have PCIe root port 0 muxed on them. The function selection is via FITC
USB3_7_PCIE0_TXN USB3_7_PCIE0_TXP	O	**USB 3.0 Differential Transmitter Pair 7**: These are USB 3.0-based high-speed differential signals for Port #7 and the xHCI Controller. It should map to a USB connector with one of the OC (overcurrent) pins. The pins also have PCIe root port 0 muxed on them. The function selection is via FITC
USB3_8_PCIE1_RXN USB3_8_PCIE1_RXP	I	**USB 3.0 Differential Receiver Pair 8**: These are USB 3.0-based high-speed differential signals for Port #8 and the xHCI Controller. It should map to a USB connector with one of the OC (overcurrent) pins. The pins also have PCIe root port 1 muxed on them. The function selection is via FITC
USB3_8_PCIE1_TXN USB3_8_PCIE1_TXP	O	**USB 3.0 Differential Transmitter Pair 8**: These are USB 3.0-based high-speed differential signals for Port #8 and the xHCI Controller. It should map to a USB connector with one of the OC (overcurrent) pins. The pins also have PCIe root port 1 muxed on them. The function selection is via FITC
USB3_9_PCIE2_RXN USB3_9_PCIE2_RXP	I	**USB 3.0 Differential Receiver Pair 9**: These are USB 3.0-based high-speed differential signals for Port #9 and the xHCI Controller. It should map to a USB connector with one of the OC (overcurrent) pins. The pins also have PCIe root port 2 muxed on them. The function selection is via FITC
USB3_9_PCIE2_TXN USB3_9_PCIE2_TXP	O	**USB 3.0 Differential TransmitterPair 9**: These are USB 3.0-based high-speed differential signals for Port #9 and the xHCI Controller. It should map to a USB connector with one of the OC (overcurrent) pins. The pins also have PCIe root port 2 muxed on them. The function selection is via FITC
USB3_10_PCIE3_GBE_ RXN USB3_10_PCIE3_GBE_ RXP	I	**USB 3.0 Differential Receiver Pair 10**: These are USB 3.0-based high-speed differential signals for Port #10 and the xHCI Controller. It should map to a USB connector with one of the OC (overcurrent) pins. The pins also have PCIe root port 3 and the PHY connection for the integrated 1GbE MAC muxed on them. The function selection is via FITC
USB3_10_PCIE3_GBE_T XN USB3_10_PCIE3_GBE_T XP	O	**USB 3.0 Differential Transmitter Pair 10**: These are USB 3.0-based high-speed differential signals for Port #10 and the xHCI Controller. It should map to a USB connector with one of the OC (overcurrent) pins. The pins also have PCIe root port 3 and the PHY connection for the integrated 1GbE MAC muxed on them. The function selection is via FITC
USB2P_1, USB2N_1	I/O	**USB 2.0 Port 1 Transmit/Receive Differential Pair 1:** This USB 2.0 signal pair are routed to xHCI Controller and should map to a USB connector with one of the overcurrent OC pins.
USB2P_2, USB2N_2	I/O	**USB 2.0 Port 2 Transmit/Receive Differential Pair 2:** This USB 2.0 signal pair are routed to xHCI Controller and should map to a USB connector with one of the overcurrent OC pins.
USB2P_3, USB2N_3	I/O	**USB 2.0 Port 3Transmit/Receive Differential Pair 3:** This USB 2.0 signal pair are routed to xHCI Controller and should map to a USB connector with one of the overcurrent OC pins.

Name	Type	Description
USB2P_4, USB2N_4	I/O	**USB 2.0 Port 4 Transmit/Receive Differential Pair 4:** This USB 2.0 signal pair are routed to xHCI Controller and should map to a USB connector with one of the overcurrent OC pins.
USB2P_5, USB2N_5	I/O	**USB 2.0 Port 5 Transmit/Receive Differential Pair 5:** This USB 2.0 signal pair are routed to xHCI Controller and should map to a USB connector with one of the overcurrent OC pins.
USB2P_6, USB2N_6	I/O	**USB 2.0 Port 6 Transmit/Receive Differential Pair 6:** This USB 2.0 signal pair are routed to xHCI Controller and should map to a USB connector with one of the overcurrent OC pins.
USB2P_7, USB2N_7	I/O	**USB 2.0 Port 7 Transmit/Receive Differential Pair 7:** This USB 2.0 signal pair are routed to xHCI Controller and should map to a USB connector with one of the overcurrent OC pins.
USB2P_8, USB2N_8	I/O	**USB 2.0 Port 8 Transmit/Receive Differential Pair 8:** This USB 2.0 signal pair are routed to xHCI Controller and should map to a USB connector with one of the overcurrent OC pins.
USB2P_9, USB2N_9	I/O	**USB 2.0 Port 9 Transmit/Receive Differential Pair 9:** This USB 2.0 signal pair are routed to xHCI Controller and should map to a USB connector with one of the overcurrent OC pins.
USB2P_10, USB2N_10	I/O	**USB 2.0 Port 10 Transmit/Receive Differential Pair 10:** This USB 2.0 signal pair are routed to xHCI Controller and should map to a USB connector with one of the overcurrent OC pins.
USB2P_11 USB2N_11	I/O	**USB 2.0 Port 11 Transmit/Receive Differential Pair 11:** This USB 2.0 signal pair are routed to xHCI Controller and should map to a USB connector with one of the overcurrent OC pins.
USB2P_12 USB2N_12	I/O	**USB 2.0 Port 12 Transmit/Receive Differential Pair 12:** This USB 2.0 signal pair are routed to xHCI Controller and should map to a USB connector with one of the overcurrent OC pins.
USB2P_13 USB2N_13	I/O	**USB 2.0 Port 13 Transmit/Receive Differential Pair 13:** This USB 2.0 signal pair are routed to xHCI Controller and should map to a USB connector with one of the overcurrent OC pins.
USB2P_14 USB2N_14	I/O	**USB 2.0 Port 14 Transmit/Receive Differential Pair 14:** This USB 2.0 signal pair are routed to xHCI Controller and should map to a USB connector with one of the overcurrent OC pins.
GPP_E9_USB2_OC0#	I	**Overcurrent Indicators**: These signals set corresponding bits in the USB controller to indicate that an overcurrent condition has occurred.
GPP_E10_USB2_OC1#	I	**Overcurrent Indicators**: These signals set corresponding bits in the USB controller to indicate that an overcurrent condition has occurred.
GPP_E11_USB2_OC2#	I	**Overcurrent Indicators**: These signals set corresponding bits in the USB controller to indicate that an overcurrent condition has occurred.
GPP_E12_USB2_OC3#	I	**Overcurrent Indicators**: These signals set corresponding bits in the USB controller to indicate that an overcurrent condition has occurred.
GPP_F15_USB2_OC4#	I	**Overcurrent Indicators**: These signals set corresponding bits in the USB controller to indicate that an overcurrent condition has occurred.
GPP_F16_USB2_OC5#	I	**Overcurrent Indicators**: These signals set corresponding bits in the USB controller to indicate that an overcurrent condition has occurred.
GPP_F17_USB2_OC6#	I	**Overcurrent Indicators**: These signals set corresponding bits in the USB controller to indicate that an overcurrent condition has occurred.
GPP_F18_USB2_OC7#	I	**Overcurrent Indicators**: These signals set corresponding bits in the USB controller to indicate that an overcurrent condition has occurred.
USB2_COMP	I	USB Resistor Bias, analog connection points for an external resistor to ground.
USB2_VBUS	I	Not used in the Intel® C620 Series Chipset. This signal can either be tied to ground via external resistor for simple termination, or it can be left floating.

29.5 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
USB2N_[14:1]	Internal Pull-Down	14.25k-24.8k	1,2
USB2P_[14:1]	Internal Pull-Down	14.25k-24.8k	1,2
USB2_OC[7:0]#	None		
USB2_COMP	External bias resistor		

Notes:
1. Internal Pull-Down (14.25k-24.8k) and series resistors (45 ohm ±10%)
2. Internal pull-ups and pull-downs will not be valid until all the voltages are valid.

29.6 I/O Signal Planes and States

Signal Name	Power Plane	During Reset	Immediately after Reset	S3/S4/S5	Deep Sx
USB3_[10:1]_RXN10 **USB3_[10:1]_RXP10**	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF
USB3_[10:1]_TXN10 **USB3_[10:1]_TXP10**	Primary	Internal Pull-down	Internal Pull-down	Internal Pull-down	OFF
USB2N_[14:1]	DSW	Internal Pull-down	Internal Pull-down	Internal Pull-down	Internal Pull-down
USB2P_[14:1]	DSW	Internal Pull-down	Internal Pull-down	Internal Pull-down	Internal Pull-down
USB2_OC[7:0]#	Primary	Undriven	Undriven	Undriven	OFF
USB2_COMP	Primary	Undriven	Undriven	Undriven	OFF

29.7 Functional Description

29.7.1 eXtensible Host Controller Interface Controller (D20:F0)

The PCH contains an eXtensible Host Controller Interface host controller which supports up to 1 USB 2.0 ports and up to 10 USB 3.0 ports with board routing, ACPI table and BIOS considerations. This controller allows data transfers of up to 5 Gb/s. The controller supports SuperSpeed, High-Speed, Full-Speed and Low-Speed traffic on the bus. The xHCI controller supports USB Debug port on all USB3.0 capable ports. The xHCI supports USB debug capability on all USB 3.0 capable ports. The xHCI also supports USB Attached SCIS Protocol.

29.8 xHCI Configuration Registers Summary

Table 29-1. Summary of xHCI Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	1h	Vendor ID (VID)—Offset 0h	8086h
2h	3h	Device ID (DID)—Offset 2h	8C31h
4h	5h	Command (CMD)—Offset 4h	0h

Table 29-1. Summary of xHCI Configuration Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
6h	7h	Device Status (STS)—Offset 6h	290h
8h	8h	Revision ID (RID)—Offset 8h	0h
9h	9h	Programming Interface (PI)—Offset 9h	30h
Ah	Ah	Sub Class Code (SCC)—Offset Ah	3h
Bh	Bh	Base Class Code (BCC)—Offset Bh	Ch
Dh	Dh	Master Latency Timer (MLT)—Offset Dh	0h
Eh	Eh	Header Type (HT)—Offset Eh	0h
10h	17h	Memory Base Address (MBAR)—Offset 10h	4h
2Ch	2Dh	USB Subsystem Vendor ID (SSVID)—Offset 2Ch	0h
2Eh	2Fh	USB Subsystem ID (SSID)—Offset 2Eh	0h
34h	34h	Capabilities Pointer (CAP_PTR)—Offset 34h	70h
3Ch	3Ch	Interrupt Line (ILINE)—Offset 3Ch	0h
3Dh	3Dh	Interrupt Pin (IPIN)—Offset 3Dh	0h
40h	43h	XHC System Bus Configuration 1 (XHCC1)—Offset 40h	1FDh
44h	47h	XHC System Bus Configuration 2 (XHCC2)—Offset 44h	3C000h
50h	53h	Clock Gating (XHCLKGTEN)—Offset 50h	0h
58h	5Bh	Audio Time Synchronization (AUDSYNC)—Offset 58h	0h
60h	60h	Serial Bus Release Number (SBRN)—Offset 60h	30h
61h	61h	Frame Length Adjustment (FLADJ)—Offset 61h	60h
62h	62h	Best Effort Service Latency (BESL)—Offset 62h	0h
70h	70h	PCI Power Management Capability ID (PM_CID)—Offset 70h	1h
71h	71h	Next Item Pointer #1 (PM_NEXT)—Offset 71h	80h
72h	73h	Power Management Capabilities (PM_CAP)—Offset 72h	C1C2h
74h	75h	Power Management Control/Status (PM_CS)—Offset 74h	8h
80h	80h	Message Signaled Interrupt CID (MSI_CID)—Offset 80h	5h
81h	81h	Next item pointer (MSI_NEXT)—Offset 81h	0h
82h	83h	Message Signaled Interrupt Message Control (MSI_MCTL)—Offset 82h	86h
84h	87h	Message Signaled Interrupt Message Address (MSI_MAD)—Offset 84h	0h
88h	8Bh	Message Signaled Interrupt Upper Address (MSI_MUAD)—Offset 88h	0h
8Ch	8Dh	Message Signaled Interrupt Message Data (MSI_MD)—Offset 8Ch	0h
94h	97h	Vendor Specific Header (VSHDR)—Offset 94h	1400010h
A2h	A3h	Power Control Enable (PCE_REG) —Offset A2h	0008h
B0h	B3h	XHCI USB2 Overcurrent Pin Mapping N (U2OCM)—Offset B0h	0h
D0h	D3h	XHCI USB3 Overcurrent Pin Mapping N (U3OCM)—Offset D0h	0h

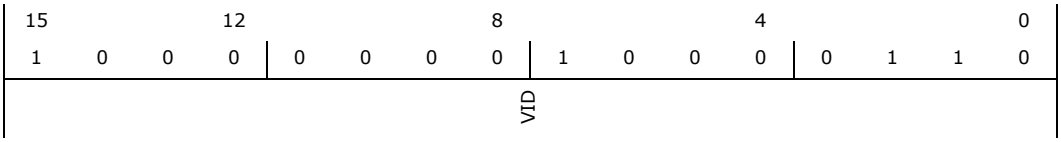
29.8.1 Vendor ID (VID)—Offset 0h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 20
Function: 0

Default: 8086h



Bit Range	Default and Access	Field Name (ID): Description
15:0	8086h RO	**Vendor ID (VID)**

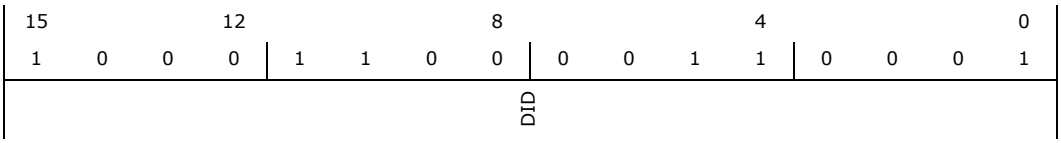
29.8.2 Device ID (DID)—Offset 2h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 20
Function: 0

Default: 8C31h



Bit Range	Default and Access	Field Name (ID): Description
15:0	8C31h RO/V	**Device ID (DID):** See Global Device ID table in Chap. 6 for value

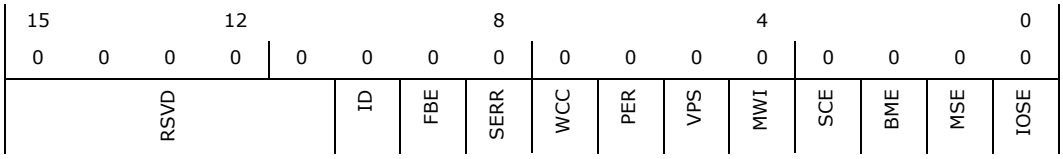
29.8.3 Command (CMD)—Offset 4h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:11	0h RO	Reserved.
10	0h RW	**Interrupt Disable (ID):** When cleared to 0, the function is capable of generating interrupts. When 1, the function can not generate its interrupt to the interrupt controller. Note that the corresponding Interrupt Status bit is not affected by the interrupt enable.
9	0h RO	**Fast Back to Back Enable (FBE)**
8	0h RW	**SERR# Enable (SERR):** When set to 1, the XHC is capable of generating (internally) SERR#. See section on error handling.
7	0h RO	**Wait Cycle Control (WCC)**
6	0h RW	**Parity Error Response (PER):** When set to 1, the XHCI Host Controller will check for correct parity (on its internal interface) and halt operation when bad parity is detected during the data phase as recommended by the XHCI specification. Note that this applies to both requests and completions from the system interface. This bit must be set in order for the parity errors to generate SERR#.
5	0h RO	**VGA Palette Snoop (VPS)**
4	0h RO	**Memory Write Invalidate (MWI)**
3	0h RO	**Special Cycle Enable (SCE)**
2	0h RW	**Bus Master Enable (BME):** When set, it allows XHC to act as a bus master. When cleared, it disable XHC from initiating transactions on the system bus.
1	0h RW	**Memory Space Enable (MSE):** This bit controls access to the XHC Memory Space registers. If this bit is set, accesses to the XHC registers are enabled. The Base Address register for the XHC should be programmed before this bit is set.
0	0h RO	**I/O Space Enable (IOSE):** Reserved as 0. Read-Only.

29.8.4 Device Status (STS)—Offset 6h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 20
Function: 0

Default: 290h

15			12				8				4				0
0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0
DPE	SSE	RMA	RTA	STA	DEVT		MDPED	FBBC	UDF	MC	CL	IS	RSVD		

Bit Range	Default and Access	Field Name (ID): Description
15	0h RW/1C	**Detected Parity Error (DPE):** This bit is set by the Intel PCH whenever a parity error is seen on the internal interface to the XHC host controller, regardless of the setting of bit 6 or bit 8 in the Command register or any other conditions. Software clears this bit by writing a 1 to this bit location.
14	0h RW/1C	**Signaled System Error (SSE):** This bit is set by the Intel PCH whenever it signals SERR# (internally). The SERR_EN bit (bit 8 in the Command Register) must be 1 for this bit to be set. See error handling section for complete list of conditions handled. Software clears this bit by writing a 1 to this bit location.
13	0h RW/1C	**Received Master-Abort Status (RMA):** This bit is set when XHC, as a master, receives a master-abort status on a memory access. This is treated as a Host Error and halts the DMA engines. Software clears this bit by writing a 1 to this bit location.
12	0h RW/1C	**Received Target Abort Status (RTA):** This bit is set when XHC, as a master, receives a target abort status on a memory access. This is treated as a Host Error and halts the DMA engines. Software clears this bit by writing a 1 to this bit location.
11	0h RW/1C	**Signaled Target-Abort Status (STA):** This bit is used to indicate when the XHC function responds to a cycle with a target abort.
10:9	1h RO	**DEVSEL# Timing Status (DEVT):** This 2-bit field defines the timing for DEVSEL# assertion. Read-Only.
8	0h RW/1C	**Master Data Parity Error Detected (MDPED):** This bit is set by the Intel PCH whenever a data parity error is detected on a XHC read completion packet on the internal interface to the XHC host controller and bit 6 of the Command register is set to 1. Software clears this bit by writing a 1 to this bit location.
7	1h RO	**Fast Back-to-Back Capable (FBBC):** Reserved as 1 Read-Only.
6	0h RO	**User Definable Features (UDF):** Reserved as 0. Read-Only.
5	0h RO	**66 MHz Capable (MC):** Reserved as 0. Read-Only.
4	1h RO	**Capabilities List (CL):** Hardwired to 1 indicating that offset 34h contains a valid capabilities pointer.
3	0h RO/V	**Interrupt Status (IS):** This read-only bit reflects the state of this function's interrupt at the input of the enable/disable logic. This bit is a 1 when the interrupt is asserted. This bit will be 0 when the interrupt is deasserted. The value reported in this bit is independent of the value in the Interrupt Enable bit.
2:0	0h RO	Reserved.

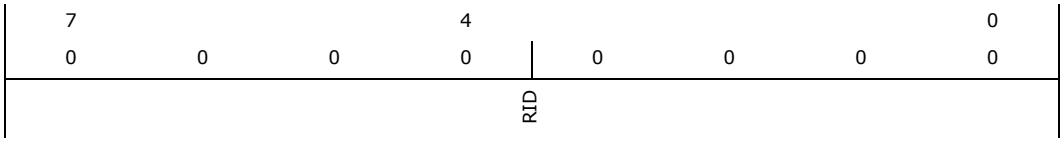
29.8.5 Revision ID (RID)—Offset 8h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO/V	**Revision ID (RID):** See Chap 2 for value.

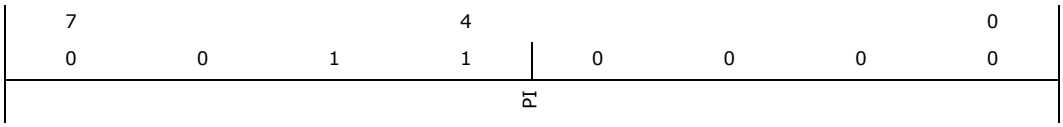
29.8.6 Programming Interface (PI)—Offset 9h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 0

Default: 30h



Bit Range	Default and Access	Field Name (ID): Description
7:0	30h RO	**Programming Interface (PI):** A value of 30h indicates that this USB Host Controller conforms to the XHCI specification.

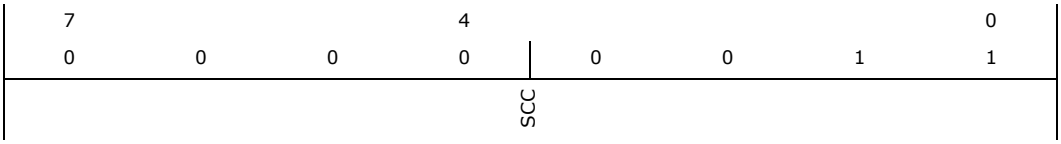
29.8.7 Sub Class Code (SCC)—Offset Ah

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 0

Default: 3h



Bit Range	Default and Access	Field Name (ID): Description
7:0	3h RO	**Sub Class Code (SCC):** A value of 03h indicates that this is a Universal Serial Bus Host Controller.

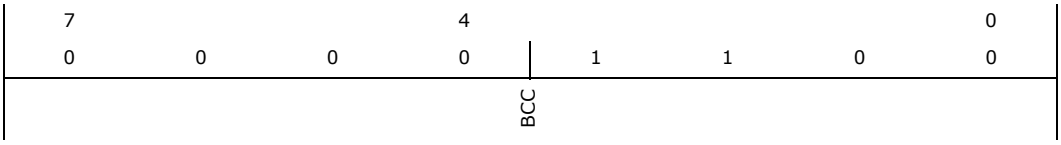
29.8.8 Base Class Code (BCC)—Offset Bh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 0

Default: Ch



Bit Range	Default and Access	Field Name (ID): Description
7:0	Ch RO	**Base Class Code (BCC):** A value of 0Ch indicates that this is a Serial Bus controller.

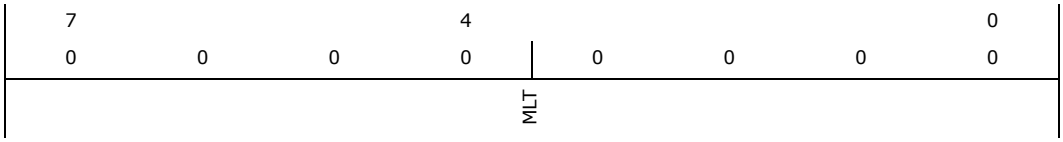
29.8.9 Master Latency Timer (MLT)—Offset Dh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO	**Master Latency Timer (MLT):** Because the XHC controller is internally implemented with arbitration on an internal interface, it does not need a master latency timer. The bits will be fixed at 0.

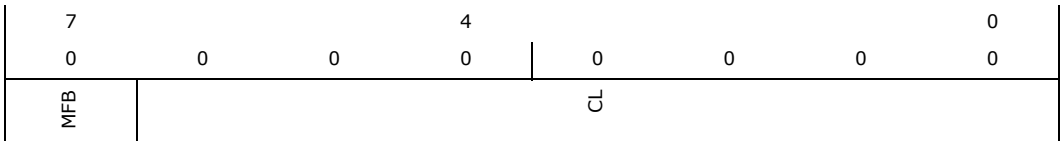
29.8.10 Header Type (HT)—Offset Eh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7	0h RO	**Multi-Function Bit (MFB):** Read only indicating single function device.
6:0	0h RO	**Configuration layout (CL):** Hardwired to 0 to indicate a standard PCI configuration layout.

29.8.11 Memory Base Address (MBAR)—Offset 10h

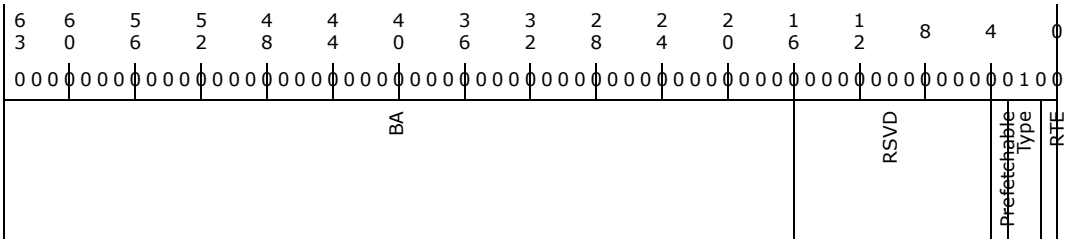
Value in this register will be different after the enumeration process.

Access Method

Type: CFG Register
(Size: 64 bits)

Device: 20
Function: 0

Default: 4h



Bit Range	Default and Access	Field Name (ID): Description
63:16	0h RW	**Base Address (BA):** Bits (63:16) correspond to memory address signals (63:16), respectively. This gives 64 KB of relocatable memory space aligned to 64 KB boundaries.
15:4	0h RO	Reserved.
3	0h RO	**Prefetchable (Prefetchable):** This bit is hardwired to 0 indicating that this range should not be prefetched.
2:1	2h RO	**Type (Type):** If this field is hardwired to 00 it indicates that this range can be mapped anywhere within 32-bit address space. If this field is hardwired to 10 it indicates that this range can be mapped anywhere within 64-bit address space.
0	0h RO	**Resource Type Indicator (RTE):** This bit is hardwired to 0 indicating that the base address field in this register maps to memory space

29.8.12 USB Subsystem Vendor ID (SSVID)—Offset 2Ch

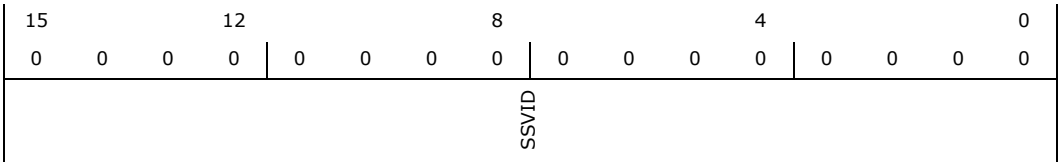
This register is modified and maintained by BIOS.

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:0	0h RW/L	**USB Subsystem Vendor ID (SSVID):** This register, in combination with the USB Subsystem ID register, enables the operating system to distinguish each subsystem from the others.

29.8.13 USB Subsystem ID (SSID)—Offset 2Eh

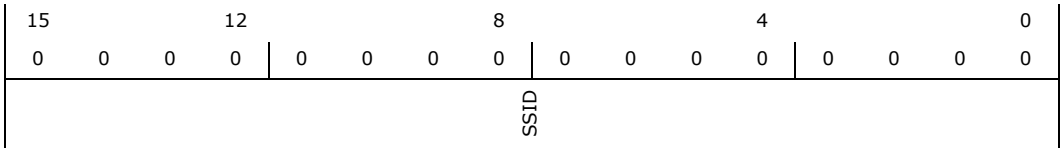
This register is modified and maintained by BIOS.

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:0	0h RW/L	**USB Subsystem ID (SSID):** BIOS sets the value in this register to identify the Subsystem ID. This register, in combination with the Subsystem Vendor ID register, enables the operating system to distinguish each subsystem from other(s).

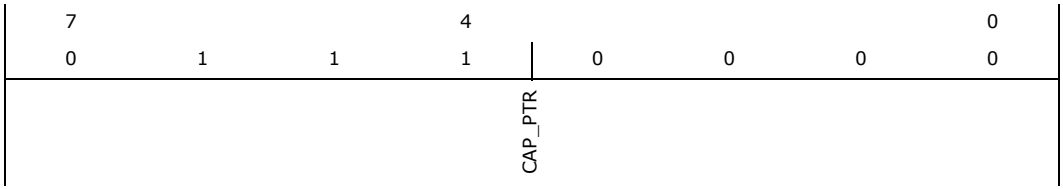
29.8.14 Capabilities Pointer (CAP_PTR)—Offset 34h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 0

Default: 70h



Bit Range	Default and Access	Field Name (ID): Description
7:0	70h RO	**Capabilities Pointer (CAP_PTR):** This register points to the starting offset of the capabilities ranges.

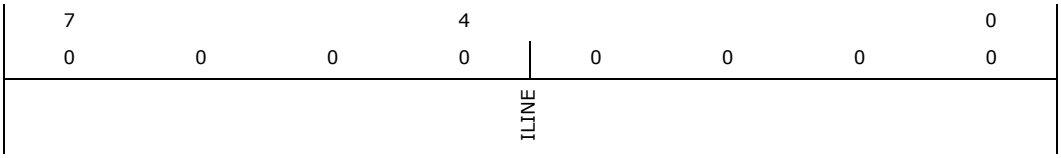
29.8.15 Interrupt Line (ILINE)—Offset 3Ch

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW	**Interrupt Line (ILINE):** This data is not used by the Intel PCH. It is used as a scratchpad register to communicate to software the interrupt line that the interrupt pin is connected to.

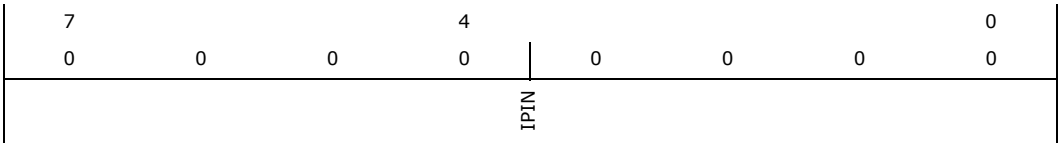
29.8.16 Interrupt Pin (IPIN)—Offset 3Dh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW/L	**Interrupt pin (IPIN):** Bits 7:0 reflect the Interrupt Pin assigned to the host controller by the platform (and are hardwired).

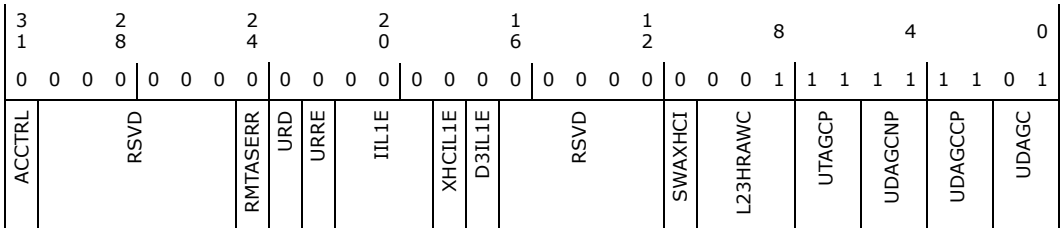
29.8.17 XHC System Bus Configuration 1 (XHCC1)—Offset 40h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 1FDh



Bit Range	Default and Access	Field Name (ID): Description
31	0h RW/O	**Access Control (ACCTRL):** This bit is used by BIOS to lock/unlock lockable bits. When set to '1' the write access to bits locked by this bit is disabled (locked state). When set to '0', the write access to bit locked by this bit is enabled (unlocked state). Writable once after platform reset.
30:25	0h RO	Reserved.
24	0h RW	**Master/Target Abort SERR (RMTASERR):** When set, it allows the out-of-band error reporting from the xHCI Controller to be reported as SERR# (if SERR# reporting is enabled) and thus set the STS.SSE bit.
23	0h RW/C	**Unsupported Request Detected (URD):** Set the HW when xHCI Controller received an unsupported request posted cycle. Cleared by SW when the bit is written with value of '1'.
22	0h RW	**Unsupported Request Report Enable (URRE):** When set this bit allows the URD bit to be reported as SERR# (if SERR# reporting is enabled) and thus set the STS.SSE bit.
21:19	0h RW	**Inactivity Initiated L1 Enable (IIL1E):** If programmed to non-zero, it allows L1 power managed to be enabled after the time-out period specified. 000: Disabled 001: 32 bb_cclk 010: 64 bb_cclk 011: 128 bb_cclk 100: 256 bb_cclk 101: 512 bb_cclk 110: 1024 bb_cclk 111: 131072 bb_cclk
18	0h RW	**XHC Initiated L1 Enable (XHCIL1E):** If set, allow the XHC initiated L1 power management to be enabled.
17	0h RW	**D3 Initiated L1 Enable (D3IL1E):** If set, allow PCI device state D3 initiated L1 power management to be enables. This bit can only be set if the XHCI L1 Override P2 chicken bit is set.
16:12	0h RO	Reserved.
11	0h RW	**SW Assisted xHC Idle (SWAXHCI):** This bit being set will indicate xHC idleness (through SW means), which must be a '1' to allow L1 entry, and subsequently allow backbone clock to be gated. This bit is to be set by Intel xHCI driver after checking that the xHCI Controller will stay in idle state for a significant period of time e.g., all ports disconnected. This bit can be cleared under the following conditions (see SWAXHCI Policy bits in xHC System Bus Configuration 2 register) SW: SW could write 0 to clear this bit.n HW: HW, under policy control, will clear this bit on an MMIO access to the Host Controller.n HW: HW, under policy control, will clear this bit when HW exits Idle state.

Bit Range	Default and Access	Field Name (ID): Description
10:8	1h RW	**L23 to Host Reset Acknowledge Wait Count (L23HRAWC):** If programmed to non zero, it allows a wait period after the L23 PHY has shutdown before returning host reset acknowledge to PMC. 000: Disabled 001: 128 bb_cclk 010: 256 bb_cclk 011: 512 bb_cclk 100: 1024 bb_cclk 101: 2048 bb_cclk 110: 4096 bb_cclk 111: 131072 bb_cclk
7:6	3h RW	**Upstream Type Arbiter Grant Count Posted (UTAGCP):** Grant count for IOSF upstream L2 request type arbiter for posted type
5:4	3h RW	**Upstream Type Arbiter Grant Count Non Posted (UDAGCNP):** Grant count for IOSF upstream L2 type arbiter for non-posted type
3:2	3h RW	**Upstream Type Arbiter Grant Count Completion (UDAGCCP) (UDAGCCP):** Grant count for IOSF upstream L2 type arbiter for completion type
1:0	1h RW	**Upstream Device Arbiter Grant Count (UDAGC) (UDAGC):** Grant count for IOSF upstream L1 device arbiter

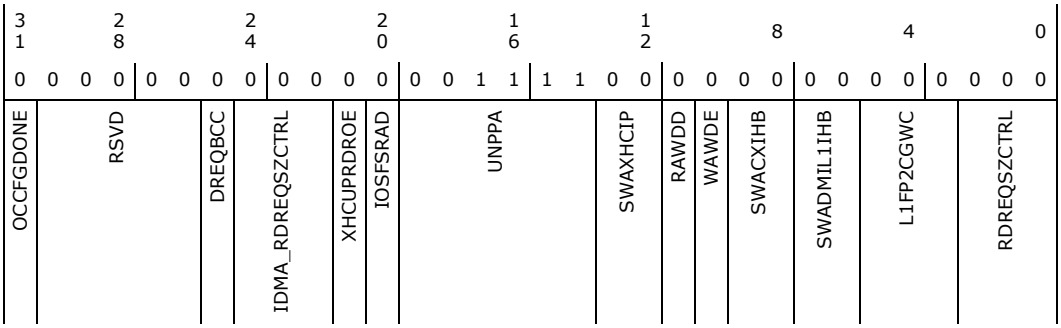
29.8.18 XHC System Bus Configuration 2 (XHCC2)—Offset 44h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 3C000h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RW	**OC Configuration Done (OCCFGDONE):** This bit is used by BIOS to prevent spurious switching during OC configuration. It must be set by BIOS after configuration of the OC mapping bits is complete. Once this bit is set, OC mapping shall not be changed by SW.
30:26	0h RO	Reserved.
25	0h RW	**DMA Request Boundary Crossing Control (DREQBCC):** This bit controls the boundary crossing limit of each Read/Write Request. 0: 4 KB 1: 64B

Bit Range	Default and Access	Field Name (ID): Description
24:22	0h RW	**IDMA Read Request Size Control (IDMA_RDREQSZCTRL):** Read Request Size Control: This bit controls the maximum size of each Read Request. 000: 128B 001: 256B 011 - 110: Reserved 111: 64B
21	0h RW	**XHC Upstream Read Relaxed Ordering Enable (XHCUPRDROE):** This policy controls the Relaxed Ordering attribute for upstream reads. 0 - xHC will clear RO for all upstream read requests. 1 - xHC will set RO for all upstream read requests.
20	0h RW	**IOSF Sideband Register Access Disable (IOSFSRAD):** When set, it disables the IOSF sideband interface from accepting any host space register access.
19:14	Fh RW	**Upstream Non-Posted Pre-Allocation (UNPPA):** This field reserves data sizes, in 64 byte chunks, of the downstream completion resource. This value is zero based. 000000 - 111111: Pre-allocate 64 bytes - 4096 bytes If set greater than the default allows over-allocation If set less than default allows under-allocation Only allowed to be programmed when BME = 0 and no outstanding downstream completion
13:12	0h RW	**SW Assisted xHC Idle Policy (SWAXHCIP):** ***Note:*** Irrespective of the setting of this field, SW write of 0 to SWAXHCI will clear the bit. 00b (default): xHC HW clears SWAXHCI bit upon OR xHC HW exits Idle state 01b: xHC HW does not autonomously clear SWAXHCI bit. The bit could be cleared only by SW. 10b: xHC HW clears SWAXHCI upon MMIO access to Host Controller. xHC HW exit from Idle state will not clear SWAXHCI. 11b: Reserved
11	0h RW	**MMIO Read After MMIO Write Delay Disable (RAWDD):** This field controls delay on MMIO Read after MMIO Write. 0b (Default): Delay MMIO Read after MMIO Write 1b: Do not delay MMIO Read after MMIO Write Note that this delay applies after the second of the two DW writes in the case where the IOSF Gasket splits a QW write into two single DW writes to the IP.
10	0h RW	**MMIO Write After MMIO Write Delay Enable (WAWDE):** This field controls delay on MMIO Write after previous MMIO Write. 0b (Default): Do not delay MMIO Write after previous MMIO Write 1b: Delay MMIO Write after previous MMIO Write Note that the delay count does not apply on the second of the two DW writes that are generated by IOSF Gasket when it splits a QW write into two. In other words, the second of the two DW writes could happen without any delay with respect to the first DW write. This choice is being made for ease of ECO. The delay count, in this case, will apply after the second of the two DW writes.
9:8	0h RW	**SW Assisted Cx Inhibit (SWACXIHB):** This field controls how the DMI L1 inhibit signal from USB3 to PMC will behave. 00: Never inhibit Cx 01: Inhibit Cx when Isochronous Endpoint is active (PPT Behavior) 10: Inhibit Cx when Periodic Active as defined in 40.4.3.2. 111: Always inhibit Cx
7:6	0h RW	**SW Assisted DMI L1 Inhibit (SWADMIL1IHB):** This field controls how the DMI L1 inhibit signal from USB3 to DMI will behave. 00: Never inhibit DMI L1. 01: Inhibit DMI L1 when Isochronous Endpoint is active (PPT Behavior). 10: Inhibit DMI L1 when Periodic Active as defined in 40.4.3.2.1. 11: Inhibit DMI L1 if XHCC1.SWAXHCI = 0.
5:3	0h RW	**L1 Force P2 Clock Gating Wait Count (L1FP2CGWC):** If programmed to non zero, it allows L1 force P2 gating off the clock to be delayed after the time-out period specified. If wake up event is detected before the time-out, pclk remains alive and trigger L1 exit as though CPU host is causing the wake, 000: Disabled 001: 128 bb_cclk 010: 256 bb_cclk 011: 512 bb_cclk 100: 1024 bb_cclk 101: 2048 bb_cclk 110: 4096 bb_cclk 111: 131072 bb_cclk
2:0	0h RW	**Read Request Size Control (RDREQSZCTRL):** Read Request Size Control: This bit controls the maximum size of each Read Request. 000: 128B 001: 256B 010: 512B 011 - 110: Reserved 111: 64B

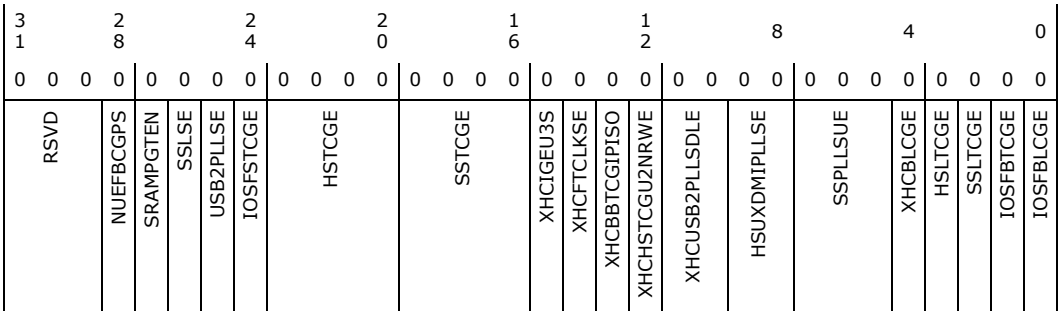
29.8.19 Clock Gating (XHCLKGTEN)—Offset 50h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:29	0h RO	Reserved.
28	0h RW	**Naking USB2.0 EPs for Backbone Clock Gating and PLL Shutdown (NUEFBCGPS):** This field controls whether Naking USB2.0 EPs, once in Naking low priority schedule, should be considered as active for the considerations for backbone clock gating and PLL shutdown or not. 0: Naking USB2.0 EPs are not considered to be active for Backbone Clock Gating and PLL Shutdown evaluation. 1: Naking USB2.0 EPs are considered to be active for Backbone clock gating and PLL shutdown evaluation.
27	0h RW	**SRAM Power Gate Enable (SRAMPGTEN):** This register enables the SRAM Power Gating when PLL shutdown conditions for all clock domains have been met 0 - Disallow SRAM Power Gating. 1 - Allow SRAM Power Gating
26	0h RW	**SS Link PLL Shutdown Enable (SSLSE):** This register enables the SS P3 state to be exposed to PXP PLL Shutdown conditions on behalf of all USB SS Ports on top of trunk clock gating. 0 - P3 state NOT allowed to result in PXP PLL shutdown. 1- P3 state allowed to result in PXP PLL shutdown
25	0h RW	**USB2 PLL Shutdown Enable (USB2PLLSE):** When set, this bit allows USB2 PLL to be shutdown when HS Link trunk clock is gated, and xHC can tolerate PLL spin up time for subsequent clock request. ***Note:*** If USB2 PLL Shutdown Disable Fuse is '1', hardware will always see '0' as an output from this register. BIOS reading this register should always return the correct value.
24	0h RW	**IOSF Sideband Trunk Clock Gating Enable (IOSFSTCGE):** When set, this bit allows the IOSF sideband clock trunk to be gated when idle conditions are met.
23:20	0h RW	**HS Backbone PXP Trunk Clock Gate Enable (HSTCGE):** This register determines the HS Ux state(s) which will be exposed to Backbone PXP trunk gating of core clock. Uy is a state allowed to result in trunk gating when ss_tcg_ux_en(x) is asserted. (0) ==) U0 or deeper (1) ==) NA (no support for U1) (2) ==) U2 (L1) or deeper (3) ==) U3 (L2) or deeper
19:16	0h RW	**SS Backbone PXP Trunk Clock Gate Enable (SSTCGE):** This register determines the SS Ux state(s) which will be exposed to Backbone PXP trunk gating of core clock. Uy is a state allowed to result in trunk gating when ss_tcg_ux_en(x) is asserted. (0) ==> U0 or deeper (1) ==> U1 or deeper (2) ==> U2 or deeper (3) ==> U3 or deeper

Bit Range	Default and Access	Field Name (ID): Description
15	0h RW	**XHC Ignore_EU3S (XHCIGEU3S):** This register determines if the xHC will use the EU3S as a condition to allow for Frame timer gating. 0 - xHC may allow frame timer to be gated when EU3S is set and all ports are in the required state. 1 - xHC may allow frame timer to be gated regardless of EU3S.
14	0h RW	**XHC Frame Timer Clock Shutdown Enable (XHCFTCLKSE):** This register determines if the xHC will allow the frame timer clock to be gated. 0 - xHC will not allow ICC PLL 96MHz output to be shutdown thus keeping the frame timer running. 1 - xHC will allow ICC PLL 96MHz output to be shutdown under specific conditions.
13	0h RW	**XHC Backbone PXP Trunk Clock Gate In Presence of ISOCH EP (XHCBBTCGIPISO):** This register controls the policy on allowing Backbone PXP trunk clock gate in the presence of IDLE ISOCH EP s with active DB. Allows the periodic active to be used in enabling Backbone PXP trunk clock gating of core clock. 0 - Trunk gate of core clock is not allowed when ISOCH EP DB is set and ISOCH EP s are idle. 1 - Allow trunk gate of core clock when ISOCH EP DB is set and ISOCH EP s are idle.
12	0h RW	**XHC HS Backbone PXP Trunk Clock Gate U2 non RWE (XHCHSTCGU2NRWE):** This register controls the policy on allowing Backbone PXP trunk gating of core clock when there is at least 1 non Remote Wake Enabled HS Port in U2. 0 - Prevent trunk gate of core clock when a non RWE HS Port is in U2. 1 - Allow trunk gate of core clock when a non RWE HS Port is in U2.
11:10	0h RW	**XHC USB2 PLL Shutdown Lx Enable (XHCUSB2PLLSDLE):** This register determines the HS Link state(s) which will be exposed to USB2 PLL Shutdown conditions on behalf of all USB2 HS Ports. Ly is a state allowed to result in USB2 PLL Shutdown when en(x) is asserted. (0) ==> L1 or deeper (1) ==> L2 or deeper
9:8	0h RW	**HS Backbone PXP PLL Shutdown Ux Enable (HSUXDMIPLLSE):** This register determines the Ux state(s) which will be exposed to PXP PLL Shutdown conditions. PLL Shutdown is allowed in: 00b Disabled (Link states shall be disabled for DMI PLL shutdown) 01b U0 or conditions for 10b setting. 10b U2 or conditions for 11b setting. 10b U3, Disconnected, Disabled or Powered-Off.
7:5	0h RW	**SS Backbone PXP PLL Shutdown Ux Enable (SSPLLSUE):** This register determines the Ux state(s) which will be exposed to DMI PLL Shutdown conditions. PLL Shutdown is allowed in: 000b Disabled (Link states shall be ignored for DMI PLL shutdown). 001b U0 or conditions for 010b setting. 010b U1 or conditions for 011b setting. 011b U2 or conditions for 100b setting. 100b U3, Disconnected, Disabled or Powered-Off
4	0h RW	**XHC Backbone Local Clock Gating Enable (XHCBLCGE):** When set, this bit allows XHCI Controller IP backbone clock to be locally gated when idle conditions are met. ***Note:*** If XHC Dynamic Clock Gating Disable Fuse is '1', hardware will always see '0' as an output from this register. BIOS reading this register should always return the correct value.
3	0h RW	**HS Link Trunk Clock Gating Enable (HSLTCGE):** When set, this bit allows High Speed Link control's 480 MHz and its 48/60 MHz link clock trunk to be gated when idle conditions are met. ***Note:*** If XHC Dynamic Clock Gating Disable Fuse is '1', hardware will always see '0' as an output from this register. BIOS reading this register should always return the correct value. BIOS need to query fuses to see if HW is enabled.
2	0h RW	**SS Link Trunk Clock Gating Enable (SSLTCGE):** When set, this bit allows the SuperSpeed Link control's 250 MHz and its divided 125 MHz link clock trunk to GE gated when idle conditions are met. ***Note:*** If XHC Dynamic Clock Gating Disable Fuse is '1', hardware will always see '0' as an output from this register. BIOS reading this register should always return the correct value. BIOS need to query fuses to see if HW is enabled.
1	0h RW	**IOSF Backbone Trunk Clock Gating Enable (IOSFBTCGE):** When set, this bit allows the IOSF backbone clock trunk to be gated when idle conditions are met.
0	0h RW	**IOSF Gasket Backbone Local Clock Gating Enable (IOSFBLCGE):** When set, this bit allows the IOSF Gasket backbone clock to be locally gated when idle conditions are met. ***Note:*** If XHC Dynamic Clock Gating Disable Fuse is '1', hardware will always see '0' as an output from this register. BIOS reading this register should always return the correct value. BIOS need to query fuses to see if HW is enabled.

29.8.20 Audio Time Synchronization (AUDSYNC)—Offset 58h

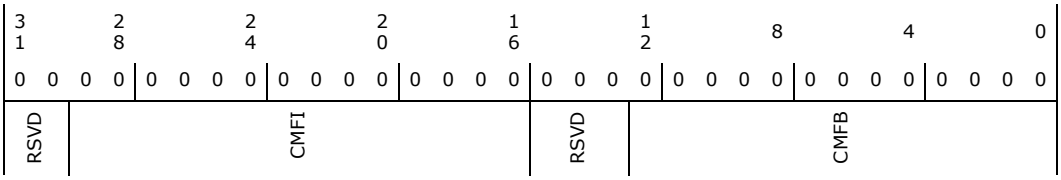
This 32 bit register is used for audio stream synchronization across different devices. Global signal sample_now captures a value in AUDSYNC register.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:30	0h RO	Reserved.
29:16	0h RO/V	**Captured Frame List Current Index/Frame Number (CMFI):** The value in this register is updated in response to sample_now signal. Bits (29:16) reflect state of bits (13:0) of FRINDEX
15:13	0h RO	Reserved.
12:0	0h RO/V	**Captured Micro-frame BLIF (CMFB):** The value is updated in response to sample_now signal and provides information about offset within micro-frame. Captured value represents number of 8 high-speed bit time units from start of micro-frame. At the beginning of micro-frame captured value will be 0 and increase to maximum value at the end. Default maximum value is 7499 but it may be changed as result of adjustment done in FLA.

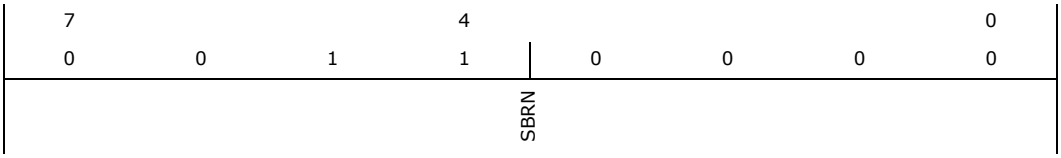
29.8.21 Serial Bus Release Number (SBRN)—Offset 60h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 0

Default: 30h



Bit Range	Default and Access	Field Name (ID): Description
7:0	30h RO	**Serial Bus Release Number (SBRN):** A value of 30h indicates that this controller follows USB release 3.0.

29.8.22 Frame Length Adjustment (FLADJ)—Offset 61h

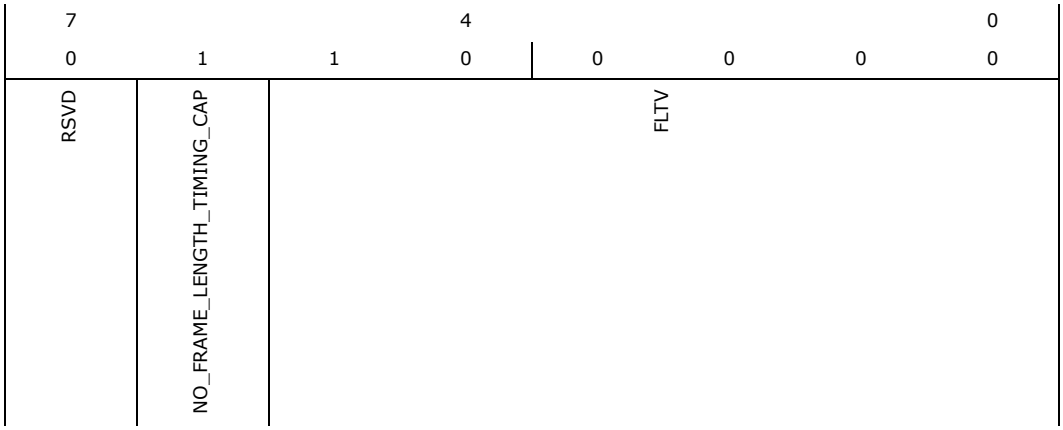
This feature is used to adjust any offset from the clock source that generates the clock that drives the SOF counter. When a new value is written into these six bits, the length of the frame is adjusted. Its initial programmed value is system dependent based on the accuracy of hardware USB clock and is initialized by system BIOS. This register should only be modified when the HChalted bit in the USBSTS register is a one. Changing value of this register while the host controller is operating yields undefined results.

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 0

Default: 60h



Bit Range	Default and Access	Field Name (ID): Description
7	0h RO	Reserved.
6	1h RO	**No Frame Length Timing Capability (NO_FRAME_LENGTH_TIMING_CAP):** This flag is set to 1 to indicate that the host controller does not support a programmable Frame Length Timing Value field.
5:0	20h RO	**Frame Length Timing Value (FLTV):** SOF micro-frame length) is equal to 59488 + value in this field. The default value is decimal 32 (20h), which gives a SOF cycle time of 60000. Frame Length (number of High Speed bit times) FLADJ Value (decimal) (decimal) 59488 0 (00h) 59504 1 (01h) 59520 2 (02h)... 59984 31 (1Fh) 60000 32 (20h)... 60480 62 (3Eh) 60496 63 (3Fh) Each decimal value change to this register corresponds to 16 high-speed bit times. The SOF cycle time (number of SOF counter clock periods to generate a SOF micro-frame length) is equal to 59488 + value in this field. The default value is decimal 32 (20h), which gives a SOF cycle time of 60000. Frame Length (# High Speed bit times) FLADJ Value

29.8.23 Best Effort Service Latency (BESL)—Offset 62h

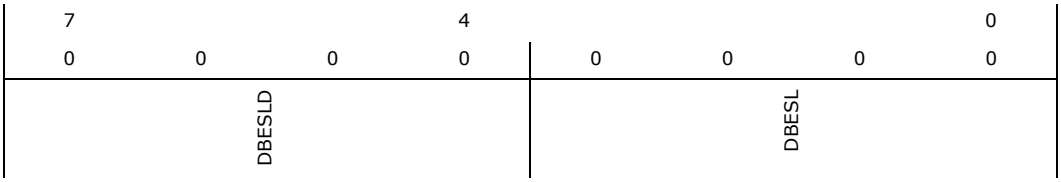
Bset Effort Service Latency.

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:4	0h RW/L	**Default Best Effort Service Latency Deep (DBESLD):** Default Best Effort Service Latency (DBESLD) If the value of this field is non-zero, it defines the recommended value for programming the PORTPMSC register BESLD field. This is programmed by BIOS based on platform parameters.
3:0	0h RW/L	**Default Best Effort Service Latency (DBESL):** If the value of this field is non-zero, it defines the recommended value for programming the PORTPMSC register BESL field. This is programmed by BIOS based on platform parameters.

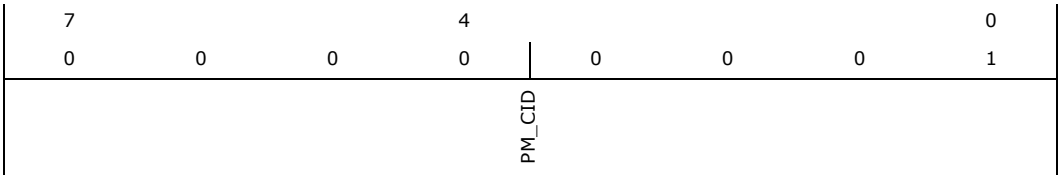
29.8.24 PCI Power Management Capability ID (PM_CID)—Offset 70h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 0

Default: 1h



Bit Range	Default and Access	Field Name (ID): Description
7:0	1h RO	**PCI Power Management Capability ID (PM_CID):** A value of 01h indicates that this is a PCI Power Management capabilities field.

29.8.25 Next Item Pointer #1 (PM_NEXT)—Offset 71h

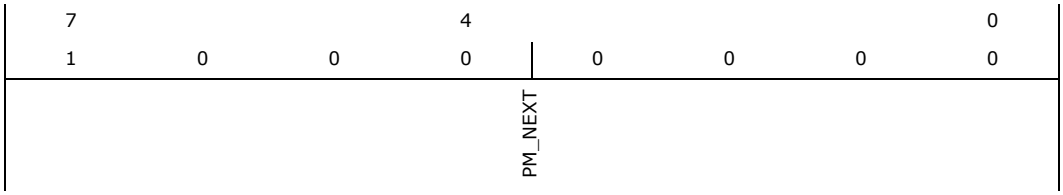
This register is modified and maintained by BIOS.

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 0

Default: 80h



Bit Range	Default and Access	Field Name (ID): Description
7:0	80h RW/L	**Next Item Pointer #1 (PM_NEXT):** This register defaults to 80h, which indicates that the next capability registers begin at configuration offset 80h. This register is writable when the Access Control bit is set to '0'. This allows BIOS to effectively hide the next capability registers, if necessary. This register should only be written during system initialization before the plug-and-play software has enabled any master-initiated traffic. Values of: 80h implies next capability is MSI 00h implies that MSI capability is hidden. ***Note:*** This value is never expected to be programmed.

29.8.26 Power Management Capabilities (PM_CAP)—Offset 72h

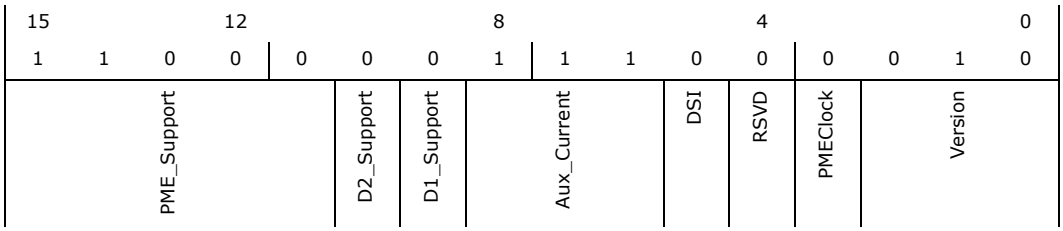
Normally, this register is read-only to report capabilities to the power management software. In order to report different power management capabilities depending on the system in which the Intel PCH is used, the write access to this register is controlled by the Access Control bit (ACCTRL). The value written to this register does not affect the hardware other than changing the value returned during a read. This register is modified and maintained by BIOS.

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 20
Function: 0

Default: C1C2h



Bit Range	Default and Access	Field Name (ID): Description
15:11	18h RW/L	**PME_Support (PME_Support):** This 5-bit field indicates the power states in which the function may assert PME#. The Intel PCH XHC does not support the D1 or D2 states. For all other states, the Intel PCH XHC is capable of generating PME#. Software should never need to modify this field.
10	0h RW/L	**D2_Support (D2_Support):** The D2 state is not supported.
9	0h RW/L	**D1_Support (D1_Support):** The D1 state is not supported.
8:6	7h RW/L	**Aux_Current (Aux_Current):** The Intel PCH XHC reports 375mA maximum Suspend well current required when in the $D3_{COLD}$ state. This value can be written by BIOS when a more accurate value is known.
5	0h RW/L	**DSI (DSI):** The Intel PCH reports 0, indicating that no device-specific initialization is required.
4	0h RO	Reserved.
3	0h RW/L	**PME Clock (PMEClock):** The Intel PCH reports 0, indicating that no PCI clock is required to generate PME#.
2:0	2h RW/L	**Version (Version):** The Intel PCH reports 010, indicating that it complies with Revision 1.1 of the PCI Power Management Specification.

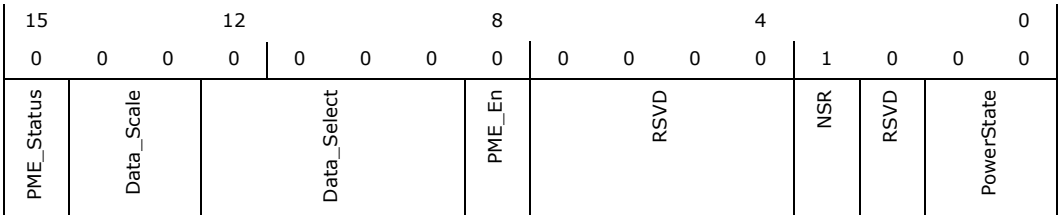
29.8.27 Power Management Control/Status (PM_CS)—Offset 74h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 20
Function: 0

Default: 8h



Bit Range	Default and Access	Field Name (ID): Description
15	0h RW/1C	**PME_Status (PME_Status):** This bit is set when the Intel PCH XHC would normally assert the PME# signal independent of the state of the PME_En bit. Writing a 1 to this bit will clear it and cause the internal PME to deassert (if enabled). Writing a 0 has no effect. This bit must be explicitly cleared by the operating system each time the operating system is loaded.
14:13	0h RO	**Data_Scale (Data_Scale):** The Intel PCH hardwires these bits to 00 because it does not support the associated Data register.
12:9	0h RO	**Data_Select (Data_Select):** The Intel PCH hardwires these bits to 0000 because it does not support the associated Data register.
8	0h RW	**PME_En (PME_En):** A 1 enables the Intel PCH XHC to generate an internal PME signal when PME_Status is 1. This bit must be explicitly cleared by the operating system each time it is initially loaded.

Bit Range	Default and Access	Field Name (ID): Description
7:4	0h RO	Reserved.
3	1h RO	**No Soft Reset (NSR):**), this bit indicates that devices transitioning from $D3_{HOT}$ to D0 because of PowerState commands do not perform an internal reset. Configuration Context is preserved. Upon transition from the $D3_{HOT}$ to the D0 Initialized state, no additional operating system intervention is required to preserve Configuration Context beyond writing the PowerState bits. Transition from $D3_{HOT}$ to D0 by a system or bus segment reset will return to the device state D0 Uninitialized with only PME context preserved if PME is supported and enabled.
2	0h RO	Reserved.
1:0	0h RW	**PowerState (PowerState):** This 2-bit field is used both to determine the current power state of XHC function and to set a new power state. The definition of the field values are: 00b - D0 state 11b - $D3_{HOT}$ state If software attempts to write a value of 10b or 01b in to this field, the write operation must complete normally, however, the data is discarded and no state change occurs. When in the $D3_{HOT}$ state, the Intel PCH must not accept accesses to the XHC memory range, but the configuration space must still be accessible.

29.8.28 Message Signaled Interrupt CID (MSI_CID)—Offset 80h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 0

Default: 5h

7			4				0
0	0	0	0	0	1	0	1
CID							

Bit Range	Default and Access	Field Name (ID): Description
7:0	5h RO	**Capability ID (CID):** Indicates that this is an MSI capability

29.8.29 Next item pointer (MSI_NEXT)—Offset 81h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 0

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
NEXT							

Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW/L	**Next Pointer (NEXT):** Indicates that this is the last item on the capability list

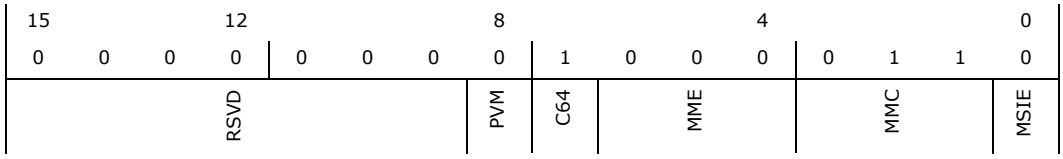
29.8.30 Message Signaled Interrupt Message Control (MSI_MCTL)—Offset 82h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 20
Function: 0

Default: 86h



Bit Range	Default and Access	Field Name (ID): Description
15:9	0h RO	Reserved.
8	0h RO	**Per-Vector Masking Capable (PVM):** Specifies whether controller supports MSI per vector masking. Not supported
7	1h RO	**64 Bit Address Capable (C64):** Specifies whether capable of generating 64-bit messages. This device is 64-bit capable.
6:4	0h RW	**Multiple Message Enable (MME):** Indicates the number of messages the controller should assert. This device supports multiple message MSI.
3:1	3h RO	**Multiple Message Capable (MMC):** Indicates the number of messages the controller wishes to assert.This field must be set by HW to reflect the number of Interrupters supported. Encoding number of Vectors requested (number of Interrupters) 000 1 001 2 010 4 011 8 100 16 101 32 110-111 Reserved
0	0h RW	**MSI Enable (MSIE):** If set to 1, MSI is enabled and the traditional interrupt pins are not used to generate interrupts. If cleared to 0, MSI operation is disabled and the traditional interrupt pins are used.

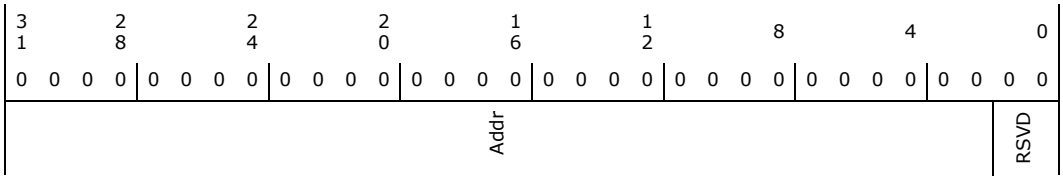
29.8.31 Message Signaled Interrupt Message Address (MSI_MAD)—Offset 84h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:2	0h RW	**Addr (Addr):** Lower DW of system specified message address, always DWORD aligned
1:0	0h RO	Reserved.

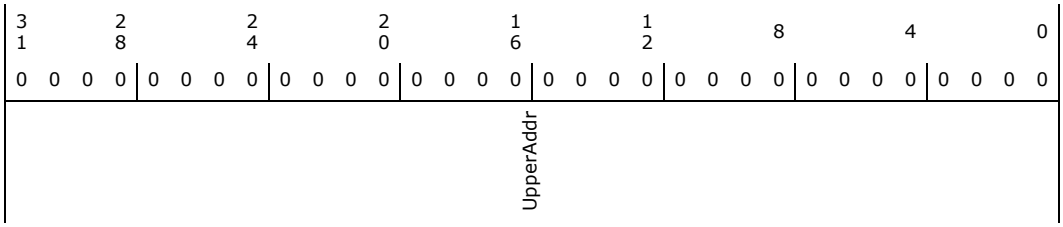
29.8.32 Message Signaled Interrupt Upper Address (MSI_MUAD)—Offset 88h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Upper Addr (UpperAddr):** Upper DW of system specified message address.

29.8.33 Message Signaled Interrupt Message Data (MSI_MD)—Offset 8Ch

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 20
Function: 0

Default: 0h

Bit Range	Default and Access	Field Name (ID): Description
15:0	0h RW	**Data (Data):** This 16-bit field is programmed by system software if MSI is enabled. Its content is driven onto the lower word (PCI AD(15:0)) during the data phase of the MSI memory write transaction. The Multiple Message Enable field (bits 6-4 of the Message Control register) defines the number of low order message data bits the function is permitted to modify to generate its system software allocated vectors. For example, a Multiple Message Enable encoding of 010 indicates the function has been allocated four vectors and is permitted to modify message data bits 1 and 0 (a function modifies the lower message data bits to generate the allocated number of vectors). If the Multiple Message Enable field is 000, the function is not permitted to modify the message data.

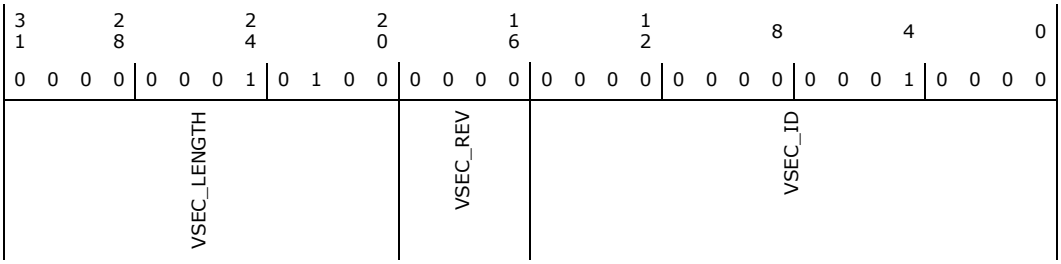
29.8.34 Vendor Specific Header (VSHDR)—Offset 94h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 1400010h



Bit Range	Default and Access	Field Name (ID): Description
31:20	14h RO	**VSEC Length (VSEC_LENGTH):** This field indicates the number of bytes in the entire VSEC structure, including the PCI Extended Capability header, the Vendor-Specific header, and the Vendor-Specific register
19:16	0h RO	**VSEC Rev (VSEC_REV):** This field is a vendor-defined version number that indicates the version of the VSEC structure. Software must qualify the Vendor ID and VSEC ID before interpreting this field.
15:0	10h RO	**VSEC ID (VSEC_ID):** This field is a vendor-defined ID number that indicates the nature and format of the VSEC structure. Software must qualify the Vendor ID before interpreting this field.

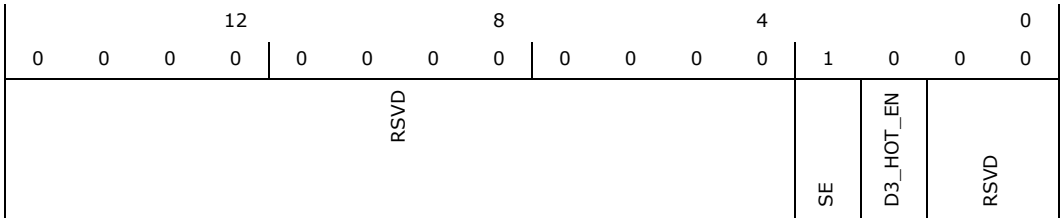
29.8.35 Power Control Enable (PCE_REG) —Offset A2h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 0008h



Bit Range	Default and Access	Field Name (ID): Description
31:4	0000000h RO	**Reserved**
3	1b RW	**Sleep Enable (SE):** 0: xHCI will never assert Sleep 1: xHCI may assert Sleep.
2	0b RW	**D3 Hot Enable (D3_HOT_EN):** 0: xHCI will not power gate when Idle 1: xHCI will power gate when idle.
1:0	00b RO	**Reserved**

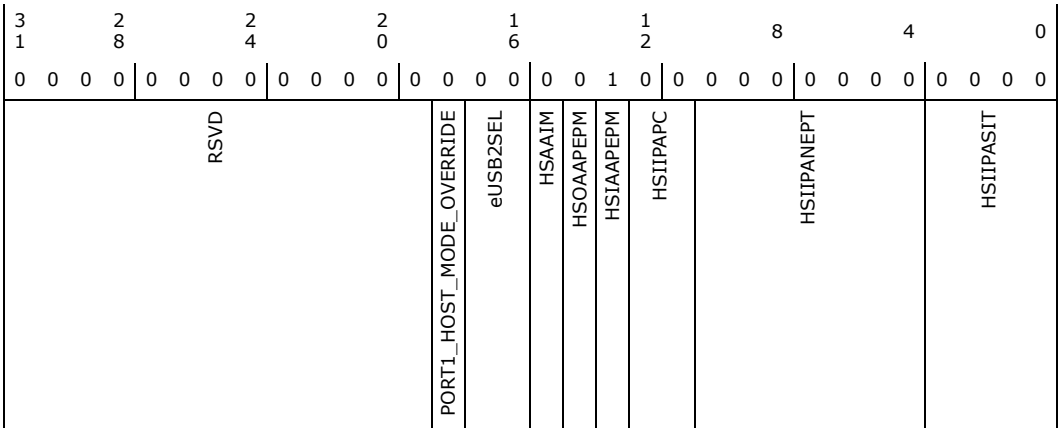
29.8.36 High Speed Configuration 2 (HSCFG2)—Offset A4h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 2000h



Bit Range	Default and Access	Field Name (ID): Description
31:19	0h RO	Reserved.
18	0h RW	**PORT1_HOST_MODE_OVERRIDE (PORT1_HOST_MODE_OVERRIDE):** When set, this bit causes the Host_Device mux on port 1 to be forced into the Host mode.
17:16	0h RW	**eUSB2SEL (eUSB2SEL):** The two bits are associate with USB2 ports 1 - bit 16 and 2 - bit 2 0: Port is mapped to USB2 1: Port is mapped to eUSB2
15	0h RW	**HS ASYNC Active IN Mask (HSAAIM):** Determines if the Async Active will mask/ ignore IN EP s. 0 HS ASYNC Active will include IN EP s. 1 HS ASYNC Active will mask/ ignore IN EP s.
14	0h RW	**HS OUT ASYNC Active Polling EP Mask (HSOAAPEPM):** Determines if the Async Active for OUT HS/FS/LS masks/ignores EP s that are polling/PINGing (HS) due to NAK. 0 HS OUT ASYNC Active will include EP s that are polling. 1 HS OUT ASYNC Active will mask/ignore EP s that are polling.
13	1h RW	**HS IN ASYNC Active Polling EP Mask (HSIAAPEPM):** Determines if the Async Active for IN HS/FS/LS masks/ignores EP s that are polling due to NAK. 0 HS IN ASYNC Active will include EP s that are polling. 1 HS IN ASYNC Active will mask/ ignore EP s that are polling.
12:11	0h RW	**HS INTR IN Periodic Active Policy Control (HSIIPAPC):** Controls how the HS INTR IN periodic active is used to generate the global periodic active. This will determine how the smallest service interval among active EP s and number of active EP s are used. 0 HS INTR IN periodic active will be used to generate periodic active if Service Interval Threshold OR Numb of EP Threshold values meet the requirement. 1 HS INTR IN periodic active will be used to generate periodic active if Service Interval Threshold AND Numb of EP Threshold values meet the requirement. 2 Always allow HS INTR EP s to be used in the generation of the global Periodic Active indication. 3 Never allow HS INTR EP s to be used in the generation of the global Periodic Active indication
10:4	0h RW	**HS INTR IN Periodic Active Num of EP Threshold (HSIIPANEPT):** Defines the threshold used to determine if Periodic Active may include HS/FS/LS INTR IN EP active indication. If there are more than NumEPThreshold active HS/FS/LS INTR EP s then they may be included as part of the periodic active generation.

Bit Range	Default and Access	Field Name (ID): Description
3:0	0h RW	**HS INTR IN Periodic Active Service Interval Threshold (HSIIPASIT):** Defines the Service Interval threshold used to determine if Periodic Active will include HS/FS/LS INTR IN EP active indication. If there are any active HS/FS/LS INTR EP s with a service interval less than or equal to this threshold then they may be included as part of the periodic active generation.

29.8.37 XHCI USB2 Overcurrent Pin Mapping N (U2OCM)—Offset B0h

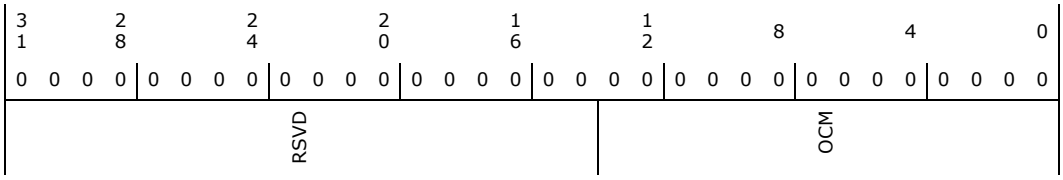
Address Offset: B0-B3h, ... (B0h+(NumOC-1)*4) to (B3h+(NumOC–1)*4)
The RW/L property of this register is controlled by OCCFDONE bit.
Each OC pin can be assigned to one or more of up to 32 Standard USB2 ports.
Each DWord maps one OC pin across up to 32 USB2 ports.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:0	0h RW/L	**OC Mapping (OCM):** USB2 Port assignment When Set to 1, Bit 0 maps the OC pin N to USB2 std. port 1 Bit 1 maps the OC pin N to USB2 std port 2 ... Bit (NumUSB2std-1) maps the OC pin N to USB2 Std port NumUSB2std ***Note:*** The USB-R port which is the most significant USB2 port does not have an OC pin. Thus the OC assignment for the USB-R port is ignored.

29.8.38 XHCI USB3 Overcurrent Pin Mapping N (U3OCM)—Offset D0h

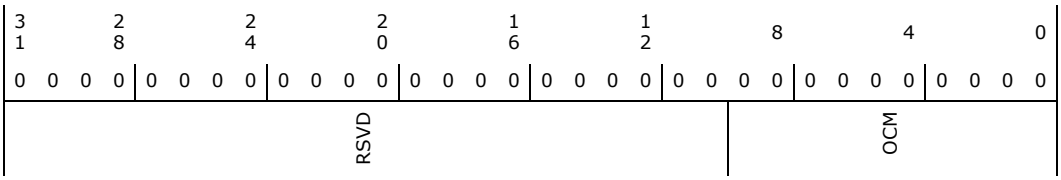
Address Offset: D0-D3h, ... (D0h+(NumOC-1)*4) to (D3h+(NumOC–1)*4)
The RW/L property of this register is controlled by OCCFDONE bit.
Each OC pin can be assigned to one or more of up to 32 Standard USB2 ports.
Each DWord maps one OC pin across up to 32 USB2 ports.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:10	0h RO	Reserved.
9:0	0h RW/L	**OC Mapping (OCM):** USB3 Port assignment When Set to 1, Bit 0 maps the OC pin N to USB3 std. port 1 Bit 1 maps the OC pin N to USB3 std port 2 ... Bit (NumUSB3std-1) maps the OC pin N to USB3 Std port NumUSB3std

29.9 xHCI Memory Mapped Registers Summary

Table 29-2. Summary of xHCI Memory Mapped Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	0h	Capability Registers Length (CAPLENGTH)—Offset 0h	80h
2h	3h	Host Controller Interface Version Number (HCIVERSION)—Offset 2h	100h
4h	7h	Structural Parameters 1 (HCSPARAMS1)—Offset 4h	18000840h
8h	Bh	Structural Parameters 2 (HCSPARAMS2)—Offset 8h	14200054h
Ch	Fh	Structural Parameters 3 (HCSPARAMS3)—Offset Ch	40001h
10h	13h	Capability Parameters (HCCPARAMS)—Offset 10h	200077C1h
14h	17h	Doorbell Offset (DBOFF)—Offset 14h	3000h
18h	1Bh	Runtime Register Space Offset (RTSOFF)—Offset 18h	2000h
80h	83h	USB Command (USBCMD)—Offset 80h	0h
84h	87h	USB Status (USBSTS)—Offset 84h	1h
88h	8Bh	Page Size (PAGESIZE)—Offset 88h	1h
94h	97h	Device Notification Control (DNCTRL)—Offset 94h	0h
98h	9Bh	Command Ring Low (CRCR_LO)—Offset 98h	0h

Table 29-2. Summary of xHCI Memory Mapped Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
9Ch	9Fh	Command Ring High (CRCR_HI)—Offset 9Ch	0h
B0h	B3h	Device Context Base Address Array Pointer Low (DCBAAP_LO)—Offset B0h	0h
B4h	B7h	Device Context Base Address Array Pointer High (DCBAAP_HI)—Offset B4h	0h
480h	483h	Port N Status and Control USB2 (PORTSCN)—Offset 480h	2A0h
484h	487h	Port Power Management Status and Control USB2 (PORTPMSCN)—Offset 484h	0h
48Ch	48Fh	Port N Hardware LPM Control Register (PORTHLPMCN)—Offset 48Ch	0h
580h	583h	Port Status and Control USB3 (PORTSCXUSB3)—Offset 580h	2A0h
584h	587h	Port Power Management Status and Control USB3 (PORTPMSCX)—Offset 584h	0h
588h	58Bh	USB3 Port Link Info (PORTLI)—Offset 588h	0h
2000h	2003h	Microframe Index (RTMFINDEX)—Offset 2000h	0h
2020h	2023h	Interrupt x Management (IMANx) - Offset 2020h	0h
2024h	2027h	Interrupter X Moderation (IMODx) - Offset 2024h	00000FA0h
2028h	202Bh	Event Ring Segment Table Size x (ERSTSZx)—Offset 2028h	0h
2030h	2033h	Event Ring Segment Table Base Address Low x (ERSTBA_LOx)—Offset 2030h	0h
2034h	2037h	Event Ring Segment Table Base Address High x (ERSTBA_HIx)—Offset 2034h	0h
2038h	203Bh	Event Ring Dequeue Pointer Low x (ERDP_LOx)—Offset 2038h	0h
203Ch	203Fh	Event Ring Dequeue Pointer High x (ERDP_HIx)—Offset 203Ch	0h
3000h	3003h	Door Bell x (DBx)—Offset 3000h	0h
8000h	8003h	XECP_SUPP_USB2_0 (XECP_SUPP_USB2_0)—Offset 8000h	2000802h
8004h	8007h	XECP_SUPP_USB2_1 (XECP_SUPP_USB2_1)—Offset 8004h	20425355h
8008h	800Bh	XECP_SUPP_USB2_2 (XECP_SUPP_USB2_2)—Offset 8008h	30180C01h
8010h	8013h	XECP_SUPP_USB2_3 (Full Speed) (XECP_SUPP_USB2_3)—Offset 8010h	C0021h
8014h	8017h	XECP_SUPP_USB2_4 (Low Speed) (XECP_SUPP_USB2_4)—Offset 8014h	5DC0012h
8018h	801Bh	XECP_SUPP_USB2_5 (High Speed) (XECP_SUPP_USB2_5)—Offset 8018h	1E00023h
8020h	8023h	XECP_SUPP_USB3_0 (XECP_SUPP_USB3_0)—Offset 8020h	3001402h
8024h	8027h	XECP_SUPP_USB3_1 (XECP_SUPP_USB3_1)—Offset 8024h	20425355h
8028h	802Bh	XECP_SUPP_USB3_2 (XECP_SUPP_USB3_2)—Offset 8028h	30000A0Eh
8030h	8033h	XECP_SUPP_USB3_3 (XECP_SUPP_USB3_3)—Offset 8030h	4E00121h
8034h	8037h	XECP_SUPP_USB3_4 (XECP_SUPP_USB3_4)—Offset 8034h	9C00122h
8038h	803Bh	XECP_SUPP_USB3_5 (XECP_SUPP_USB3_5)—Offset 8038h	13800123h
803Ch	803Fh	XECP_SUPP_USB3_6 (XECP_SUPP_USB3_6)—Offset 803Ch	50134h
8040h	8043h	XECP_SUPP_USB3_7 (XECP_SUPP_USB3_7)—Offset 8040h	5B10125h
8044h	8047h	XECP_SUPP_USB3_8 (XECP_SUPP_USB3_8)—Offset 8044h	B630126h
8048h	804Bh	XECP_SUPP_USB3_9 (XECP_SUPP_USB3_9)—Offset 8048h	16C60127h
8094h	9097h	Host Control Scheduler (HOST_CTRL_SCH_REG)—Offset 8094h	00000100h
80A4h	80A7h	Power Management Control (PMCTRL_REG)—Offset 80A4h	002D5090h
80B8	80BB	Super Speed Port Enable (SSPE) —Offset 80B8h	2FFh
846Ch	846Fh	USB Legacy Support Capability (USBLEGSUP)—Offset 846Ch	2201h

Table 29-2. Summary of xHCI Memory Mapped Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
84F4h	84F7h	Port Disable Override Capability Register (PDO_CAPABILITY)—Offset 84F4h	3C6h
84F8h	84FBh	USB2 Port Disable Override (USB2PDO)—Offset 84F8h	0h
84FCh	84FFh	USB3 Port Disable Override (USB3PDO)—Offset 84FCh	0h
8700h	8703h	Debug Capability ID Register (DCID)—Offset 8700h	5100Ah

29.9.1 Capability Registers Length (CAPLENGTH)—Offset 0h

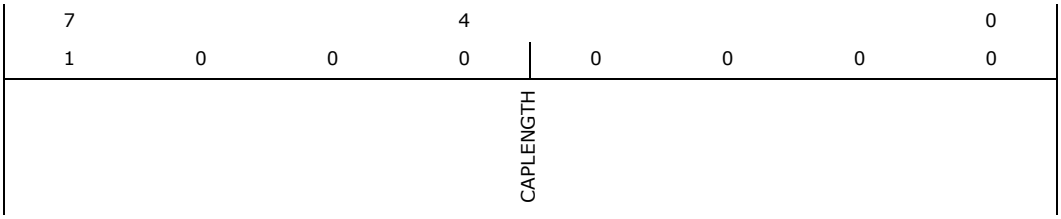
This register is modified and maintained by BIOS.

Access Method

Type: MEM Register
(Size: 8 bits)

Device: 20
Function: 0

Default: 80h



Bit Range	Default and Access	Field Name (ID): Description
7:0	80h RW/L	**Capability Registers Length (CAPLENGTH):** This register is used as an offset to add to the Memory Base Register (D29:F0:10h) to find the beginning of the Operational Register Space. This field is hardwired to 20h indicating that the Operation Registers begin at offset 20h

29.9.2 Host Controller Interface Version Number (HCIVERSION)—Offset 2h

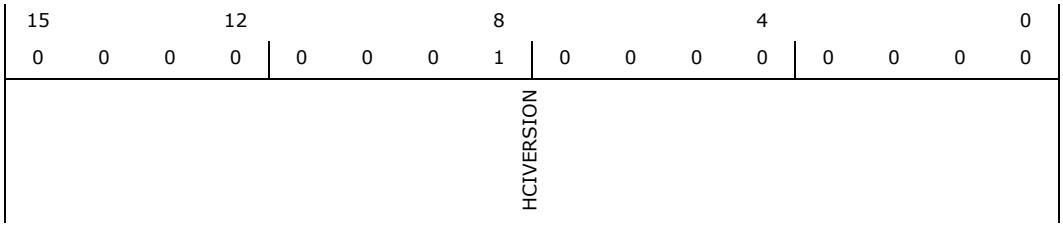
This register is modified and maintained by BIOS.

Access Method

Type: MEM Register
(Size: 16 bits)

Device: 20
Function: 0

Default: 0100h



Bit Range	Default and Access	Field Name (ID): Description
15:0	100h RO	**Host Controller Interface Version Number (HCIVERSION):** This is a two-byte register containing a BCD encoding of the version number of interface that this host controller interface conforms.

29.9.3 Structural Parameters 1 (HCSPARAMS1)—Offset 4h

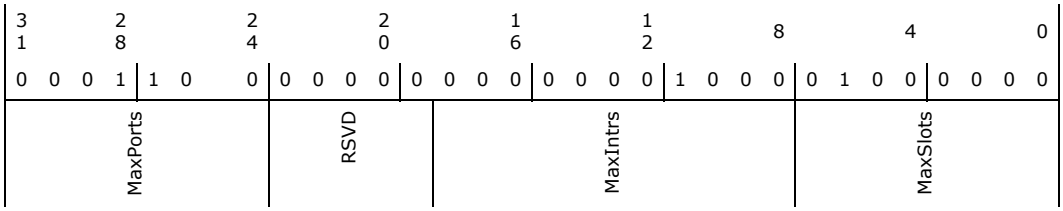
This register is modified and maintained by BIOS.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 18000840h



Bit Range	Default and Access	Field Name (ID): Description
31:24	18h RO	**Number of Ports (MaxPorts):** This field specifies the number of physical downstream ports implemented on this host controller. The value of this field determines how many port registers are addressable in the Operational Register Space.
23:19	0h RO	Reserved.
18:8	8h RW/L	**Number of Interrupters (MaxIntrs):** This field specifies the number of interrupters implemented on this host controller. Each interrupter is allocated to a vector of MSI and controls its generation and moderation.
7:0	40h RW/L	**Number of Device Slots (MaxSlots):** This field specifies the number of Device Context Structures and Doorbell Array entries this host controller can support. Valid values are in the range of 1 to 255.

29.9.4 Structural Parameters 2 (HCSPARAMS2)—Offset 8h

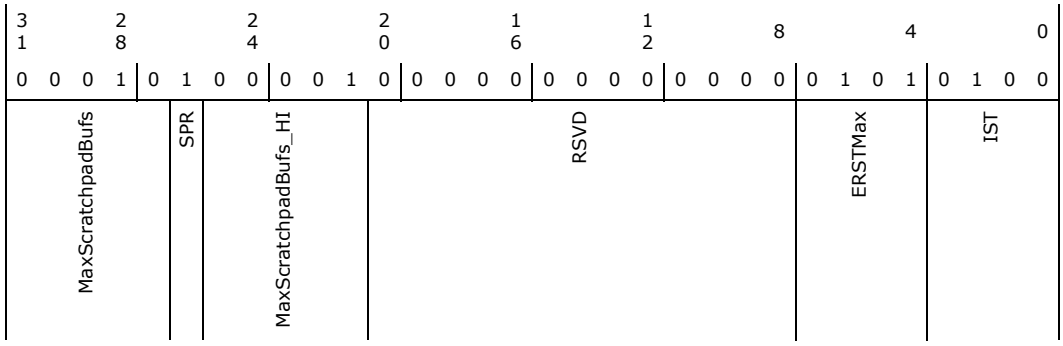
This register is modified and maintained by BIOS.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 14200054h



Bit Range	Default and Access	Field Name (ID): Description
31:27	2h RW/L	**Max Scratchpad Buffers LO (MaxScratchpadBufs):** Indicates the number of Scratchpad Buffers system software shall reserve for the xHC.
26	1h RW/L	**Scratchpad Restore (SPR):** 0 = Indicates the Scratchpad buffer space may be freed and reallocated between power events. 1 = Indicates that the xHC requires the integrity of the Scratchpad buffer space to be maintained across power events.
25:21	1h RW/L	**Max Scratchpad Buffers HI (MaxScratchpadBufs_HI)**

Bit Range	Default and Access	Field Name (ID): Description
20:8	0h RO	Reserved.
7:4	5h RW/L	**Event Ring Segment Table Max (ERSTMax):** This field determines the maximum value supported by the Event Ring Segment Table Base Size registers.
3:0	4h RW/L	**Isochronous Scheduling Threshold (IST):** This field indicates to system software the minimum distance (in time) that it is required to stay ahead of the xHC while adding TRBs, in order to have the xHC process them at the correct time. The value is specified in the number if frames/ microframes. If bit [3] of IST is cleared to 0b, software can add a TRB no later than IST [2:0] microframes before that TRB is scheduled to be executed. If bit [3] of IST is set to 1b, software can add a TRB no later than IST[2:0] frames before that TRB is scheduled to be executed.

29.9.5 Structural Parameters 3 (HCSPARAMS3)—Offset Ch

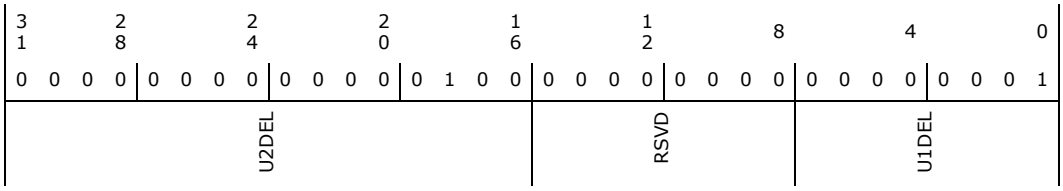
This register is modified and maintained by BIOS.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 40001h



Bit Range	Default and Access	Field Name (ID): Description
31:16	4h RW/L	**U2 Device Exit Latency (U2DEL):** Indicates the worst case latency to transition from U2 to U0. Applies to all root hub ports. The following are permissible values: Value Description 00h Zero 01h Less than 1 µs 02h Less than 2 µs ... 0Bh-FFh Reserved
15:8	0h RO	Reserved.
7:0	1h RW/L	**U1 Device Exit Latency (U1DEL):** Worst case latency to transition a root hub Port Link State (PLS) from U1 to U0. Applies to all root hub ports. The following are permissible values: Value Description 00h Zero 01h Less than 1 µs 02h Less than 2 µs ... 0800h-FFFFh Reserved

29.9.6 Capability Parameters (HCCPARAMS)—Offset 10h

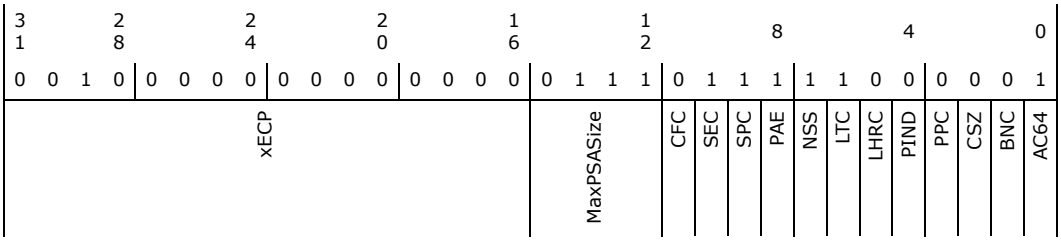
This register is modified and maintained by BIOS.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 200077C1h



Bit Range	Default and Access	Field Name (ID): Description
31:16	2000h RW/L	**xHCI Extended Capabilities Pointer (xECP):** This field indicates the existence of a capabilities list. The value of this field indicates a relative offset, in 32-bit words, from Base to the beginning of the first extended capability.
15:12	7h RW/L	**Maximum Primary Stream Array Size (MaxPSASize):** RW/L. This fields identifies the maximum size Primary Stream Array that the xHC supports. The Primary Stream Array size = 2MaxPSASize+1. Valid MaxPSASize values are 1 to 15.
11	0h RW/L	**Contiguous Frame ID Capability (CFC)**
10	1h RW/L	**Stopped EDLTA Capability (SEC):** This flag indicates that the host controller implementation Stream Context support a Stopped EDLTA field.
9	1h RW/L	**Stopped - Short Packet Capability (SPC):** This flag indicates that the host controller implementation is capable of generating a Stopped-Short Packet Completion Code.
8	1h RW/L	**Parst All Event Data (PAE)**
7	1h RW/L	**No Secondary SID Support (NSS):** Hardwired to '0' indicating Secondary Stream ID decoding is supported.
6	1h RW/L	**Latency Tolerance Messaging Capability (LTC):** 0 = Latency Tolerance Messaging is not supported. 1 = Latency Tolerance Messaging is supported
5	0h RW/L	**Light HC Reset Capability (LHRC):** 0 = Light Host Controller Reset is not supported. 1 = Light Host Controller Reset is supported
4	0h RW/L	**Port Indicators (PIND):** This bit indicates whether the xHC root hub ports support port indicator control. When this bit is a '1', the port status and control registers include a read/writeable field for controlling the state of the port indicator.
3	0h RW/L	**Port Power Control (PPC):** This bit indicates whether the host controller implementation includes port power control. A '1' in this bit indicates the ports have port power switches. A '0' in this bit indicates the port do not have port power switches.

Bit Range	Default and Access	Field Name (ID): Description
2	0h RW/L	**Context Size (CSZ):** If this bit is set to '1', then the xHC uses 64 byte Context data structures. If this bit is cleared to '0', then the xHC uses 32 byte Context data structures.
1	0h RW/L	**BW Negotiation Capability (BNC):** 0 = Not capable of BW Negotiation. 1 = Capable of BW Negotiation.
0	1h RW/L	**64-bit Addressing Capability (AC64):** This bit documents the addressing range capability of the xHC. The value of this flag determines whether the xHC has implemented the high order 32- bits of 64-bit register and data structure pointer fields. Values for this flag have the following interpretation: 0 = Supports 32-bit address memory pointers 1 = Supports 64-bit address memory pointers If 32-bit address memory pointers are implemented, the xHC shall ignore the high order 32- bits of 64-bit data structure pointer fields, and system software shall ignore the high order 32- bits of 64- bit xHC registers.

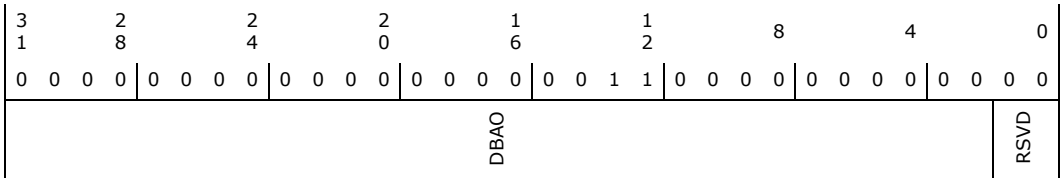
29.9.7 Doorbell Offset (DBOFF)—Offset 14h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 3000h



Bit Range	Default and Access	Field Name (ID): Description
31:2	C00h RO	**Doorbell Array Offset (DBAO):** This field defines the DWord offset of the Doorbell Array base address from the Base (for example, the base address of the xHCI Capability register address space).
1:0	0h RO	Reserved.

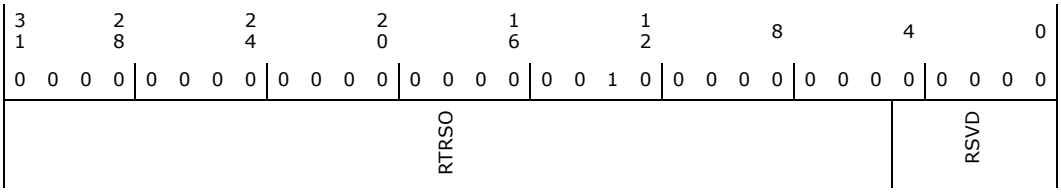
29.9.8 Runtime Register Space Offset (RTSOFF)—Offset 18h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 2000h



Bit Range	Default and Access	Field Name (ID): Description
31:5	100h RO	**Runtime Register Space Offset (RTRSO):** This field defines the 32-byte offset of the xHCI Runtime Registers from the Base. That is, Runtime Register Base Address = Base + Runtime Register Set Offset.
4:0	0h RO	Reserved.

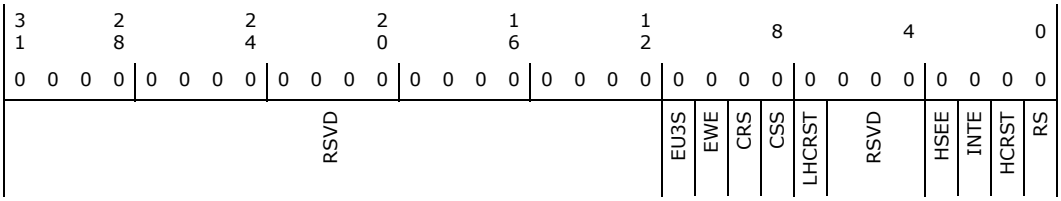
29.9.9 USB Command (USBCMD)—Offset 80h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:12	0h RO	Reserved.
11	0h RW	**Enable U3 MFINDEX Stop (EU3S):** When set to 1b, the xHC may stop the MFINDEX counting action if all Root Hub ports are in the U3, Disconnected, Disabled, or Powered-off state. When cleared to 0b, the xHC may stop the MFINDEX counting action if all Root Hub ports are in the Disconnected, Disabled, or Powered-off state.
10	0h RW	**Enable Wrap Event (EWE):** When set to 1b, the xHC shall generate a MFINDEX Wrap Event every time the MFINDEX register transitions from 03FFFh to 0. When cleared to 0b, no MFINDEX Wrap Events are generated.

Bit Range	Default and Access	Field Name (ID): Description
9	0h RW	**Controller Restore State (CRS):** When set to 1b, MEM_BASE+80h:bit 0= 0b, and MEM_BASE+80h:bit 8 = 1b, the xHC shall perform a Restore State operation and restore its internal state. When set to 1b and MEM_BASE+80h:bit 0= 1b or MEM_BASE+80h:bit 8 = 0b, or when cleared to '0', no Restore State operation shall be performed.
8	0h RW	**Controller Save State (CSS):** When written by software with 1b and MEM_BASE+80h:bit 0=0b, the xHC shall save any internal state that will be restored by a subsequent Restore State operation. When written by software with 1b and MEM_BASE+80h:bit 0= 1b, or written with '0', no Save State operation shall be performed.
7	0h RW	**Light Host Controller Reset (LHCRST):** If the Light HC Reset Capability (LHRC) bit (MEM_BASE=10h:bit 5) is 1b, then setting this bit to 1b allows the driver to reset the xHC without affecting the state of the ports. A system software read of this bit as 0b indicates the Light Host Controller Reset has completed and it is safe for software to re-initialize the xHC. A software read of this bit as a 1b indicates the Light Host Controller Reset has not yet completed.
6:4	0h RO	Reserved.
3	0h RW	**Host System Error Enable (HSEE):** When this bit is set to 1b, and the HSE bit (MEM_BASE+84h:bit 2) is set to 1b, the xHC shall assert out-of-band error signaling to the host. The signaling is acknowledged by software clearing the HSE bit.
2	0h RW	**Interrupter Enable (INTE):** This control bit is used by software to reset the host controller. When software sets this bit to 1b, the Host Controller resets its internal pipelines, timers, counters, state machines, and so forth, to their initial value. Any transaction currently in progress on USB is immediately terminated. A USB reset is not driven on downstream ports. PCI Configuration registers are not affected by this reset. All operational registers, including port registers and port state machines are set to their initial values.
1	0h RW	**Host Controller Reset (HCRST):** This control bit is used by software to reset the host controller. When software sets this bit to 1b, the Host Controller resets its internal pipelines, timers, counters, state machines, and so forth, to their initial value. Any transaction currently in progress on USB is immediately terminated. A USB reset is not driven on downstream ports. PCI Configuration registers are not affected by this reset. All operational registers, including port registers and port state machines are set to their initial values.
0	0h RW	**Run/Stop (RS):** When set to 1b, the xHC proceeds with execution of the schedule. The xHC continues execution as long as this bit is set to 1b. When this bit is cleared to 0b, the xHC completes the current and any actively pipelined transactions on the USB and then halts. The xHC shall halt within 16 microframes after software clears the Run/ Stop bit. The HCHalted (HCH) bit (MEM_BASE+84h:bit 0) indicates when the xHC has finished its pending pipelined transactions and has entered the stopped state. Software shall not write a '1' to this flag unless the xHC is in the Halted state (that is, HCH in the USBSTS register is '1'); doing so will yield undefined results.

29.9.10 USB Status (USBSTS)—Offset 84h

Offset: MEM_BASE + 84h–87h Attribute: R/WC, RO
Default Value: 00000001h Size: 32 bits
This register indicates pending interrupts and various states of the Host controller. The status resulting from a transaction on the serial bus is not indicated in this register. Seethe Interrupts description in Section 4 of the *xHCI Specification* for additional information concerning interrupt conditions.

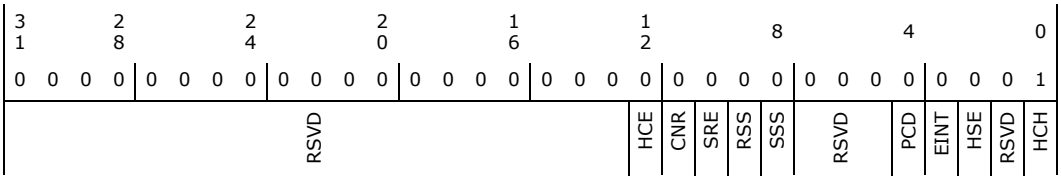
Note: For the writable bits, software must write a 1 to clear bits that are set. Writing a 0 has no effect.

Access Method

Type: MEM Register (Size: 32 bits)

Device: 20
Function: 0

Default: 1h



Bit Range	Default and Access	Field Name (ID): Description
31:13	0h RO	Reserved.
12	0h RO	**Host Controller Error (HCE):** This flag shall be set to indicate that an internal error condition has been detected which requires software to reset and re-initialize the xHC. 0 = No internal xHC error conditions exist. 1 = Internal xHC error condition exists.
11	0h RO	**Controller Not Ready (CNR):** 0 = Ready 1 = Not Ready Software shall not write any Doorbell or Operational register of the xHC, other than the USBSTS register, until CNR = 0b. This flag is set by the xHC after a Hardware Reset and cleared when the xHC is ready to begin accepting register writes. This flag shall remain cleared (0b) until the next Chip Hardware Reset.
10	0h RW/C	**Save/Restore Error (SRE):** If an error occurs during a Save or Restore operation this bit shall be set to 1b. This bit shall be cleared to 0b when a Save or Restore operation is initiated or when written with 1b.
9	0h RO	**Restore State Status (RSS):** When the Controller Restore State (CRS) flag in the USB_CMDregister is written with 1b this bit shall be set to 1b and remain set while the xHC restores its internal state. ***Note:*** When the Restore State operation is complete, this bit shall be cleared to 0b.
8	0h RO	**Save State Status (SSS)**
7:5	0h RO	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
4	0h RW/1C	**Port Change Detect (PCD):** This bit is allowed to be maintained in the Auxiliary power well. Alternatively, it is also acceptable that on a D3 to D0 transition of the xHC, this bit is loaded with the OR of all of the PORTSC change bits (including: Force port resume, overcurrent change, enable/ disable change and connect status change). Regardless of the implementation, when this bit is readable (that is, in the D0 state), it must provide a valid view of the Port Status registers. 0 = No change bit transition from a 0 to 1 or No Force Port Resume bit transition from 0 to 1 as a result of a J-K transition detected on a suspended port. 1 = The Host controller sets this bit to 1 when any port for which the Port Owner bit is set to 0 has a change bit transition from a 0 to 1 or a Force Port Resume bit transition from 0 to 1 as a result of a J-K transition detected on a suspended port.
3	0h RW/1C	**Event Interrupt (EINT):** The xHC sets this bit to 1b when the Interrupt Pending (IP) bit of any Interrupter is transitions from 0b to 1b. Software that uses EINT shall clear it prior to clearing any IP flags. A race condition will occur if software clears the IP flags then clears the EINT flag, and between the operations another IP '0' to '1' transition occurs. In this case the new IP transition will be lost.
2	0h RW/1C	**Host System Error (HSE):** The xHC sets this bit to 1b when a serious error is detected, either internal to the xHC or during a host system access involving the xHC module. Conditions that set this bit to '1' include PCI Parity error, PCI Master Abort, and PCI Target Abort. When this error occurs, the xHC clears the Run/Stop (R/S) bit in the USB_CMD register to prevent further execution of the scheduled TDs. If the HSEE bit in the USB_CMD register is 1b, the xHC shall also assert out-of-band error signaling to the host.
1	0h RO	Reserved.
0	1h RO	**HCHalted (HCH):** This bit is a '0' whenever the Run/Stop (R/S) bit is set to 1b. The xHC sets this bit to 1b after it has stopped executing as a result of the Run/Stop (R/S) bit being cleared to 0b, either by software or by the xHC hardware (for example, internal error). If this bit is set to1b, then SOFs, microSOFs, or Isochronous Timestamp Packets (ITP) shall not be generated by the xHC.

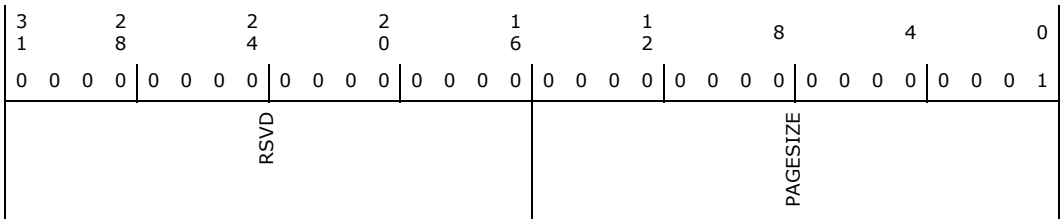
29.9.11 Page Size (PAGESIZE)—Offset 88h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 1h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:0	1h RO	**Page Size (PAGESIZE):** Hardwired to 1h to indicate support for 4 Kbyte page sizes.

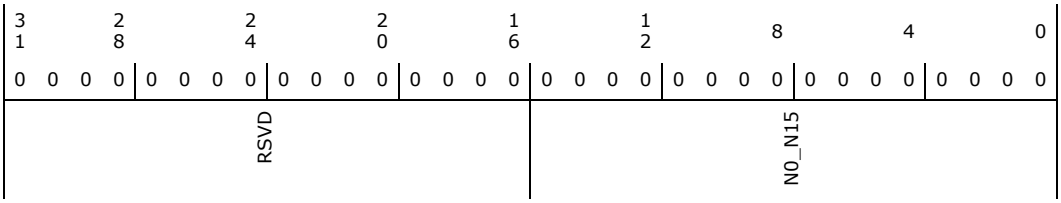
29.9.12 Device Notification Control (DNCTRL)—Offset 94h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:0	0h RW	**Notification Enable (N0_N15):** When a Notification Enable bit is set, a Device Notification Event will be generated when a Device Notification Transaction Packet is received with the matching value in the Notification Type field. For example, setting N1 to '1' enables Device Notification Event generation if a Device Notification TP is received with its Notification Type field set to '1' (FUNCTION_WAKE), and so on

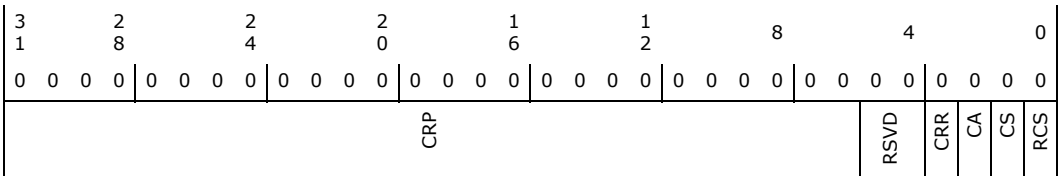
29.9.13 Command Ring Low (CRCR_LO)—Offset 98h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:6	0h RW	**Command Ring Pointer (CRP):** This field defines low order bits of the initial value of the 64-bit Command Ring Dequeue Pointer. ***Notes:*** 1. Writes to this field are ignored when Command Ring Running bit (CRR) = 1b. 2. If the CRCR register is written while the Command Ring is stopped (CRR = 0b), the value of this field shall be used to fetch the first Command TRB the next time the Host Controller Doorbell register is written with the DB Reason field set to Host Controller Command. 3. If the CRCR register is not written while the Command Ring is stopped (CRR = 0b), then the Command Ring shall begin fetching Command TRBs at the current value of the internal xHC Command Ring Dequeue Pointer. 4. Reading this field always returns 0b.

Bit Range	Default and Access	Field Name (ID): Description
5:4	0h RO	Reserved.
3	0h RO	**Command Ring Running (CRR):** This bit is set to 1b if the Run/Stop (R/S) bit is 1b and the Host Controller Doorbell register is written with the DB Reason field set to Host Controller Command. It is cleared to 0b when the Command Ring is stopped after writing a 1b to the Command Stop (CS) or Command Abort (CA) bits, or if the R/S bit is cleared to 0b.
2	0h RW/1S	**Command Abort (CA):** Writing a 1b to this bit shall immediately terminate the currently executing command, stop the Command Ring, and generate a Command Completion Event with the Completion Code set to Command Ring Stopped. The next write to the Host Controller Doorbell with DB Reason field set to Host Controller Command shall restart the Command Ring operation. ***Notes:*** 1. Writes to this flag are ignored by the xHC if Command Ring Running (CRR) = 0b. 2. Reading this bit always returns 0b.
1	0h RW/1S	**Command Stop (CS):** Writing a 1b to this bit shall stop the operation of the Command Ring after the completion of the currently executing command, and generate a Command Completion Event with the Completion Code set to Command Ring Stopped and the Command TRB Pointer set to the current value of the Command Ring Dequeue Pointer. The next write to the Host Controller Doorbell with DB Reason field set to Host Controller Command shall restart the Command Ring operation. ***Notes:*** 1. Writes to this flag are ignored by the xHC if Command Ring Running (CRR) bit = 0b. 2. Reading this bit always returns 0b.
0	0h RW	**Ring Cycle State (RCS):** This bit identifies the value of the xHC Consumer Cycle State (CCS) flag for the TRB referenced by the Command Ring Pointer. ***Notes:*** 1. Writes to this bit are ignored when the Command Ring Running (CRR) bit = 1b. 2. If the CRCR register is written while the Command Ring is stopped (CCR = 0b), then the value of this flag shall be used to fetch the first Command TRB the next time the Host Controller Doorbell register is written with the DB Reason field set to Host Controller Command. 3. If the CRCR register is not written while the Command Ring is stopped (CCR = 0b), then the Command Ring will begin fetching Command TRBs using the current value of the internal Command Ring CCS flag. 4. Reading this flag always returns 0b.

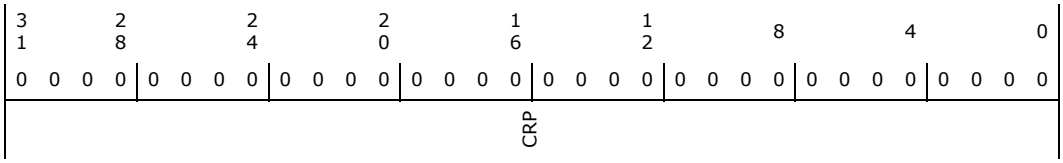
29.9.14 Command Ring High (CRCR_HI)—Offset 9Ch

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Command Ring Pointer (CRP):** Command Ring Pointer—R/W. This field defines high order bits of the initial value of the 64-bit Command Ring Dequeue Pointer. ***Notes:*** 1. Writes to this field are ignored when Command Ring Running bit (CRR) = 1b. 2. If the CRCR register is written while the Command Ring is stopped (CRR = 0b), the value of this field shall be used to fetch the first Command TRB the next time the Host Controller Doorbell register is written with the DB Reason field set to Host Controller Command. 3. If the CRCR register is not written while the Command Ring is stopped (CRR = 0b), then the Command Ring shall begin fetching Command TRBs at the current value of the internal xHC Command Ring Dequeue Pointer. 4. Reading this field always returns 0b.

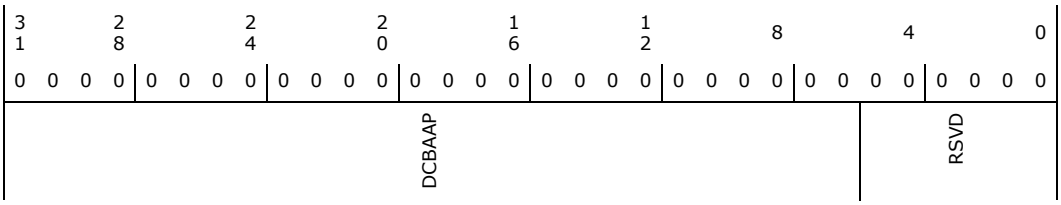
29.9.15 Device Context Base Address Array Pointer Low (DCBAAP_LO)—Offset B0h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:6	0h RW	**Device Context Base Address Array Pointer (DCBAAP):** This field defines low order bits of the 64-bit base address of the Device Context Pointer Array table (a table of address pointers that reference Device Context structures for the devices attached to the host.
5:0	0h RO	Reserved.

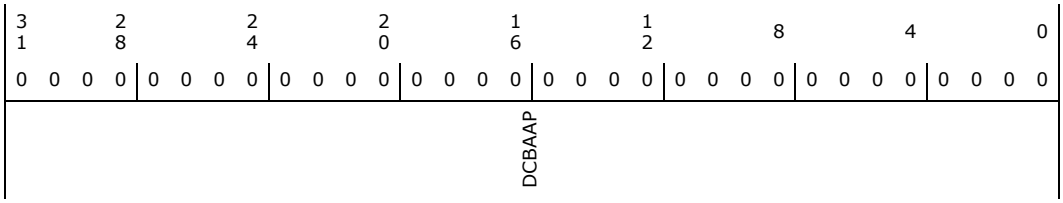
29.9.16 Device Context Base Address Array Pointer High (DCBAAP_HI)—Offset B4h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Device Context Base Address Array Pointer (DCBAAP):** This field defines high order bits of the 64-bit base address of the Device Context Pointer Array table (a table of address pointers that reference Device Context structures for the devices attached to the host.)

29.9.17 Port N Status and Control USB2 (PORTSCN)—Offset 480h

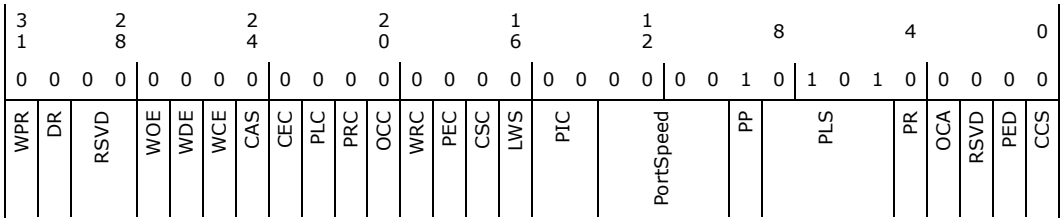
There are NumUSB2 USB2 PORTSC registers at offsets:
480h, 490h, 4A0h,... (480h + (NumUSB2-1)*10h)

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 2A0h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RW/1S	**Warm Port Reset (WPR):** When software sets this bit to 1b, the Warm Reset sequence is
30	0h RW/L	**Device Removable (DR):** This bit indicates if this port has a removable device. 0 = Device is removable. 1 = Device is non-removable.
29:28	0h RO	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
27	0h RW	**Wake on Over-current Enable (WOE):** 0 = Disable. (Default) 1 = Enable. Writing this bit to a 1b enables the port to be sensitive to over-current conditions as system wake-up events.
26	0h RW	**Wake on Disconnect Enable (WDE):** 0 = Disable. (Default) 1 = Enable. Writing this bit to a 1b enables the port to be sensitive to device disconnects as system wake-up events.
25	0h RW	**Wake on Connect Enable (WCE):** 0 = Disable. (Default) 1 = Enable. Writing this bit to a 1b enables the port to be sensitive to device connects as system wake-up events.
24	0h RO	**Cold Attach Status (CAS):** This bit indicates that far-end terminations were detected in the Disconnected state and the Root Hub Port State Machine was unable to advance to the Enabled state. Software shall clear this bit by writing a 1b to the WPR bit or the xHC shall clear this bit if the CSS bit transitions to 1.
23	0h RW/1C	**Port Config Error Change (CEC):** ***Note:*** This register is sticky.
22	0h RW/1C	**Port Link State Change (PLC):** 0 = No change 1 = Link Status Change This flag is set to '1' due to the following Port Link State (PLS) transitions:
21	0h RW/1C	**Port Reset Change (PRC):** This flag is set to '1' due a '1' to '0' transition of Port Reset (PR). for example, when any reset processing on this port is complete. 0 = No change 1 = Reset Complete
20	0h RW/1C	**Over-current Change (OCC):** The functionality of this bit is not dependent upon the port owner. Software clears this bit by writing a 1 to it. 0 = No change. (Default) 1 = There is a change to Overcurrent Active.
19	0h RW/1C	**Warm Port Reset Change (WRC):** This bit is set when Warm Reset processing on this port completes. 0 = No change. (Default) 1 = Warm reset complete
18	0h RW/1C	**Port Enabled Disabled Change (PEC):** 0 = No change. (Default) 1 = There is a change to PED bit.
17	0h RW/1C	**Connect Status Change (CSC):** R/WC. This flag indicates a change has occurred in the port's Current Connect Status (CCS) or Cold Attach Status (CAS) bits. 0 = No change. (Default) 1 = There is a change to the CCS or CAS bit. The xHC sets this bit to 1b for all changes to the port device connect status, even if system software has not cleared an existing Connect Status Change. For example, the insertion status changes twice before system software has cleared the changed condition, root hub hardware will be "setting" an already-set bit (that is, the bit will remain 1b).
16	0h RW	**Port Link State Write Strobe (LWS):** 0 = When 0b, write data in PLS field is ignored. (Default) 1 = When this bit is set to 1b on a write reference to this register, this flag enables writes to the PLS field. Reads to this bit return '0'.
15:14	0h RW	**Port Indicator Control (PIC):** ***Note:*** This register is sticky.
13:10	0h RW	**Port Speed (PortSpeed):** A device attached to this port operates at a speed defined by the following codes: Value Speed 0001b Full-speed 0010b Low speed 0011b Highspeed All other values reserved. Refer to the eXtensible Host Controller Interface for Universal Serial Bus Specification for additional details.
9	1h RW	**Port Power (PP):** Read-only with a value of 1. This indicates that the port does have power.

Bit Range	Default and Access	Field Name (ID): Description
8:5	5h RW	**Port Link State (PLS):** This field is used to power manage the port and reflects its currentlink state.When the port is in the Enabled state, system software may set the link U-state by writing this field. System software may also write this field to force a Disabled to Disconnected state transition of the port. Write Value Description 0 The link shall transition to a U0 state from any of the U-states. 2 USB 2.0 ports only. The link should transition to the U2 State. 3 The link shall transition to a U3 state from any of the U-states. This action selectively suspends the device connected to this port. While the Port LinkState = U3, the hub does not propagate downstream-directed traffic to this port, but the hub will respond to resume signaling from the port. 5 USB 3.0 ports only. If the port is in the Disabled state (PLS = Disabled, PP= 1), then the link shall transition to a RxDetect state and the port shall transition to the Disconnected state, else ignored. 15 USB 2.0 ports only. If the port is in the U3 state (PLS = U3), then the link shall remain in the U3 state and the port shall transition to the U3Exit substate, else ignored. All other values Ignored Read Value Definition 0 Link is in the U0 State 1 Link is in the U1 State 2 Link is in the U2 State 3 Link is in the U3 State (Device Suspended) 4 Link is in the Disabled State 5 Link is in the RxDetect State 6 Link is in the Inactive State 7 Link is in the Polling State 8 Link is in the Recovery State 9 Link is in the Hot Reset State 10 Link is in the Compliance Mode State 11 Link is in the Test Mode State 12–14 Reserved 15 Link is in the Resume State
4	0h RW/1S	**Port Reset (PR):** When software writes a 1 to this bit (from a 0), the bus reset sequence as 1=port in reset 0=port not in reset
3	0h RW	**Over-current Active (OCA):** 0 = This port does not have an overcurrent condition. (Default) 1 = This port currently has an overcurrent condition. This bit will automatically transition from 1 to 0 when the over current condition is removed. The PCH automatically disables the port when the overcurrent active bit is 1.
2	0h RO	Reserved.
1	0h RW/C	**Port Enabled Disabled (PED):** Ports can only be enabled by the host controller as a part of the reset and enable. Software cannot enable a port by writing a 1 to this bit. Ports can be disabled by either a fault condition (disconnect event or other fault condition) or by host software. The bit status does not change until the port state actually changes. There may be a delay in disabling or enabling a port due to other host controller and bus events. 0=disable 1=enable(default)
0	0h RW	**Current Connect Status (CCS):** This value reflects the current state of the port, and may not correspond directly to the event that caused the Connect Status Change bit (Bit 1) to be set. 0=no device is present 1=device is present on port.

29.9.18 Port Power Management Status and Control USB2 (PORTPMSCN)—Offset 484h

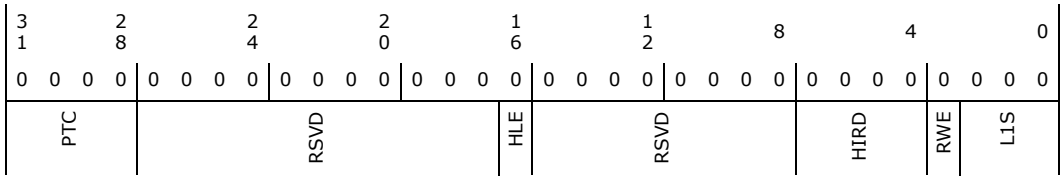
Address Offset: Port 1 484-487h
Default Value: 00000000h
Access: RO; RW; RWS;
Size: 32 bits
There are 6 USB2 PORTPMSC registers at offsets:
484h, 494h, ... (484h + (NumUSB2-1)*10h)

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:28	0h RW	**Port Test Control (PTC):** When this field is '0', the port is not operating in a test mode. (Default) A non-zero value indicates that the port is operating in test mode and the specific test mode is indicated by the specific value. A non-zero Port Test Control value is only valid to a port that is in the Disabled state. If the port is not in this state, the xHC shall respond with the Port Test Control field set to Port Test Control Error. The encoding of the Test Mode bits for a USB 2.0 port are: Value Test Mode 0h Test mode not enabled 1h Test J_STATE 2h Test K_STATE 3h Test SE0_NAK 4h Test Packet 5h Test FORCE_ENABLE 6h–14h Reserved. 15 Port Test Control Error
27:17	0h RO	Reserved.
16	0h RW	**Hardware LPM Enable (HLE):** 0=disable 1=Enable. When this bit is a 1, hardware controlled LPM shall be enabled for this port. Refer to Section 4 of the *USB 2.0 LPM Specification* for more information.
15:8	0h RO	Reserved.
7:4	0h RW	**Host Initiated Resume Duration (HIRD):** ***Note:*** This register is sticky.
3	0h RW	**Remote Wake Enable (RWE):** The host system sets this flag to enable or disable the device for remote wake from L1. 0=disable 1=enable The value of this flag will temporarily (while in L1) override the current setting of the Remote Wake feature set by the standard Set/ClearFeature() commands defined in Universal Serial Bus Specification, revision 2.0, Chapter 9.
2:0	0h RW	**L1 Status (L1S):** ***Note:*** This register is sticky.

29.9.19 Port N Hardware LPM Control Register (PORTHLPMCN)—Offset 48Ch

There are 9 PORTHLPMC registers at offsets 48Ch, 49Ch, 4ACh, 4BCh, 4CCh, 4DCh, 4ECh, 4FCh, 50Ch.This register is reset only by platform hardware during cold reset or in response to a Host Controller Reset (HCRST).The definition for the fields depend on the protocol supported. For USB3 this register is reserved and shall be treated by software as RsvdP. For USB2 the definition is given below. Fields contain parameters necessary for xHC to automatically generate an LPM Token to the downstream device.

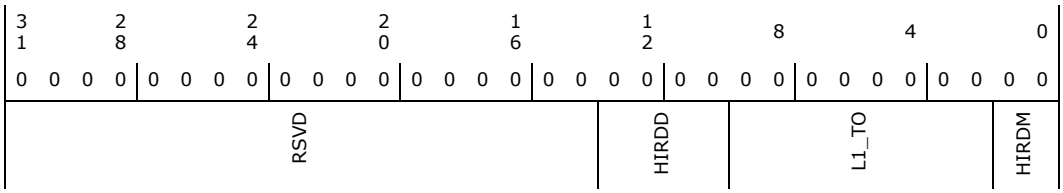
Access Method

Type: MEM Register (Size: 32 bits)

Device: 20

Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:10	0h RW	**Host Initiated Resume Duration-Deep (HIRDD):** System software sets this field to indicate to the recipient device how long the xHC will drive resume if an exit from L1. The HIRDD value is is encoded as follows: 0h: 50 us (default) 1h: 125 us 2h: 200 us...Fh: 1.175ms The value of 0h is interpreted as 50 us. Each incrementing value adds 75 us to the previous value.
9:2	0h RW	**L1 Timeout (L1_TO):** Timeout value for L1 inactivity timer. This field shall be set to 00h by assertion of PR to '1'. Following are permissible values: 00h: 128 us (default) 01h: 256 us....FFh: 65,280us
1:0	0h RW	**Host Initiated Resume Duration Mode (HIRDM):** Indicates which HIRD value should be used. Following are permissible values: 0: Initiate L1 using HIRD only time out (default) 1: Initiate HIRDDon timeout. If rejected by device, initiate L1 using HIRD 2,3: Reserved ***Note:*** This register is sticky.

29.9.20 Port Status and Control USB3 (PORTSCXUSB3)—Offset 580h

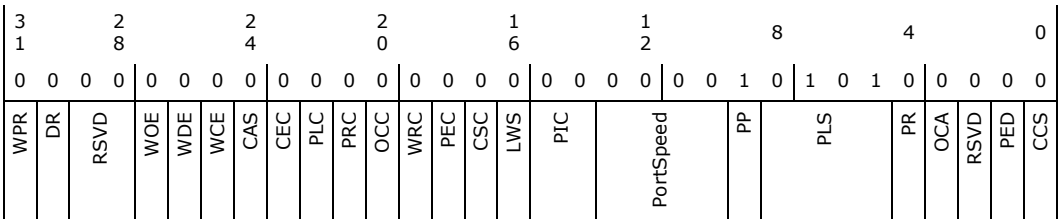
There are USB3 port status and control registers at offsets:
580h, 590h,5A0h,... (580h + (#of USB3 ports -1)*10h)

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 000002A0h



Bit Range	Default and Access	Field Name (ID): Description
31	0b RW/1S	**Warm Port Reset (WPR):** When software sets this bit to 1b, the Warm Reset sequence is enabled.
30	0b RW/L	**Device Removable (DR):** This bit indicates if this port has a removable device. 0 = Device is removable. 1 = Device is non-removable.
29:28	00b RO	Reserved.
27	0b RW	**Wake on Over-current Enable (WOE):** 0 = Disable. (Default) 1 = Enable. Writing this bit to a 1b enables the port to be sensitive to over-current conditions as system wake-up events.
26	0b RW	**Wake on Disconnect Enable (WDE):** 0 = Disable. (Default) 1 = Enable. Writing this bit to a 1b enables the port to be sensitive to device disconnects as system wake-up events.
25	0b RW	**Wake on Connect Enable (WCE):** 0 = Disable. (Default) 1 = Enable. Writing this bit to a 1b enables the port to be sensitive to device connects as system wake-up events.
24	0b RO	**Cold Attach Status (CAS):** This bit indicates that far-end terminations were detected in the Disconnected state and the Root Hub Port State Machine was unable to advance to the Enabled state. Software shall clear this bit by writing a 1b to the WPR bit or the xHC shall clear this bit if the CSS bit transitions to 1.
23	0b RW/1C	**Port Config Error Change (CEC):** ***Note:*** This register is sticky.
22	0b RW/1C	**Port Link State Change (PLC):** 0 = No change 1 = Link Status Change
21	0b RW/1C	**Port Reset Change (PRC):** This flag is set to ‘1’ due a '1' to '0' transition of Port Reset (PR). for example, when any reset processing on this port is complete. 0 = No change 1 = Reset Complete

Bit Range	Default and Access	Field Name (ID): Description
20	0b RW/1C	**Over-current Change (OCC):** The functionality of this bit is not dependent upon the port owner. Software clears this bit by writing a 1 to it. 0 = No change. (Default) 1 = There is a change to Overcurrent Active.
19	0b RW/1C	**Warm Port Reset Change (WRC):** This bit is set when Warm Reset processing on this port completes. 0 = No change. (Default) 1 = Warm reset complete
18	0b RW/1C	**Port Enabled Disabled Change (PEC):** 0 = No change. (Default) 1 = There is a change to PED bit.
17	0b RW/1C	**Connect Status Change (CSC):** R/WC. This flag indicates a change has occurred in the port's Current Connect Status (CCS) or Cold Attach Status (CAS) bits. 0 = No change. (Default) 1 = There is a change to the CCS or CAS bit. The xHC sets this bit to 1b for all changes to the port device connect status, even if system software has not cleared an existing Connect Status Change. For example, the insertion status changes twice before system software has cleared the changed condition, root hub hardware will be "setting" an already-set bit (that is, the bit will remain 1b).
16	0b RW	**Port Link State Write Strobe (LWS):** 0 = When 0b, write data in PLS field is ignored. (Default) 1 = When this bit is set to 1b on a write reference to this register, this flag enables writes to the PLS field. Reads to this bit return '0'.
15:14	00b RW	**Port Indicator Control (PIC):** ***Note:*** This register is sticky.
13:10	0h RW	**Port Speed (PortSpeed):** A device attached to this port operates at a speed defined by the following codes: Value Speed 0100b: Super Speed.
9	1b RW	**Port Power (PP):** Read-only with a value of 1. This indicates that the port does have power.
8:5	5h RW	**Port Link State (PLS):** This field is used to power manage the port and reflects its currentlink state.When the port is in the Enabled state, system software may set the link U-state by writing this field. System software may also write this field to force a Disabled to Disconnected state transition of the port. Write Value Description 0 The link shall transition to a U0 state from any of the U-states. 2 USB 2.0 ports only. The link should transition to the U2 State. 3 The link shall transition to a U3 state from any of the U-states. This action selectively suspends the device connected to this port. While the Port LinkState = U3, the hub does not propagate downstream-directed traffic to this port, but the hub will respond to resume signaling from the port. 5 USB 3.0 ports only. If the port is in the Disabled state (PLS = Disabled, PP= 1), then the link shall transition to a RxDetect state and the port shall transition to the Disconnected state, else ignored. 15 USB 2.0 ports only. If the port is in the U3 state (PLS = U3), then the link shall remain in the U3 state and the port shall transition to the U3Exit substate, else ignored. All other values Ignored Read Value Definition 0 Link is in the U0 State 1 Link is in the U1 State 2 Link is in the U2 State 3 Link is in the U3 State (Device Suspended) 4 Link is in the Disabled State 5 Link is in the RxDetect State 6 Link is in the Inactive State 7 Link is in the Polling State 8 Link is in the Recovery State 9 Link is in the Hot Reset State 10 Link is in the Compliance Mode State 11 Link is in the Test Mode State 12–14 Reserved 15 Link is in the Resume State
4	0b RW/1S	**Port Reset (PR):** When software writes a 1 to this bit (from a 0), the bus reset sequence as 1=port in reset 0=port not in reset

Bit Range	Default and Access	Field Name (ID): Description
3	0b RW	**Over-current Active (OCA):** 0 = This port does not have an overcurrent condition. (Default) 1 = This port currently has an overcurrent condition. This bit will automatically transition from 1 to 0 when the over current condition is removed. The PCH automatically disables the port when the overcurrent active bit is 1.
2	0b RO	Reserved.
1	0h RW/1C	**Port Enabled Disabled (PED):** Ports can only be enabled by the host controller as a part of the reset and enable. Software cannot enable a port by writing a 1 to this bit. Ports can be disabled by either a fault condition (disconnect event or other fault condition) or by host software. The bit status does not change until the port state actually changes. There may be a delay in disabling or enabling a port due to other host controller and bus events. 0=disable 1=enable(default)
0	0h RW	**Current Connect Status (CCS):** This value reflects the current state of the port, and may not correspond directly to the event that caused the Connect Status Change bit (Bit 1) to be set. 0=no device is present 1=device is present on port.

29.9.21 Port Power Management Status and Control USB3 (PORTPMSCX)—Offset 584h

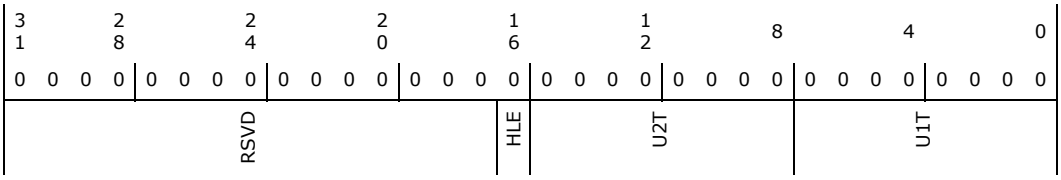
There are 10 USB3 PORTPMSC registers at offsets:
584h, 594h, ... (484h + (# of USB3 ports-1)*10h)

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 00000000h



Bit Range	Default and Access	Field Name (ID): Description
31:17	0000h RO	Reserved.
16	0h RW	**Hardware LPM Enable (HLE):** 0=disable 1=Enable. When this bit is a 1, hardware controlled LPM shall be enabled for this port. Refer to Section 4 of the *USB 2.0 LPM Specification* for more information.
15:8	00h RW/S	**U2 Timeout (U2T)**
7:0	00h RW/S	**U1 Timeout (U1T)**

29.9.22 USB3 Port Link Info (PORTLI)—Offset 588h

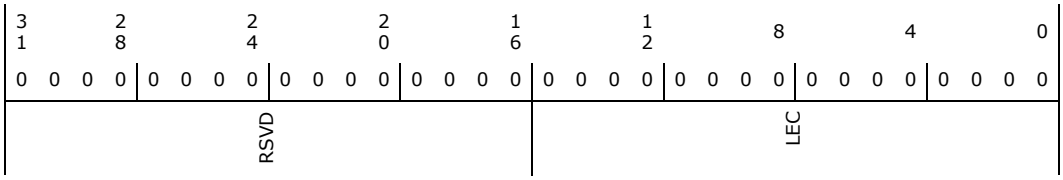
There are 10 USB3 PORTLI registers at offsets:
588h, 598h, ... (584h + (# of USB3 ports-1)*10h)

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 00000000h



Bit Range	Default and Access	Field Name (ID): Description
31:16	00h RO	Reserved.
15:0	0000h RW	**Link Error Count (LEC)** Displays the Link Error Count for the USB3 Port

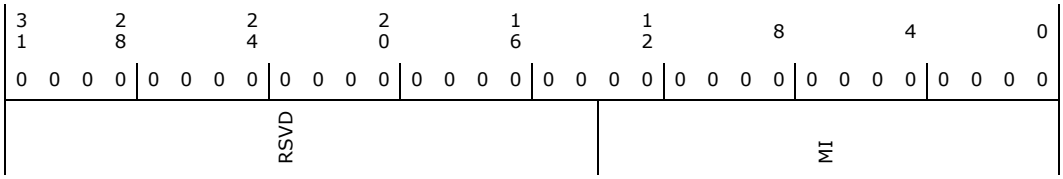
29.9.23 Microframe Index (RTMFINDEX)—Offset 2000h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 00000000h



Bit Range	Default and Access	Field Name (ID): Description
31:14	0h RO	Reserved.
15:0	0h RO	**Microframe Index (MI):** The value in this register increments at the end of each microframe (for example, 125 us). Bits [13:3] may be used to determine the current 1ms frame index.

29.9.24 Interrupt x Management (IMANx) - Offset 2020h

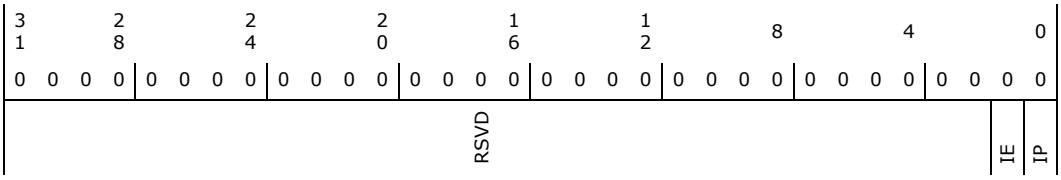
There are 8 IMAN registers at offsets 2020h, 2040h, 2060h ... 2100h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 00000000h



Bit Range	Default and Access	Field Name (ID): Description
31:2	0000000h RO	Reserved.
1	0b RW	Interrupt Enable (IE)
0	0b RW/1C	**Interrupt Pending (IP):** 0 = No interrupt is pending for the interrupter. 1 = An interrupt is pending for this interrupter. This bit is to 1b when IE =1, the IMODI interrupt Moderation Counter field = 0b, the Event Ring associated with the Interrupter is not empty (or for the Primary Interrupter when HCE flag is set to 1b), and EHB = 0. If MSI interrupts are enabled, this flag shall be cleared automatically when the PCI Dword write generated by the interrupt assertion is complete. If PCI pin Interrupts are enabled, this flag shall be cleared by SW.

29.9.25 Interrupter X Moderation (IMODx) - Offset 2024h

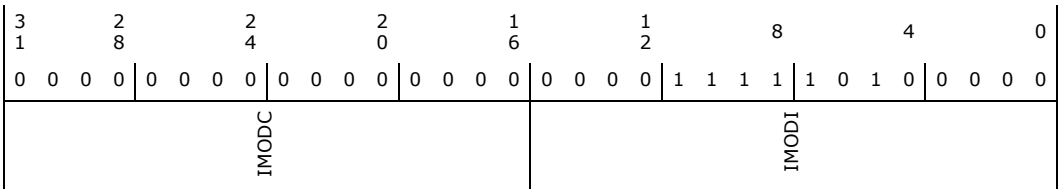
There are 8 IMOD registers at offsets 2024h, 2044h, 2064h ... 2104h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 00000FA0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0000 RW	**Interrupt Moderation Counter (IMODC)**
15:0	0FA0 RW/1C	**Interrupt Moderation Interval (IMODI):**

29.9.26 Event Ring Segment Table Size x (ERSTSZx)—Offset 2028h

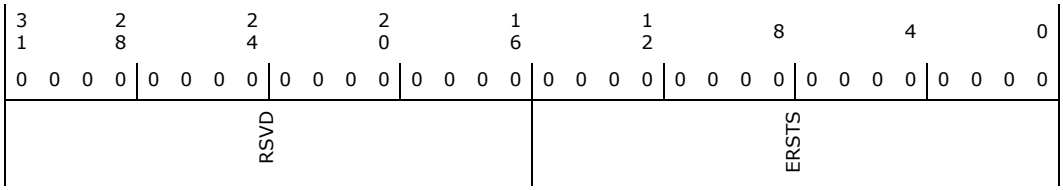
There are 8 ERSTSZ register.x = 1, 2, ..., 8
Address Offset: 2028-202Bh, 2048-204Bh, ..., 2028+(MaxInts-1)*20h-202B+(MaxInts-1)*20h

Access Method

Type: MEM Register (Size: 32 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:0	0h RW	**Event Ring Segment Table Size (ERSTS):** This field identifies the number of valid Event Ring Segment Table entries in the Event Ring Segment Table pointed to by the Event Ring Segment Table Base Address register.

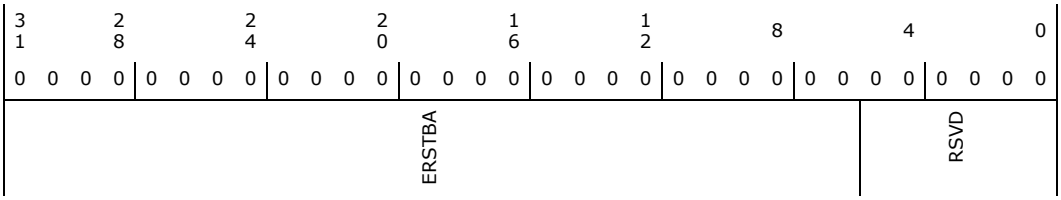
29.9.27 Event Ring Segment Table Base Address Low x (ERSTBA_LOx)—Offset 2030h

There are 8 ERSTBA_LO registersx = 1, 2, ..., 8
Address Offset: 2028-202Bh, 2048-204Bh, ..., 2028+(MaxInts-1)*20h-202B+(MaxInts-1)*20h

Access Method

Type: MEM Register (Size: 32 bits)
Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:6	0h RW	**Event Ring Segment Table Base Address Register (ERSTBA):** This field defines the low order bits of the start address of the Event Ring Segment Table. This field shall not be modified if HCHalted (HCH) = 0.
5:0	0h RO	Reserved.

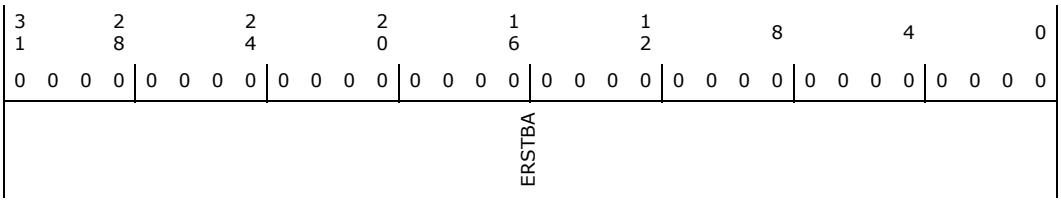
29.9.28 Event Ring Segment Table Base Address High x (ERSTBA_HIx)—Offset 2034h

Address Offset: 2034-2037h, 2054-2057h, ..., 2034+(MaxInts-1)*20h-2037+(MaxInts-1)*20h

Access Method

Type: MEM Register (Size: 32 bits)
Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Event Ring Segment Table Base Address (ERSTBA):** This field defines the low order bits of the start address of the Event Ring Segment Table. This field shall not be modified if HCHalted (HCH) = 0.

29.9.29 Event Ring Dequeue Pointer Low x (ERDP_LOx)—Offset 2038h

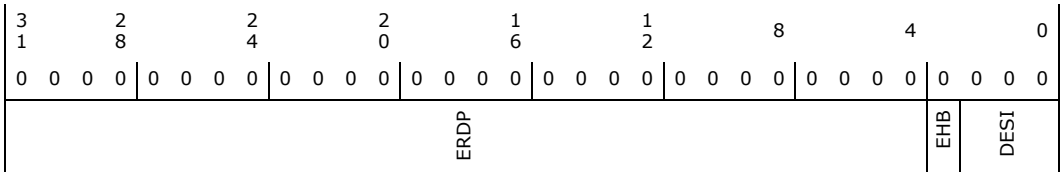
There are 8 ERDP_LO registers.x = 1, 2, ...,8
br] Address Offset: 2038-203Bh, 2058-205Bh, ..., 2038+(MaxInts-1)*20h-203B+(MaxInts-1)*20h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:4	0h RW	**Event Ring Dequeue Pointer (ERDP)**
3	0h RW/C	**Event Handler Busy (EHB)**
2:0	0h RW	**Dequeue ERST Segment Index (DESI)**

29.9.30 Event Ring Dequeue Pointer High x (ERDP_HIx)—Offset 203Ch

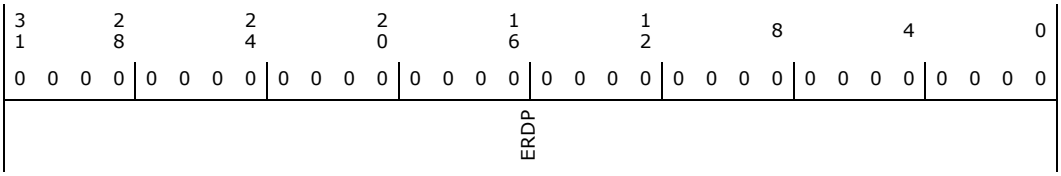
There are 8 ERDP_HI registers.x = 1, 2, ..., 8
Address Offset: 203C-203Fh, 205C-205Fh, ..., 203C+(MaxInts-1)*20h-203F+(MaxInts-1)*20h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Event Ring Dequeue Pointer (ERDP)**

29.9.31 Door Bell x (DBx)—Offset 3000h

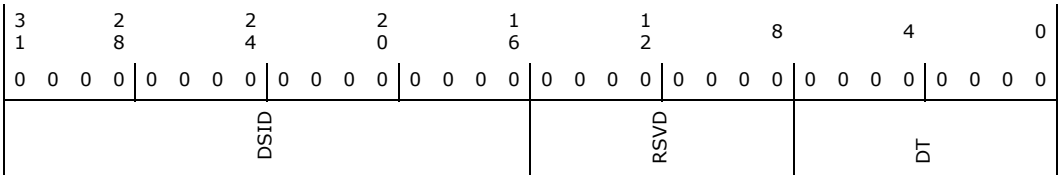
Door Bell registers are an array of 64 registers, with 0 to 32 being used by the XHC and the rest being reserved.
Address Offset: 3000-3003h, 3004-3007h, ..., 307C-307Fh

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RW	**DB Stream ID (DSID)**
15:8	0h RO	Reserved.
7:0	0h RW	**DB Target (DT):** This field defines the target of the doorbell reference. The table below defines the xHC notification that is generated by ringing the doorbell. Doorbell Register 0 is dedicated to Command Ring and decodes this field differently than the other Doorbell Registers. Refer to the xHCI Specification for definitions of the values.

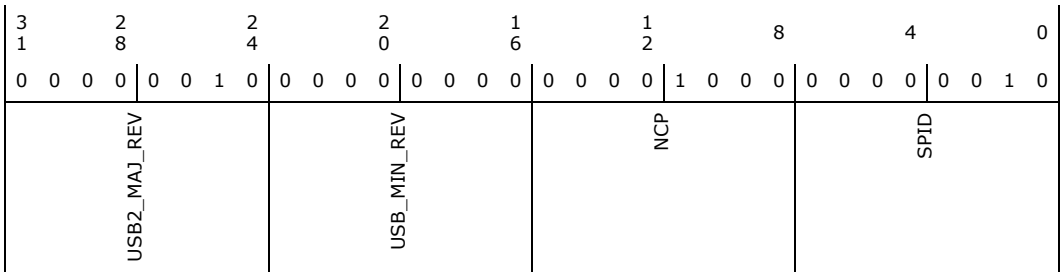
29.9.32 XECP_SUPP_USB2_0 (XECP_SUPP_USB2_0)—Offset 8000h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 2000802h



Bit Range	Default and Access	Field Name (ID): Description
31:24	2h RO	**USB Major Revision: 2.0 (USB2_MAJ_REV):** Major Specification Release Number in Binary-Coded Decimal (i.e., version 3.x is 03h). This field identifies the major release number component of the specification with which the xHC is compliant.
23:16	0h RO	**USB Minor Revision (USB_MIN_REV):** Minor Specification Release Number in Binary-Coded Decimal (i.e., version x.10 is 10h). This field identifies the minor release number component of the specification with which the xHC is compliant.
15:8	8h RO	**Next Capability Pointer (NCP):** This field indicates the location of the next capability with respect to the effective address of this capability. Refer to Table 145 for more information on this field.
7:0	2h RO	**Supported Protocol ID (SPID):** This field identifies the xHCI Extended capability. Refer to Table 146 for a list of the valid xHCI extended capabilities.

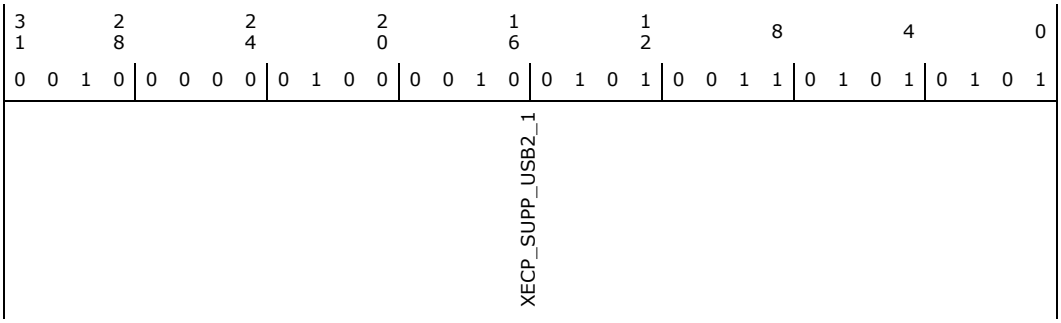
29.9.33 XECP_SUPP_USB2_1 (XECP_SUPP_USB2_1)—Offset 8004h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 20425355h



Bit Range	Default and Access	Field Name (ID): Description
31:0	20425355h RO	**XECP_SUPP_USB2_1 (XECP_SUPP_USB2_1):** This field is a mnemonic name string that references the specification with which the xHC is compliant. Four ASCII characters may be defined. Allowed characters are: alphanumeric, space, and underscore. Alpha characters are case sensitive.

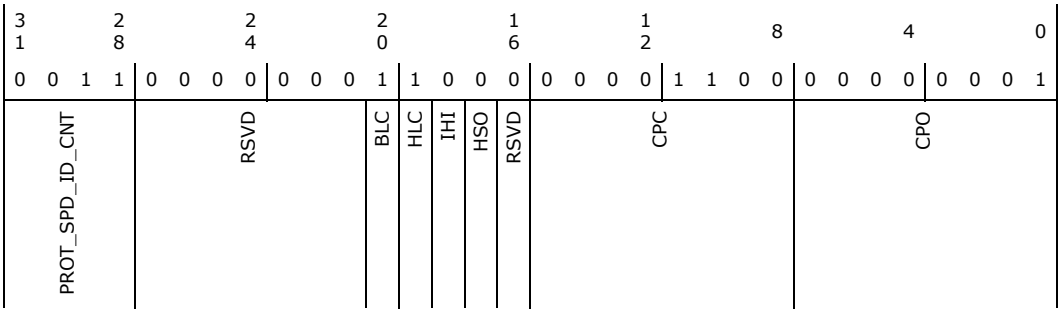
29.9.34 XECP_SUPP_USB2_2 (XECP_SUPP_USB2_2)—Offset 8008h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 30180C01h



Bit Range	Default and Access	Field Name (ID): Description
31:28	3h RO	**Protocol Speed ID Count (PROT_SPD_ID_CNT):** 3 USB 2.0 Speed (High, Full, Low)
27:21	0h RO	Reserved.
20	1h RO	**BESL LPM Capability (BLC):** Bit is set to 1 to indicate that the the ports described by this xHCI Supported Protocol Capability will apply BESL timing to BESL and BESLD fields of the PORTPMSC and PORTHLPMCC registers.
19	1h RO	**Protocol Defined - Hardware LMP Capability (HLC)**
18	0h RO	**Protocol Defined - Integrated Hub Implementation (IHI)**
17	0h RO	**Protocol Defined - High SPeed Only (HSO):** This field indicates the number of Protocol Speed ID (PSI) Dwords that the xHCI Supported Protocol Capability data structure contains. If this field is non-zero, then all speeds supported by the protocol shall be defined using PSI Dwords, i.e., no implied Speed ID mappings apply.
16	0h RO	Reserved.
15:8	Ch RO	**Compatible Port Count (CPC):** This field identifies the number of consecutive Root Hub Ports (starting at the Compatible Port Offset) that support this protocol. Valid values are 1 to MaxPorts.
7:0	1h RO	**Compatible Port Offset (CPO):** This field specifies the starting Port Number of Root Hub Ports that support this protocol. Valid values are '1' to MaxPorts.

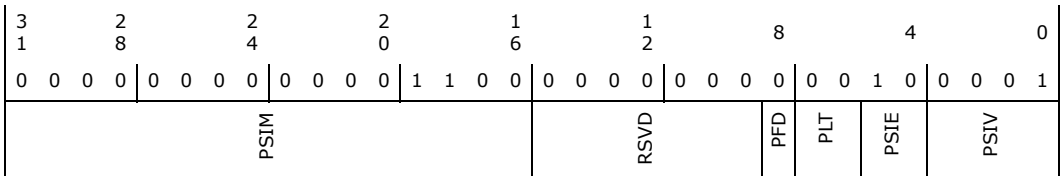
29.9.35 XECP_SUPP_USB2_3 (Full Speed) (XECP_SUPP_USB2_3)—Offset 8010h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: C0021h



Bit Range	Default and Access	Field Name (ID): Description
31:16	Ch RO	**Protocol Speed ID Mantissa (PSIM):** This field defines the mantissa that shall be applied to the PSIE when calculating the maximum bit rate represented by this PSI Dword.
15:9	0h RO	Reserved.
8	0h RO	**PSI Full Duplex (PFD):** If this bit is '1' the link is full-duplex (dual-simplex), and if '0' the link is half-duplex (simplex).

Bit Range	Default and Access	Field Name (ID): Description
7:6	0h RO	**PSI Type (PLT):** This field identifies whether the PSI Dword defines a symmetric or asymmetric bit rate, and if asymmetric, then this field also indicates if this Dword defines the receive or transmit bit rate. Note that the Asymmetric PSI Dwords shall be paired, i.e., an Rx immediately followed by a Tx, and both Dwords shall define the same value for the PSIV. PLT Value Bit Rate Note 0 Symmetric Single PSI Dword 1 Reserved 2 Asymmetric Rx Paired with Asymmetric Tx PSI Dword 3 Asymmetric Tx Immediately follows Rx Asymmetric PSI Dword
5:4	2h RO	**Protocol Speed ID Exponent (PSIE):** This field defines the base 10 exponent times 3, that shall be applied to the Protocol Speed ID Mantissa when calculating the maximum bit rate represented by this PSI Dword. PSIE Value Bit Rate 0 Bits per second 1 Kb/s 2 Mb/s 3 Gb/s
3:0	1h RO	**Protocol Speed ID Value (PSIV):** If a device is attached that operates at the bit rate defined by this PSI Dword, then the value of this field shall be reported in the Port Speed field of PORTSC register (5.4.8) of a compatible port. ***Note:*** The PSIV value of '0' is reserved and shall not be defined by a PSI.

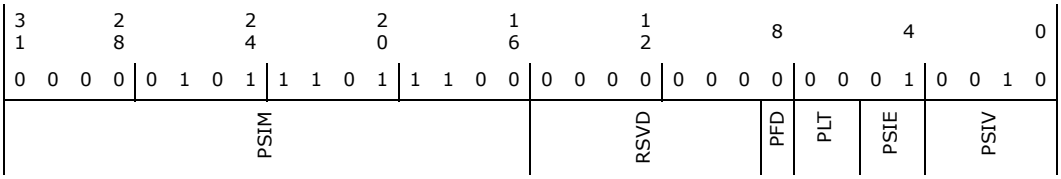
29.9.36 XECP_SUPP_USB2_4 (Low Speed) (XECP_SUPP_USB2_4)—Offset 8014h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 5DC0012h



Bit Range	Default and Access	Field Name (ID): Description
31:16	5DCh RO	**Protocol Speed ID Mantissa (PSIM)**
15:9	0h RO	Reserved.
8	0h RO	**PSI Full Duplex (PFD)**
7:6	0h RO	**PSI Type (PLT)**
5:4	1h RO	**Protocol Speed ID Exponent (PSIE)**
3:0	2h RO	**Protocol Speed ID Value (PSIV)**

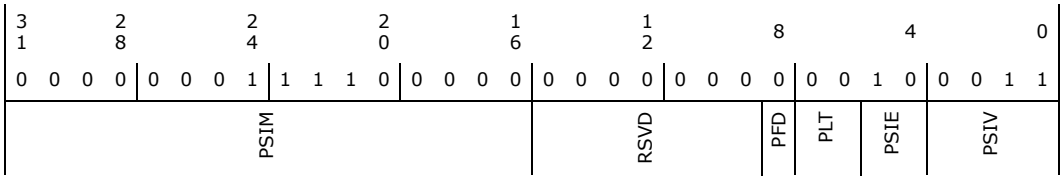
29.9.37 XECP_SUPP_USB2_5 (High Speed) (XECP_SUPP_USB2_5)—Offset 8018h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 1E00023h



Bit Range	Default and Access	Field Name (ID): Description
31:16	1E0h RO	**Protocol Speed ID Mantissa (PSIM)**
15:9	0h RO	Reserved.
8	0h RO	**PSI Full Duplex (PFD)**
7:6	0h RO	**PSI Type (PLT)**
5:4	2h RO	**Protocol Speed ID Exponent (PSIE)**
3:0	3h RO	**Protocol Speed ID Value (PSIV)**

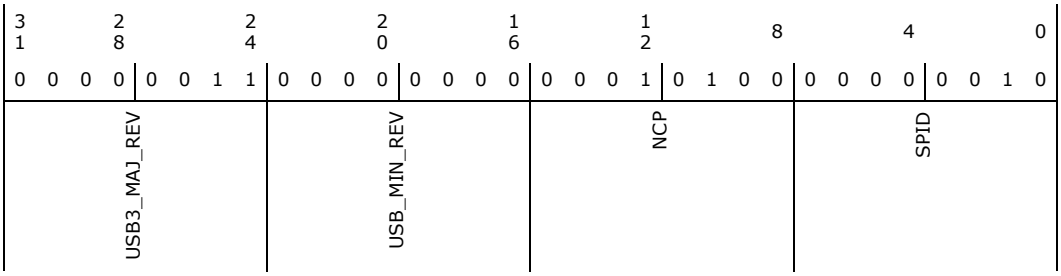
29.9.38 XECP_SUPP_USB3_0 (XECP_SUPP_USB3_0)—Offset 8020h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 3001402h



Bit Range	Default and Access	Field Name (ID): Description
31:24	3h RO	**USB Major Revision: 3.0 (USB3_MAJ_REV)**
23:16	0h RO	**USB Minor Revision (USB_MIN_REV)**
15:8	14h RO	**Next Capability Pointer (NCP)**
7:0	2h RO	**Supported Protocol ID (SPID)**

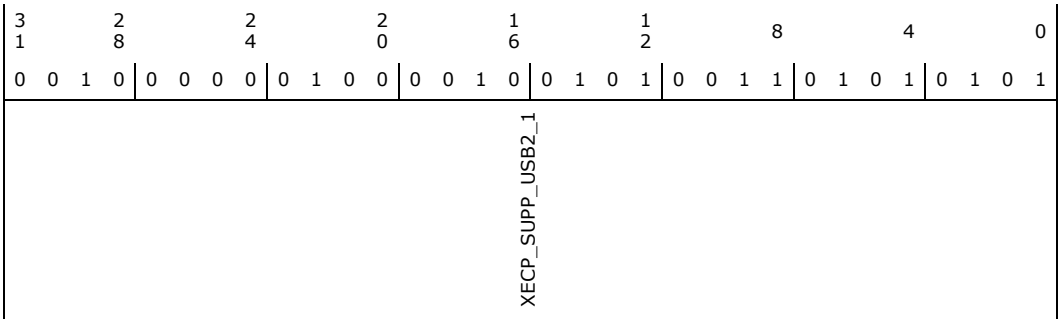
29.9.39 XECP_SUPP_USB3_1 (XECP_SUPP_USB3_1)—Offset 8024h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 20425355h



Bit Range	Default and Access	Field Name (ID): Description
31:0	20425355h RO	**XECP_SUPP_USB2_1 (XECP_SUPP_USB2_1):** Namestring USB

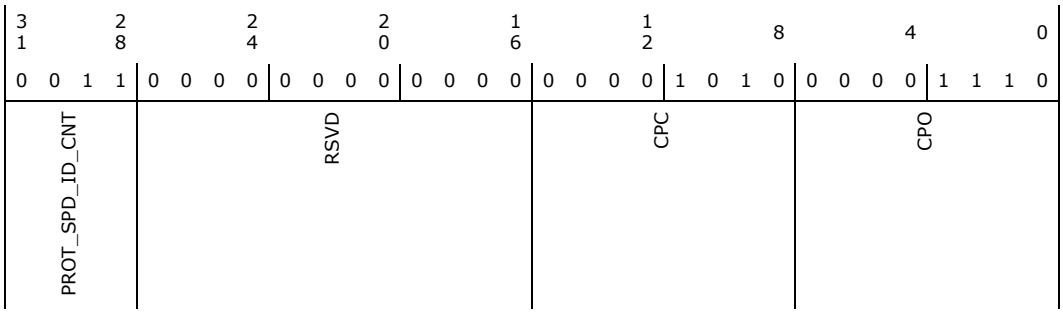
29.9.40 XECP_SUPP_USB3_2 (XECP_SUPP_USB3_2)—Offset 8028h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 30000A0Eh



Bit Range	Default and Access	Field Name (ID): Description
31:28	3h RO	**Protocol Speed ID Count (PROT_SPD_ID_CNT):** 1 USB 3.0 Speed (Supper Speed)
27:16	0h RO	Reserved.
15:8	Ah RO	**Compatible Port Count (CPC)**
7:0	Eh RO	**Compatible Port Offset (CPO)**

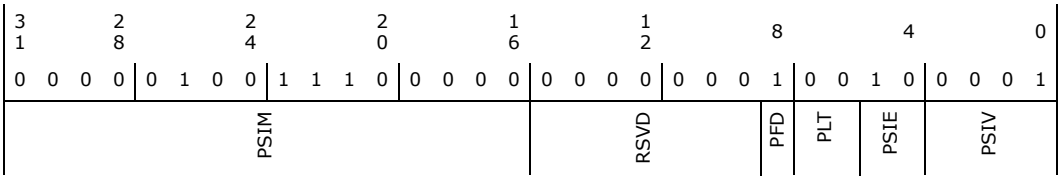
29.9.41 XECP_SUPP_USB3_3 (XECP_SUPP_USB3_3)—Offset 8030h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 4E00121h



Bit Range	Default and Access	Field Name (ID): Description
31:16	4E0h RO	**Protocol Speed ID Mantissa (PSIM)**
15:9	0h RO	Reserved.
8	1h RO	**PSI Full Duplex (PFD)**
7:6	0h RO	**PSI Type (PLT)**
5:4	2h RO	**Protocol Speed ID Exponent (PSIE)**
3:0	1h RO	**Protocol Speed ID Value (PSIV)**

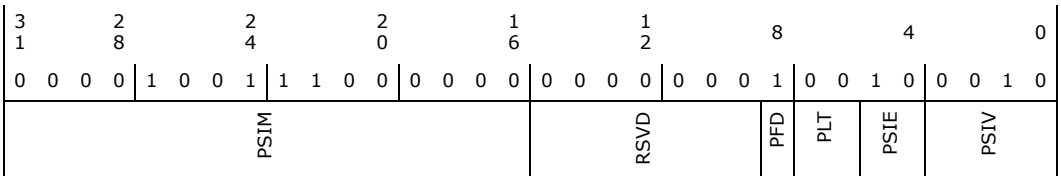
29.9.42 XECP_SUPP_USB3_4 (XECP_SUPP_USB3_4)—Offset 8034h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 9C00122h



Bit Range	Default and Access	Field Name (ID): Description
31:16	9C0h RO	**Protocol Speed ID Mantissa (PSIM)**
15:9	0h RO	Reserved.
8	1h RO	**PSI Full Duplex (PFD)**
7:6	0h RO	**PSI Type (PLT)**
5:4	2h RO	**Protocol Speed ID Exponent (PSIE)**
3:0	2h RO	**Protocol Speed ID Value (PSIV)**

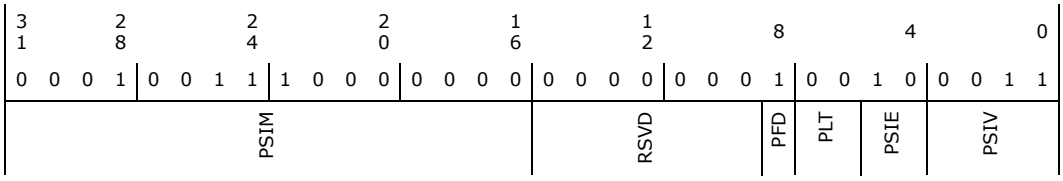
29.9.43 XECP_SUPP_USB3_5 (XECP_SUPP_USB3_5)—Offset 8038h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 13800123h



Bit Range	Default and Access	Field Name (ID): Description
31:16	1380h RO	**Protocol Speed ID Mantissa (PSIM)**
15:9	0h RO	Reserved.
8	1h RO	**PSI Full Duplex (PFD)**
7:6	0h RO	**PSI Type (PLT)**
5:4	2h RO	**Protocol Speed ID Exponent (PSIE)**
3:0	3h RO	**Protocol Speed ID Value (PSIV)**

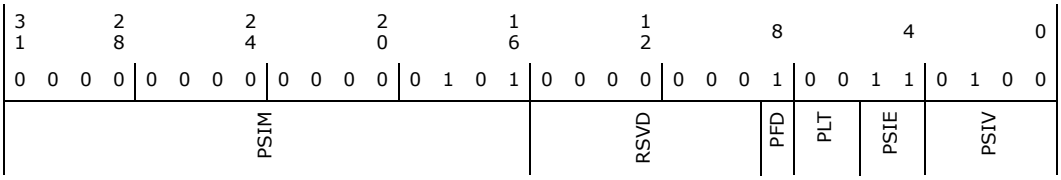
29.9.44 XECP_SUPP_USB3_6 (XECP_SUPP_USB3_6)—Offset 803Ch

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 50134h



Bit Range	Default and Access	Field Name (ID): Description
31:16	5h RO	**Protocol Speed ID Mantissa (PSIM)**
15:9	0h RO	Reserved.
8	1h RO	**PSI Full Duplex (PFD)**
7:6	0h RO	**PSI Type (PLT)**
5:4	3h RO	**Protocol Speed ID Exponent (PSIE)**
3:0	4h RO	**Protocol Speed ID Value (PSIV)**

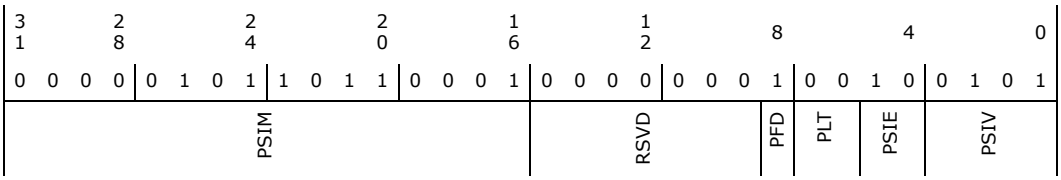
29.9.45 XECP_SUPP_USB3_7 (XECP_SUPP_USB3_7)—Offset 8040h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 5B10125h



Bit Range	Default and Access	Field Name (ID): Description
31:16	5B1h RO	**Protocol Speed ID Mantissa (PSIM)**
15:9	0h RO	Reserved.
8	1h RO	**PSI Full Duplex (PFD)**
7:6	0h RO	**PSI Type (PLT)**
5:4	2h RO	**Protocol Speed ID Exponent (PSIE)**
3:0	5h RO	**Protocol Speed ID Value (PSIV)**

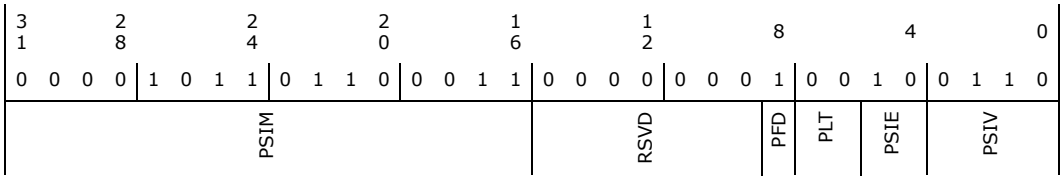
29.9.46 XECP_SUPP_USB3_8 (XECP_SUPP_USB3_8)—Offset 8044h

Access Method

Type: MEM Register (Size: 32 bits)

Device: 20
Function: 0

Default: B630126h



Bit Range	Default and Access	Field Name (ID): Description
31:16	B63h RO	**Protocol Speed ID Mantissa (PSIM)**
15:9	0h RO	Reserved.
8	1h RO	**PSI Full Duplex (PFD)**
7:6	0h RO	**PSI Type (PLT)**
5:4	2h RO	**Protocol Speed ID Exponent (PSIE)**
3:0	6h RO	**Protocol Speed ID Value (PSIV)**

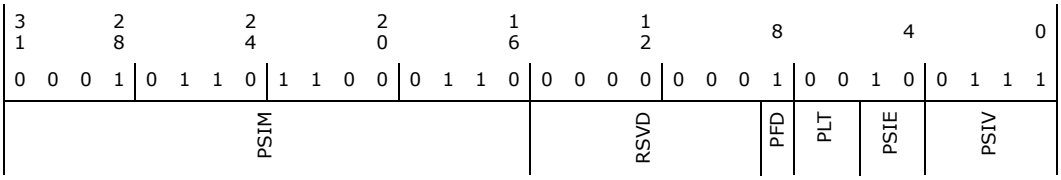
29.9.47 XECP_SUPP_USB3_9 (XECP_SUPP_USB3_9)—Offset 8048h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 16C60127h



Bit Range	Default and Access	Field Name (ID): Description
31:16	16C6h RO	**Protocol Speed ID Mantissa (PSIM)**
15:9	0h RO	Reserved.
8	1h RO	**PSI Full Duplex (PFD)**
7:6	0h RO	**PSI Type (PLT)**
5:4	2h RO	**Protocol Speed ID Exponent (PSIE)**
3:0	7h RO	**Protocol Speed ID Value (PSIV)**

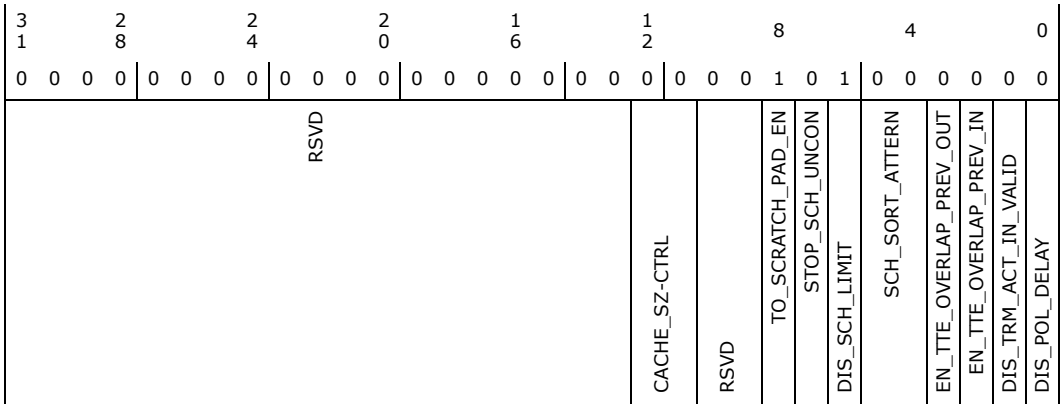
29.9.48 Host Control Scheduler (HOST_CTRL_SCH_REG)—Offset 8094h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 00000100



Bit Range	Default and Access	Field Name (ID): Description
31:13	00000 RO	**Reserved**.
12:11	00b RW	**Cache Size Control Reg (CACHE_SZ_CTRL):** 0: 64 1: 32 2,3: 16
10:9	00b RO	**Reserved**.
8	1b RW	**Turn On Scratch_pad_en (TO_SCRATCH_PAD_EN):** Enables scratch pad function.
7	0b RW	**Scheduler Host Control Reg (STOP_SCH_UNCON)** Enable check to stop scheduling on port that are not connected.
6	0b RW	**Disable Scheduling Limit (DIS_SCH_LIMIT)** Disable 1 pack scheduling limit when ISO pending in present microframe.
5:4	00b RW	**Scheduler Sort Pattern (SCH_SORT_PATTERN)** 00 (default) Search ISO ahead of interrupt within each service interval. 01 - Search USB2-ISO, USB3-ISO, USB2-Interrupt, USB3-Interrupt within each service interval. 10 - Search strictly by interval. 11 - Search all ISO intervals ahead interrupt intervals and within each interval, USB2 ahead of USB3.
3	0b RW	**Enable TTE Overlap Prevention Interrupt Out (EN_TTE_OVERLAP_PREV_OUT)** Enable TTE overlap prevention on interrupt OUT EPs (at cost of possible service interval slip.

Bit Range	Default and Access	Field Name (ID): Description
2	0b RW	**Enable TTE Overlap Prevention Interrupt IN(EN_TTE_OVERLAP_PREV_IN)** Enable TTE overlap prevention on interrupt IN EPs (at cost of possible service interval slip.
1	0b RW	**Disable TRM check (DIS_TRM_ACT_IN_VALID)** Disable TRM active IN EP valid check function.
0	0b RW	**Disable Poll Delay (DIS_POL_DELAY)** Disable poll delay function.

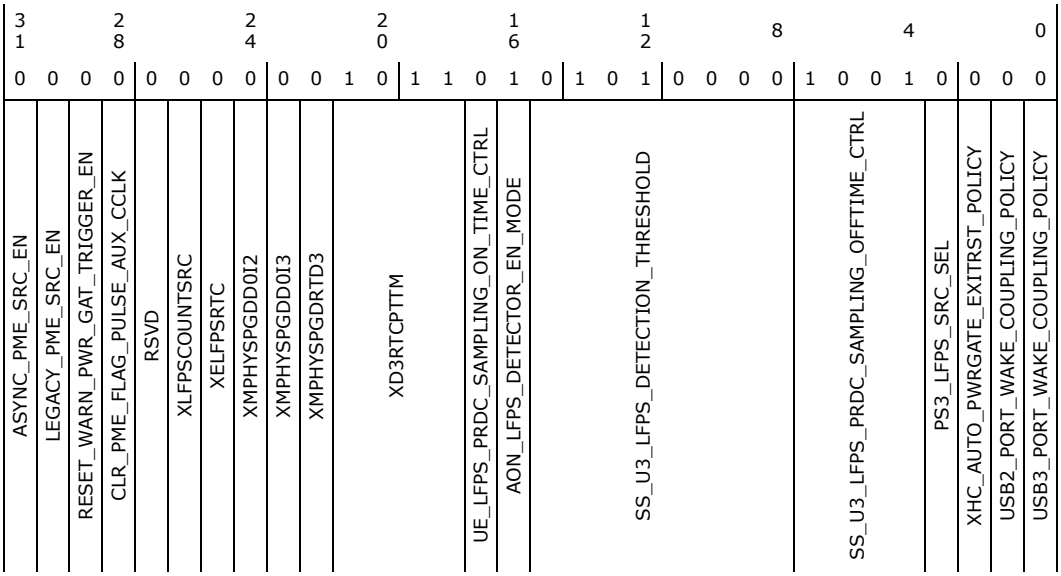
29.9.49 Power Management Control (PMCTRL_REG)—Offset 80A4h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 002D5090h



Bit Range	Default and Access	Field Name (ID): Description
31	0b RW	**Async PME Source Enable (ASYNC_PME_SRC_EN)** This field allows the async PME source to be allowed to generate PME. This is specifically required for SOCs that do not allow for any clock other than RTC to be available during RTD3.
30	0b RW	**Legacy PME Source Enable (LEGACY_PME_SRC_EN)** This field allows the legacy PME source to be used in PME generation. The legacy source in reference to the source prior to the RTD3 changes.
29	0b RW	**Reset Warn Power Gate Trigger Disable (RESET_WARN_PWR_GAT_TRIGGER_EN)** This field controls the actions taken for due to reset warn. 0 - Reset Warn will trigger a HW autonomous Power Gate. 1 - Reset Warn will not trigger a HW autonomous Power Gate.

Bit Range	Default and Access	Field Name (ID): Description
28	0b WO	**Internal PME Flag clear (CLR_PME_FLAG_PULSE_AUX_CCLK)** This Write-Only bit can be used to clear the internal PME flag. SW write to 1 will clear the PME flag. SW write to 0 will have no effect and be ignored by the controller. Read always return 0.
27	0b RW	**Reserved**
26	0b	**Source of LFPS OFF Counter (XLFPSCOUNTSRC)** 0: Central RTC Counter for LFPS detection 1: Local Counter for LFPS detection
25	0b RW	**Enable LFPS Filtering on RTC (XELFPSRTC)** 0: Use Oscillator clock for LFPS Filtering during P3. 1: Use RTC Clock for LFPS Filtering during P3.
24	0b RW	**Modphy power gate disable DOI2(XMPHYSPGDD0I2)** ModPHY SUS Well Power Gate Disable for D0I2 0: ModPHY SUS well power gating enabled 1: ModPHY SUS well power gating disabled
23	0b RW	**Modphy power gate disable DOI3 (XMPHYSPGDD0I3)** ModPHY SUS Well Power Gate Disable for D0I3 0: ModPHY SUS well power gating enabled 1: ModPHY SUS well power gating disabled
22	0b RW	**Modphy power gate disable RTD3 (XMPHYSPGDRTD3)** ModPHY SUS Well Power Gate Disable for RTD3 0: ModPHY SUS well power gating enabled 1: ModPHY SUS well power gating disabled
21:18	Bh RW	**D3 RTC Port Timer Tick Multiplier (XD3RTCPTTM)** This register will be the multiplication factor for determining SSIC Wake Detection Frequency and RXDET based on the XD3RTCPTTC value. If XD3RTCPTTC is 9h and this register is Bh, frequency for RXDET and SSIC H8EXIT detection while MODPHY SUS Power gating is enabled would be 99 ms.
17	0b RW	**U3 LFPS Periodic Sampling ON time Control (UE_LFPS_PRDC_SAMPLING_ON_TIME_CTRL);** This field controls the ON time for the LFPS periodic sampling for USB3/SSIC ports. 0: ON time is 2 RTC clocks. 1: ON time is 3 RTC clocks.
16	1b RW	**AON LFPS Detector Enable Mode (AON_LFPS_DETECTOR_EN_MODE)** 1 - Allow the LFSP Detector in AON to own LFPS detection when the port is in PS3 for U2/U3 - not RxD regardless of port ownership. 0 - Allow the LFPS Detector in AON to own the LFPS detection only when the AON owns the port and in U2/U3 - not RxD.
15:8	50h RW	**SS U3 LFPS Detection Threshold (SS_U3_LFPS_DETECTION_THRESHOLD)** This field controls the threshold used to determine when a valid U3 Wake is detected through when using the unfiltered LFPS source. The value on this field will reflect the binary count required to have been detected on the counter being clocked by the unfiltered LFPS source to result in a valid U3 wake detection.
7:4	9h RW	**SS_U3_LFPS_PRDC_SAMPLING_OFFTIME_CTRL (SS_U3_LFPS_PRDC_SAMPLING_OFFTIME_CTRL)** This field controls the OFF time for the LFPS periodic sampling for USB3 Ports 0x0 periodic sampling is disabled. 0x1 OFF time is 1 ms. 0x2 OFF time is 2 ms. 0xF OFF time is 15 ms. The ON Time is determined by the amount of time required to reliably determine if there is a valid LFPS and is HW implementation specific. A speed up mode shall be implemented where this field is in units of us. i.e., 0x1 = 1 us OFF time, 0x2 = 2 us OFF time, etc.
3	0b RW	**PS3 LFPS Source Select (PS3_LFPS_SRC_SEL)** 0: LFPS Source is unfiltered. 1: LFPS Source is filtered (Rx-Elec-Idle). LFPS Source is Rx-Elec-Idle for any non PS3 state.

Bit Range	Default and Access	Field Name (ID): Description
2	0b RW	**XHCI Engine Autonomous Power Gate Exit Reset Policy (XHC_AUTO_PWRGATE_EXITRST_POLICY)** Controls when the xHCI engine is brought out of reset due to a power ungate. 0: Engine is brought out of reset when D3 to D0 is triggered. This allows for a quick power up sequence while leaving the virtual PCIe LTSSM in L23 is power ungate is not due to D3 to D0. 1: Engine is brought out of reset along with the rest of the IP. This is required for PMC save/restore flow.
1	0b RW	**USB2 Port Wake Unit Coupling Policy (USB2_PORT_WAKE_COUPLING_POLICY)** Controls the trigger for USB2 Port Wake Units to initiate Port Level Power Off Preparation. 0: RTD3 triggered 1: - Port Triggered when in L1, L2 or Disabled, Disconnected
0	0b RW	**USB3 Port Wake Unit Coupling Policy (USB3_PORT_WAKE_COUPLING_POLICY)** Controls the trigger for USB3 Port Wake Units to initiate Port Level Power Off Preparation. 0 - RTD3 Triggered 1 - Port Triggered when in PS3 due to RxDetect, U3, U2 or Disabled

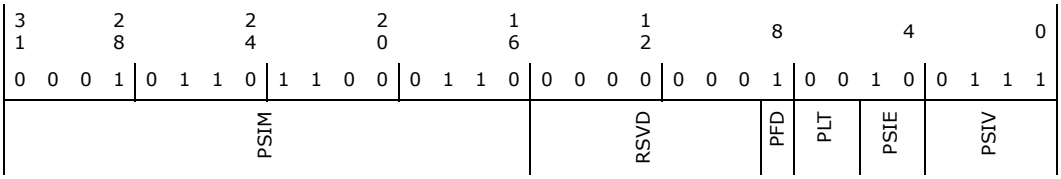
29.9.50 Super Speed Port Enable (SSPE) —Offset 80B8h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 16C60127h



Bit Range	Default and Access	Field Name (ID): Description
31:10	0h RO	**Reserved**
9:0	2FFh RW	USB3 Port Enable 1 = Enables PORTSC to see the connects on the port. 0 = Blocks PORTSC from reporting any attach/connects.

29.9.51 USB Legacy Support Capability (USBLEGSUP)—Offset 846Ch

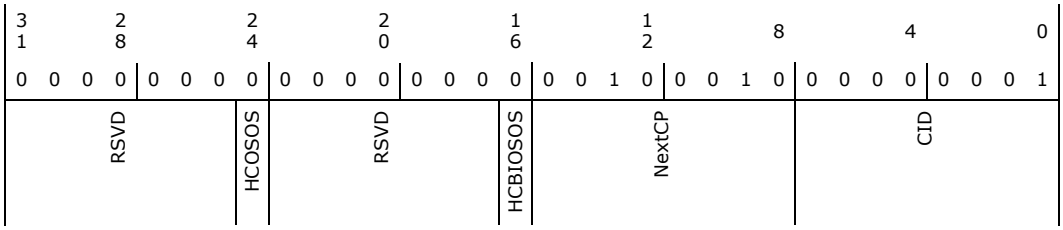
This register is modified and maintained by the BIOS.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 2201h



Bit Range	Default and Access	Field Name (ID): Description
31:25	0h RO	Reserved.
24	0h RW	**HC OS Owned Semaphore (HCOSOS):** Default = '0'. System software sets this bit to request ownership of the xHC. Ownership is obtained when this bit reads as '1' and the HC BIOS Owned Semaphore bit reads as '0'.
23:17	0h RO	Reserved.
16	0h RW	**HC BIOS Owned Semaphore (HCBIOSOS):** Default = '0'. The BIOS sets this bit to establish ownership of the xHC. System BIOS will set this bit to a '0' in response to a request for ownership of the xHC by system software.
15:8	22h RW/S	**Next Capability Pointer (NextCP):** This field indicates the location of the next capability with respect to the effective address of this capability. Refer to Table 145 for more information on this field.
7:0	1h RW/L	**Capability ID (CID):** This field identifies the extended capability. Refer to Table 146 for the value that identifies the capability as Legacy Support. This extended capability requires one additional 32-bit register for control/status information (USBLEGCTLSTS), and this register is located at offset xECP+04h.

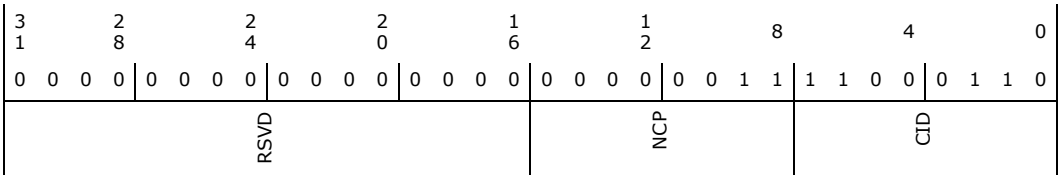
29.9.52 Port Disable Override Capability Register (PDO_CAPABILITY)—Offset 84F4h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 3C6h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:8	3h RW/S	**Next Capability Pointer (NCP)**
7:0	C6h RW/S	**Capability ID (CID)**

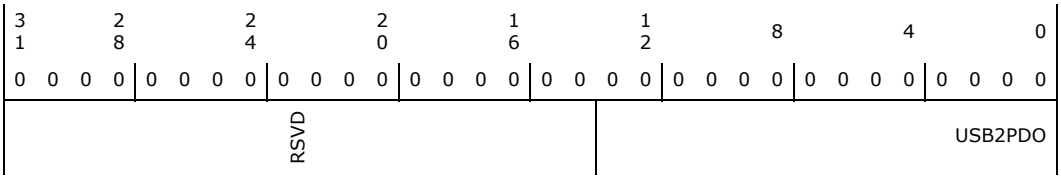
29.9.53 USB2 Port Disable Override (USB2PDO)—Offset 84F8h

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13:0	0h RW/O	**USB2PDO (USB2PDO):** A '1' in a bit position prevents the corresponding USB2 port from reporting a Device Connection to the XHC. This applies across all USB2 protocol ports. 0 = Allows corresponding USB port to report a device connection to the xHC. 1 = Prevents the corresponding USB port from reporting a device Connection to the xHC. Port to bit mapping is in one-hot encoding, that is bit 0 controls port 1 and so on. Bit 0 = USB 2.0 port 0 ... Bit N = USB 2.0 port N

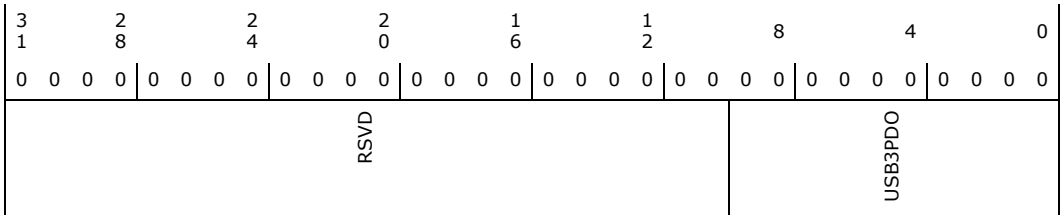
29.9.54 USB3 Port Disable Override (USB3PDO)—Offset 84FCh

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:10	0h RO	Reserved.
9:0	0h RW/O	**USB3 Port Disable Override (USB3PDO):** 0 = Allows corresponding USB port to report a Device Connection to the xHC. 1 = Prevents the corresponding USB port from reporting a Device Connection to the xHC. Bit 0 = USB 3.0 Port 1 ... Bit N = USB 3.0 Port N

29.9.55 Debug Capability ID Register (DCID)—Offset 8700h

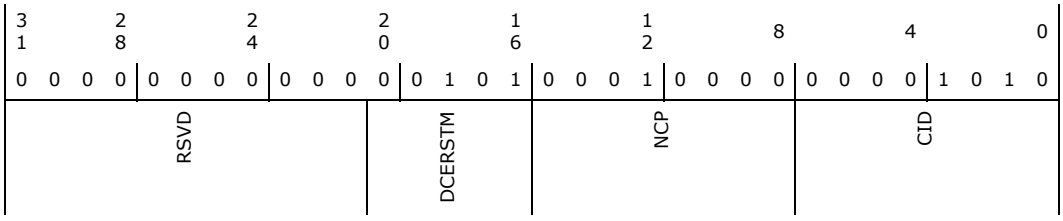
This register is modified and maintained by BIOS.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 20
Function: 0

Default: 5100Ah



Bit Range	Default and Access	Field Name (ID): Description
31:21	0h RO	Reserved.
20:16	5h RW	**Debug Capability Event Ring Segment Table Max (DCERSTM):** ***Note:*** This register is sticky.
15:8	10h RW	**Next Capability Pointer (NCP):** This register is sticky.
7:0	Ah RW	**Capability ID (CID):** This register is sticky.

§

30 Intel® Management Engine Interface (Intel® MEI) (D22:F0, D22:F1, and D22:F4)

30.1 Intel Management Engine Interface Configuration Registers Summary

This section describes the registers for the Intel Management Engine Interface 1 (Intel MEI 1) at Device 22: Function 0.

These registers also apply to the following controllers:

Intel MEI 2 at D22:F1

Intel MEI 3 at D22:F4

Table 30-1. Summary of Intel Management Engine Interface Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	Identifiers (HECI1_ID)—Offset 0h	X_8086h
4h	5h	Command (HECI1_CMD)—Offset 4h	0h
6h	7h	Status (HECI1_STS)—Offset 6h	10h
8h	Bh	Revision ID and Class Code (HECI1_RID_CC)—Offset 8h	7800000h
Ch	Ch	Cache Line Size (HECI1_CLS)—Offset Ch	0h
Dh	Dh	Master Latency Timer (HECI1_MLT)—Offset Dh	0h
Eh	Eh	Header Type (HECI1_HTYPE)—Offset Eh	80h
Fh	Fh	Built In Self-Test (HECI1_BIST)—Offset Fh	0h
10h	13h	HECI MMIO Base Address Low (HECI1_MMIO_MBAR_LO)—Offset 10h	4h
14h	17h	HECI MMIO Base Address High (HECI1_MMIO_MBAR_HI)—Offset 14h	0h
2Ch	2Fh	Sub System Identifiers (HECI1_SS)—Offset 2Ch	0h
34h	37h	Capabilities Pointer (HECI1_CAP)—Offset 34h	50h
3Ch	3Dh	Interrupt Information (HECI1_INTR)—Offset 3Ch	100h
3Eh	3Eh	Minimum Grant (HECI1_MGNT)—Offset 3Eh	0h
3Fh	3Fh	Maximum Latency (HECI1_MLAT)—Offset 3Fh	0h
40h	43h	Host Firmware Status Register 1 (HFSTS1)—Offset 40h	0h
48h	4Bh	Host Firmware Status Register 2 (HFSTS2)—Offset 48h	0h
4Ch	4Fh	Host General Status (HECI1_H_GS1)—Offset 4Ch	0h
50h	51h	PCI Power Management Capability ID (HECI1_PID)—Offset 50h	8C01h
52h	53h	PCI Power Management Capabilities (HECI1_PC)—Offset 52h	4003h
54h	55h	PCI Power Management Control and Status (HECI1_PMCS)—Offset 54h	8h
60h	63h	Host Firmware Status Register 3 (HFSTS3)—Offset 60h	0h
64h	67h	Host Firmware Status Register 4 (HFSTS4)—Offset 64h	0h
68h	6Bh	Host Firmware Status Register 5 (HFSTS5)—Offset 68h	0h

Table 30-1. Summary of Intel Management Engine Interface Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
6Ch	6Fh	Host Firmware Status Register 6 (HFSTS6)—Offset 6Ch	0h
70h	73h	Host General Status 2 (HECI1_H_GS2)—Offset 70h	0h
74h	77h	Host General Status 3 (HECI1_H_GS3)—Offset 74h	0h
8Ch	8Dh	Message Signaled Interrupt Identifiers (HECI1_MID)—Offset 8Ch	5h
8Eh	8Fh	Message Signaled Interrupt Message Control (HECI1_MC)—Offset 8Eh	80h
90h	93h	Message Signaled Interrupt Message Address (HECI1_MA)—Offset 90h	0h
94h	97h	Message Signaled Interrupt Upper Address (HECI1_MUA)—Offset 94h	0h
98h	99h	Message Signaled Interrupt Message Data (HECI1_MD)—Offset 98h	0h
A0h	A0h	HECI Interrupt Delivery Mode (HECI1_HIDM)—Offset A0h	0h

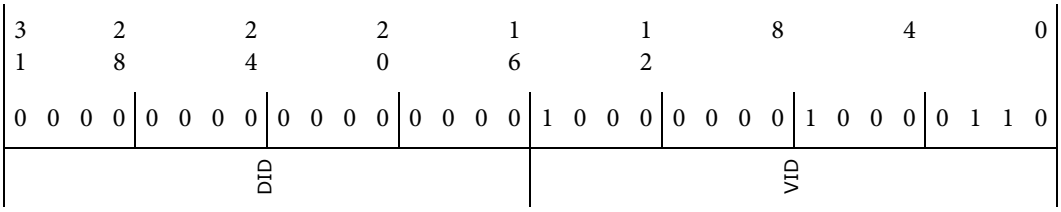
30.1.1 Identifiers (HECI1_ID)—Offset 0h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 0

Default: 8086h



Bit Range	Default and Access	Field Name (ID): Description
31:16	-- RO/V	**Device ID (DID):** Indicates what device number assigned by Intel. See the Device and Version ID Table in Volume 1 of the datasheet for the default value.
15:0	8086h RO	**Vendor ID (VID):** 16-bit field which indicates Intel is the vendor.

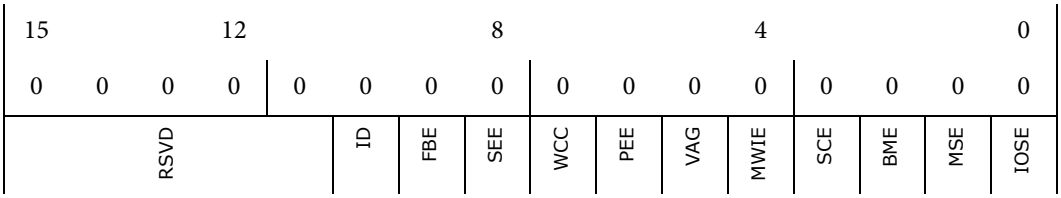
30.1.2 Command (HECI1_CMD)—Offset 4h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 22
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:11	0h RO	Reserved.
10	0h RW	**Interrupt Disable (ID)**
9	0h RO	**Fast Back-to-Back Enable (FBE):** Not implemented, hardwired to 0.
8	0h RO	**SERR# Enable (SEE):** Not implemented, hardwired to 0.
7	0h RO	**Wait Cycle Enable (WCC):** Not implemented, hardwired to 0.
6	0h RO	**Parity Error Response Enable (PEE):** Not implemented, hardwired to 0.
5	0h RO	**VGA Palette Snooping Enable (VAG):** Not implemented, hardwired to 0.
4	0h RO	**Memory Write and Invalidate Enable (MWIE):** Not implemented, hardwired to 0.
3	0h RO	**Special Cycle Enable (SCE):** Not implemented, hardwired to 0.
2	0h RW	**Bus Master Enable (BME):** Controls the HECI host controller's ability to act as a system memory master for data transfers. When this bit is cleared, HECI bus master activity stops and any active DMA engines return to an idle condition.
1	0h RW	**Memory Space Enable (MSE):** Controls access to the HECI host controllers memory mapped register space.
0	0h RO	**I/O Space Enable (IOSE):** Not implemented, hardwired to 0.

30.1.3 Status (HECI1_STS)—Offset 6h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 22
Function: 0

Default: 10h

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
DPE	SSE	RMA	RTA	STA	DEVT		DPD	FBC	RSVD	C66	CL	IS	RSVD		

Bit Range	Default and Access	Field Name (ID): Description
15	0h RO	**Detected Parity Error (DPE)**
14	0h RO	**Signaled System Error (SSE)**
13	0h RO	**Received Master-Abort (RMA)**

Bit Range	Default and Access	Field Name (ID): Description
12	0h RO	**Received Target Abort (RTA):** Not implemented, hardwired to 0.
11	0h RO	**Signaled Target-Abort (STA):** Not implemented, hardwired to 0.
10:9	0h RO	**DEVSEL# Timing (DEVT):** Not implemented, hardwired to 0.
8	0h RO	**Master Data Parity Error Detected (DPD):** Not implemented, hardwired to 0.
7	0h RO	**Fast Back-to-Back Capable (FBC):** Not implemented, hardwired to 0.
6	0h RO	Reserved.
5	0h RO	**66 MHz Capable (C66):** Not implemented, hardwired to 0.
4	1h RO	**Capabilities List (CL):** Indicates the presence of a capabilities list, hardwired to 1.
3	0h RO/V	**Interrupt Status (IS):** Reflects the state of the INTx# signal at the input of the enable/disable circuit. This bit is a 1 when the INTx# is asserted.This bit is a 0 after the interrupt is cleared (independent of the state of the Interrupt Disable bit in the command register). Note that this bit is not set by an MSI.
2:0	0h RO	Reserved.

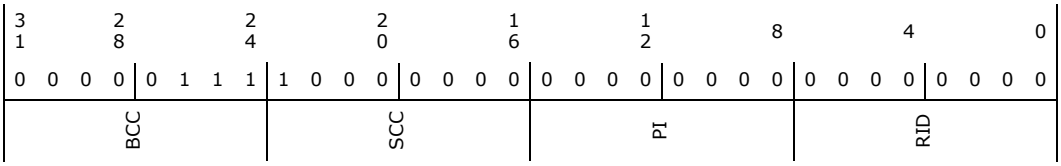
30.1.4 Revision ID and Class Code (HECI1_RID_CC)—Offset 8h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 0

Default: 7800000h



Bit Range	Default and Access	Field Name (ID): Description
31:24	7h RO	**Base Class Code (BCC):** Indicates the base class code of the host controller device.
23:16	80h RO	**Sub Class Code (SCC):** Indicates the sub class code of the host controller device.
15:8	0h RO	**Programming Interface (PI):** Indicates the programming interface of the host controller device.
7:0	0h RO/V	**Revision ID (RID):** Indicates stepping of the host controller. Refer to Device and Revision ID table for specific value.

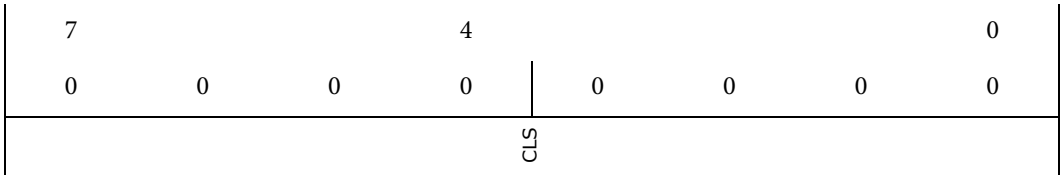
30.1.5 Cache Line Size (HECI1_CLS)—Offset Ch

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 22
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO	**Cache Line Size (CLS):** Not implemented, hardwired to 0.

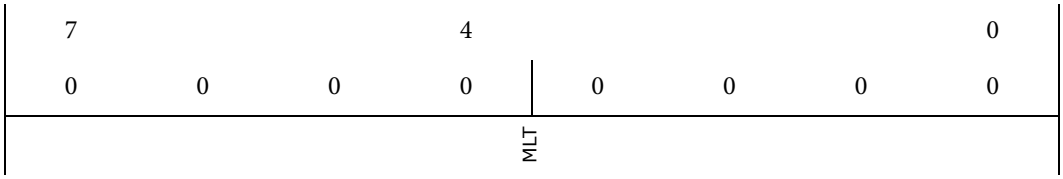
30.1.6 Master Latency Timer (HECI1_MLT)—Offset Dh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 22
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO	**Master Latency Timer (MLT):** Not implemented, hardwired to 0.

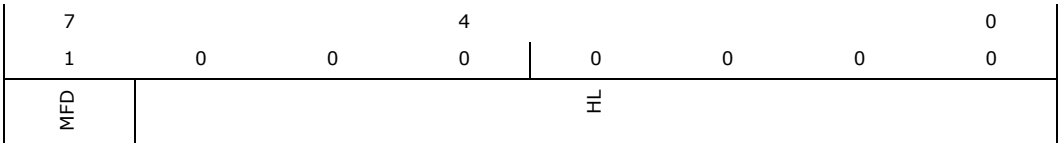
30.1.7 Header Type (HECI1_HTYPE)—Offset Eh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 22
Function: 0

Default: 80h



Bit Range	Default and Access	Field Name (ID): Description
7	1h RO	**Multi-Function Device (MFD):** Indicates the host controller is part of a multi-function device.
6:0	0h RO	**Header Layout (HL):** Indicates that the host controller uses a target device layout.

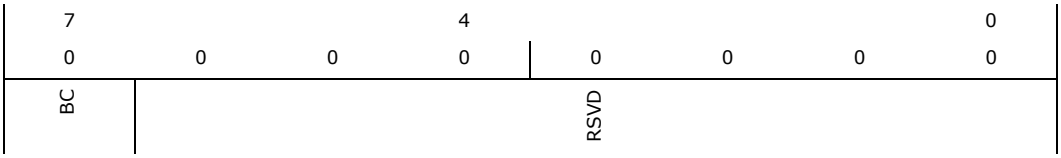
30.1.8 Built In Self-Test (HECI1_BIST)—Offset Fh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 22
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7	0h RO	**BIST Capable (BC):** Not implemented, hardwired to 0.
6:0	0h RO	Reserved.

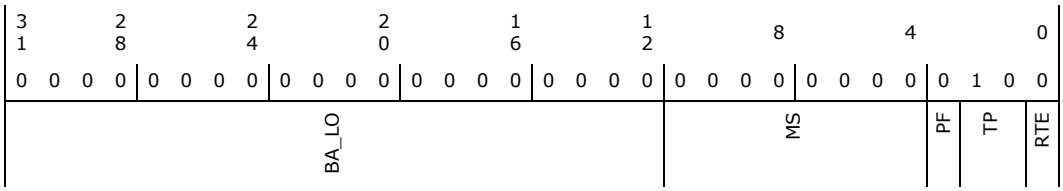
30.1.9 HECI MMIO Base Address Low (HECI1_MMIO_MBAR_LO)—Offset 10h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 0

Default: 4h



Bit Range	Default and Access	Field Name (ID): Description
31:12	0h RW	**Base Address Low (BA_LO):** Lower 32 bits of base address of register memory space.
11:4	0h RO	**Memory Size (MS):** This field is hardwired to 0 to indicate that HECI device requests 4KB of memory space.
3	0h RO	**Prefetchable (PF):** Indicates that this range is not pre-fetchable
2:1	2h RO	**Type (TP):** Indicates that this range can be mapped anywhere in 64-bit address space.
0	0h RO	**Resource Type Indicator (RTE):** Indicates a request for register memory space.

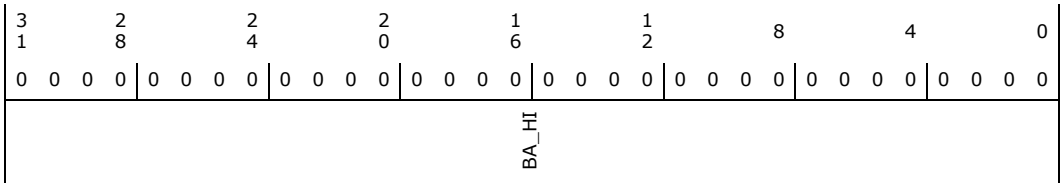
30.1.10 HECI MMIO Base Address High (HECI1_MMIO_MBAR_HI)—Offset 14h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Base Address High (BA_HI):** Upper 32 bits of base address of register memory space.

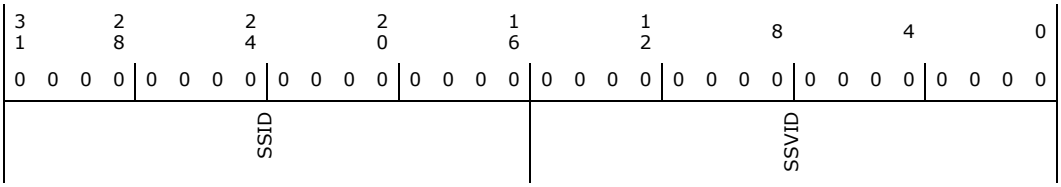
30.1.11 Sub System Identifiers (HECI1_SS)—Offset 2Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RW/O	**Subsystem ID (SSID):** Indicates the sub- system identifier. This field should be programmed by BIOS during boot-up. Once written, this register becomes Read Only.
15:0	0h RW/O	**Subsystem Vendor ID (SSVID):** Indicates the sub-system vendor identifier. This field should be programmed by BIOS during boot- up. Once written, this register becomes Read Only.

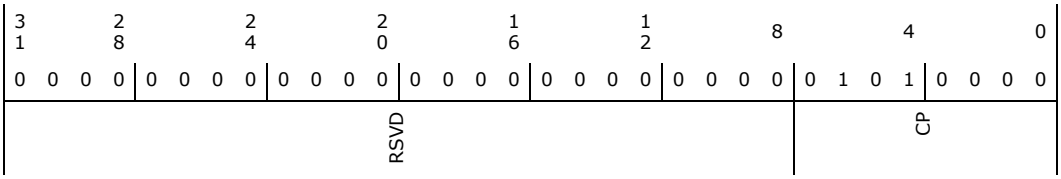
30.1.12 Capabilities Pointer (HECI1_CAP)—Offset 34h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 0

Default: 50h



Bit Range	Default and Access	Field Name (ID): Description
31:8	0h RO	Reserved.
7:0	50h RO	**Capability Pointer (CP):** Indicates the first capability pointer offset. It points to the PCI power management capability offset.

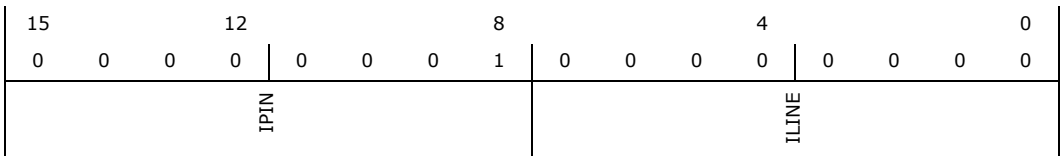
30.1.13 Interrupt Information (HECI1_INTR)—Offset 3Ch

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 22
Function: 0

Default: 100h



Bit Range	Default and Access	Field Name (ID): Description
15:8	1h RO/V	**Interrupt Pin (IPIN):** This field indicates the virtual interrupt pin the host controller uses. A value of 0x1/0x2/0x3/0x4 indicates that this function implements legacy interrupt on INTA/INTB/INTC/INTD, respectively. When legacy interrupt is not used this register should be set to 0x0.
7:0	0h RW	**Interrupt Line (ILINE):** Software written value to indicate which interrupt line (vector) theinterrupt is connected to. No hardware action is taken on this field.

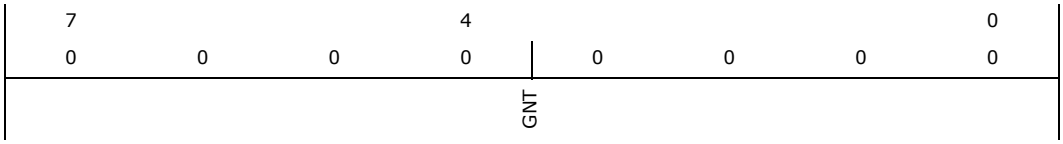
30.1.14 Minimum Grant (HECI1_MGNT)—Offset 3Eh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 22
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO	**Grant (GNT):** Not implemented, hardwired to 0.

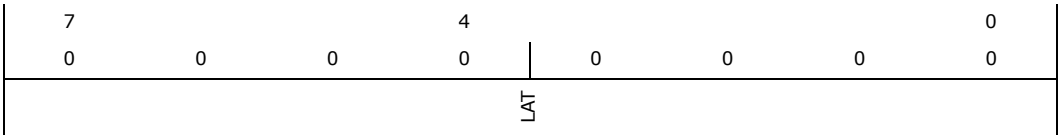
30.1.15 Maximum Latency (HECI1_MLAT)—Offset 3Fh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 22
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO	**Latency (LAT):** Not implemented, hardwired to 0.

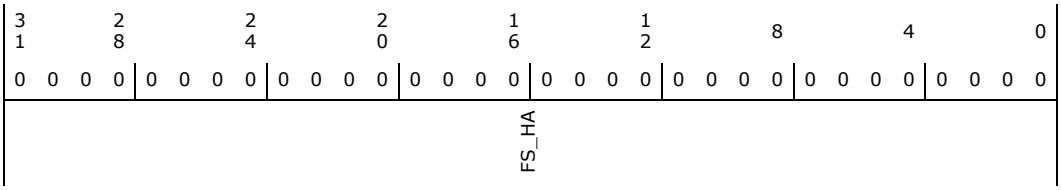
30.1.16 Host Firmware Status Register 1 (HFSTS1)—Offset 40h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RO/V	**Host Firmware Status (FS_HA):** Indicates current status of the firmware for the controller. This field is the host's read only access to the FS field in the Intel ME Firmware Status register. **This field is reset during CSE partition reset flow.**

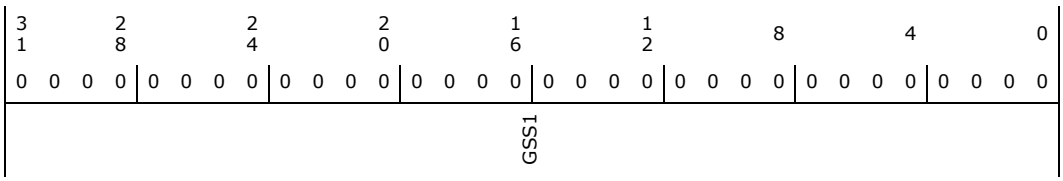
30.1.17 Host Firmware Status Register 2 (HFSTS2)—Offset 48h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RO/V	**Host Firmware Status (GSS1):** This field is host side shadow of General Status 1 (CSE_GS1) register. **This field is reset during Intel ME partition reset flow.**

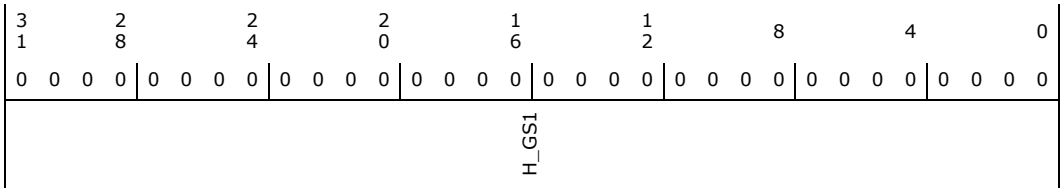
30.1.18 Host General Status (HECI1_H_GS1)—Offset 4Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Host General Status (H_GS1):** General status of Host. This field is not used by hardware.

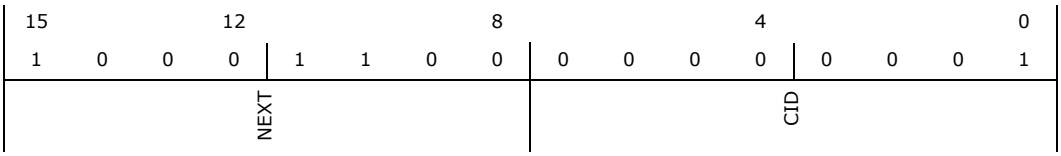
30.1.19 PCI Power Management Capability ID (HECI1_PID)—Offset 50h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 22
Function: 0

Default: 8C01h



Bit Range	Default and Access	Field Name (ID): Description
15:8	8Ch RO	**Next Capability (NEXT):** Indicates the location of the next capability item in the list. This is the Message Signaled Interrupts capability.
7:0	1h RO	**Cap ID (CID):** Indicates that this pointer is a PCI power management.

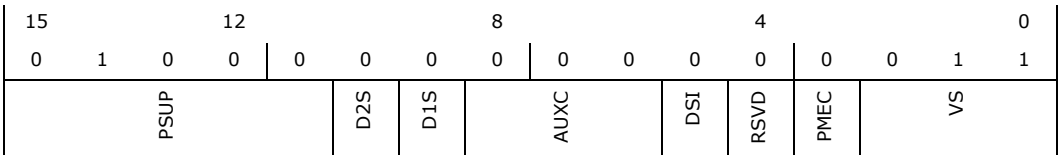
30.1.20 PCI Power Management Capabilities (HECI1_PC)—Offset 52h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 22
Function: 0

Default: 4003h



Bit Range	Default and Access	Field Name (ID): Description
15:11	8h RO	**PME Support (PSUP):** Indicates the states that can generate PME#. The controller can assert PME# from $D3_{HOT}$ only.
10	0h RO	**D2 Support (D2S):** The D2 state is not supported for the host controller.
9	0h RO	**D _Support (D1S):** The D1 state is not supported for the host controller.
8:6	0h RO	**Aux_Current (AUXC):** Reports the maximum Suspend well current required when in the $D3_{COLD}$ state. Value of 0 is reported.
5	0h RO	**Device Specific Initialization (DSI):** Indicates whether device-specific initialization is required.
4	0h RO	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
3	0h RO	**PME Clock (PMEC):** Indicates that PCI clock is not required to generate PME#.
2:0	3h RO	**Version (VS):** Indicates support for Revision 1.2 of the PCI Power Management Specification.

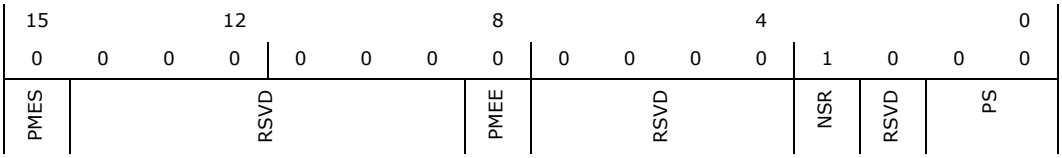
30.1.21 PCI Power Management Control and Status (HECI1_PMCS)—Offset 54h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 22
Function: 0

Default: 8h



Bit Range	Default and Access	Field Name (ID): Description
15	0h RW/1C/V	**PME Status (PMES):** The PME Status bit can be set to '1' by the FW. This bit is cleared by host CPU writing a '1' to it. FW cannot clear this bit. Host CPU writes with value '0' have no effect on this bit.
14:9	0h RO	Reserved.
8	0h RW	**PME Enable (PMEE):** When set, PME_assert and PME_deassert messages are sent over Sideband to PMC based on the PMES bit. When reset these messages may not be sent.
7:4	0h RO	Reserved.
3	1h RO	**No Soft Reset (NSR):** This bit indicates that when the controller is transitioning from $D3_{HOT}$ to D0 due to power state command, it does not perform and internal reset.
2	0h RO	Reserved.
1:0	0h RW	**Power State (PS):** This field is used both to determine the current power state of the controller and to set a new power state. The values are: 00 - D0 state 11 - $D3_{HOT}$ state. The D1 and D2 states are not supported for this controller. If software attempts to write an unsupported, optional state to this field, the write operation must complete normally on the bus; however, the data is discarded and no state change occurs. When in the $D3_{HOT}$ state, the HBA's configuration space is available, but the registers memory space is not. Additionally, interrupts are blocked.

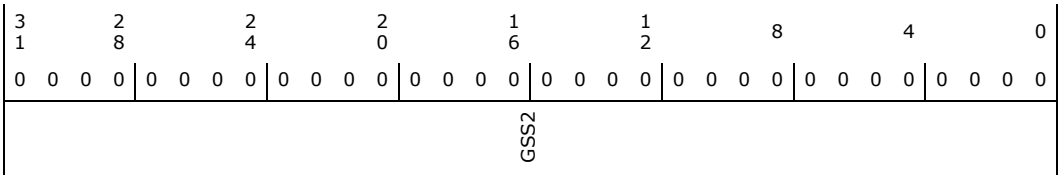
30.1.22 Host Firmware Status Register 3 (HFSTS3)—Offset 60h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RO/V	**Host Firmware Status (GSS2):** This field is host side shadow of Intel ME General Status 2 (CSE_GS2). **This field is reset during Intel ME partition reset flow.**

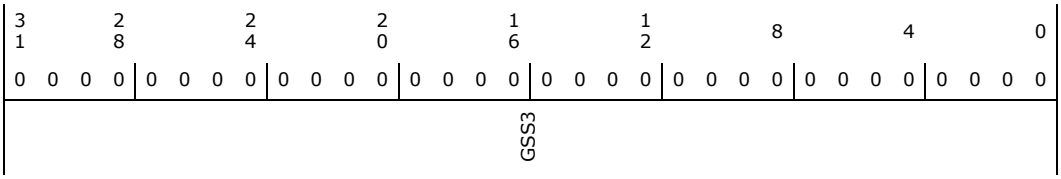
30.1.23 Host Firmware Status Register 4 (HFSTS4)—Offset 64h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RO/V	**Host Firmware Status (GSS3):** This field is host side shadow of CSE General Status 3 (CSE_GS3). **This field is reset during Intel ME partition reset flow.**

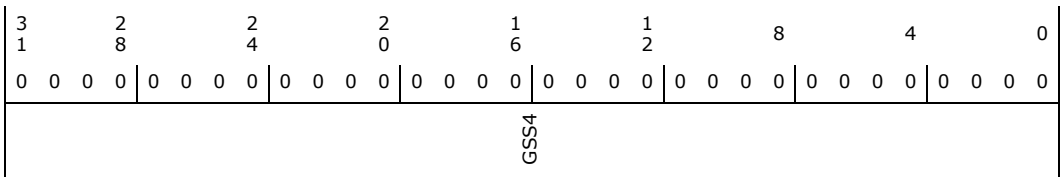
30.1.24 Host Firmware Status Register 5 (HFSTS5)—Offset 68h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RO/V	**Host Firmware Status (GSS4):** This field is host side shadow of Intel ME General Status 4 (CSE_GS4). **This field is reset during Intel ME partition reset flow.**

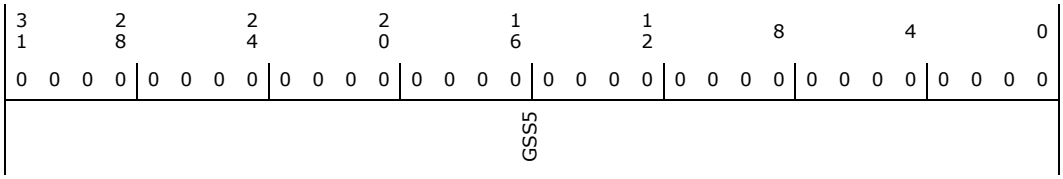
30.1.25 Host Firmware Status Register 6 (HFSTS6)—Offset 6Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RO/V	**Host Firmware Status (GSS5):** This field is host side shadow of Intel ME General Status 5 (CSE_GS5). **This field is reset during Intel ME partition reset flow.**

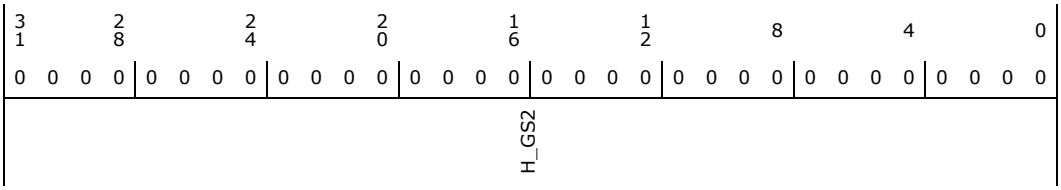
30.1.26 Host General Status 2 (HECI1_H_GS2)—Offset 70h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Host General Status 2 (H_GS2):** General status of Host. This field is not used by hardware.

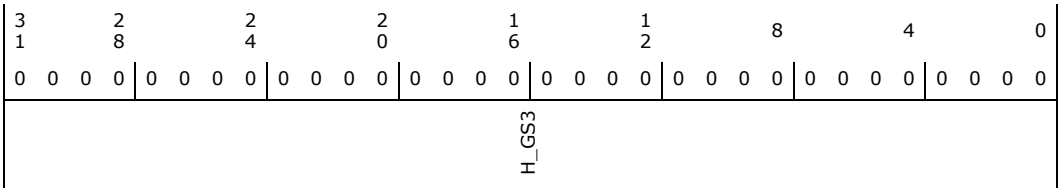
30.1.27 Host General Status 3 (HECI1_H_GS3)—Offset 74h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Host General Status 3 (H_GS3):** General status of Host. This field is not used by hardware.

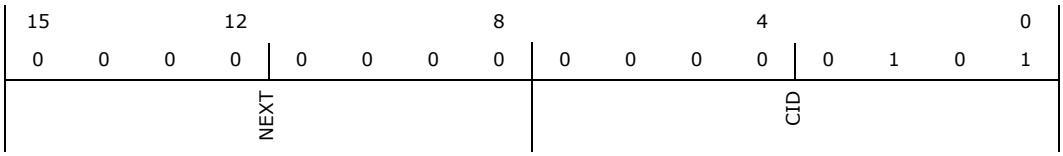
30.1.28 Message Signaled Interrupt Identifiers (HECI1_MID)—Offset 8Ch

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 22
Function: 0

Default: 5h



Bit Range	Default and Access	Field Name (ID): Description
15:8	0h RO	**Next Pointer (NEXT):** Indicates the next item in the list. This can be other capability pointers (such as PCI-X or PCI- Express) or it can be the last item in the list.
7:0	5h RO	**Capability ID (CID):** Indicates MSI.

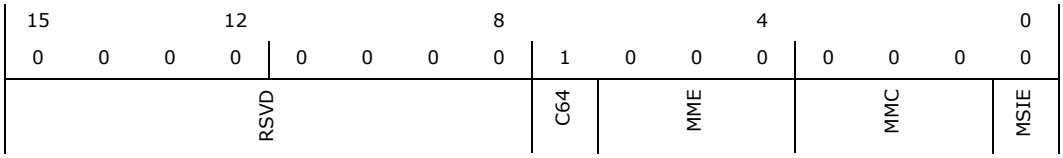
30.1.29 Message Signaled Interrupt Message Control (HECI1_MC)—Offset 8Eh

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 22
Function: 0

Default: 80h



Bit Range	Default and Access	Field Name (ID): Description
15:8	0h RO	Reserved.
7	1h RO	**64 Bit Address Capable (C64):** Specifies whether capable of generating 64-bit messages.
6:4	0h RO	**Multiple Message Enable (MME):** Not implemented, hardwired to 0.
3:1	0h RO	**Multiple Message Capable (MMC):** Not implemented, hardwired to 0.
0	0h RW	**MSI Enable (MSIE):** If set, MSI is enabled and INTx virtual wires are not used to generate interrupts.

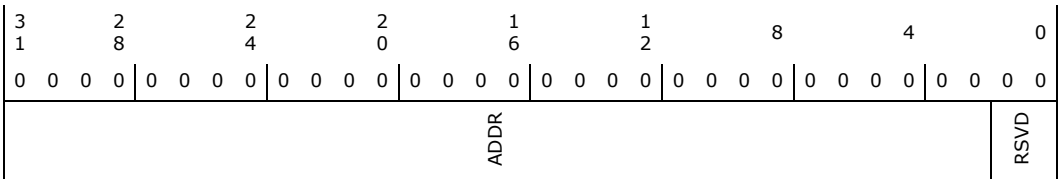
30.1.30 Message Signaled Interrupt Message Address (HECI1_MA)—Offset 90h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:2	0h RW	**Address (ADDR):** Lower 32 bits of the system specified message address, always DW aligned.
1:0	0h RO	Reserved.

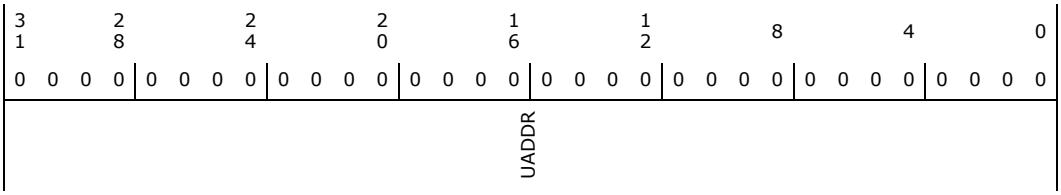
30.1.31 Message Signaled Interrupt Upper Address (HECI1_MUA)—Offset 94h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Upper Address (UADDR):** Upper 32 bits of the system specified message address. This register is optional and only implemented if MC.C64=1.

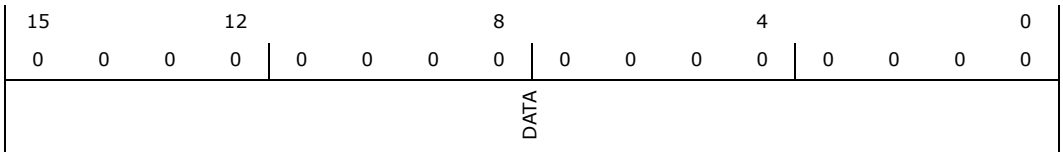
30.1.32 Message Signaled Interrupt Message Data (HECI1_MD)—Offset 98h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 22
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:0	0h RW	**Data (DATA):** This 16-bit field is programmed by system software if MSI is enabled.

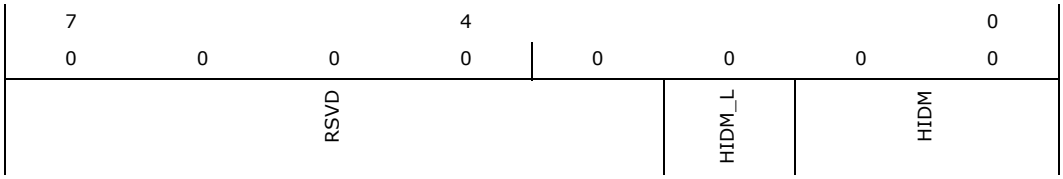
30.1.33 HECI Interrupt Delivery Mode (HECI1_HIDM)—Offset A0h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 22
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:3	0h RO	Reserved.
2	0h RW/1S	**HIDM Lock (HIDM_L):** Writing 1 to this bit locks the HIDM field.
1:0	0h RW/L	**HECI Interrupt Delivery Mode (HIDM):** These bits control what type of interrupt the controller will send when Intel ME FW writes to set the CSE_IG bit. They are interpreted as follows: 00 - Generate Legacy or MSI interrupt; 01 - Generate SCI; 10 - Generate SMI; This field may be locked by writing 1 to HIDM_L bit.

30.2 Intel MEI MMIO Registers Summary

Table 30-2. Summary of Intel MEI MMIO Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
800h	803h	D0i3 Control (HECI1_D0I3C)—Offset 800h	0h

30.2.1 D0i3 Control (HECI1_D0I3C)—Offset 800h

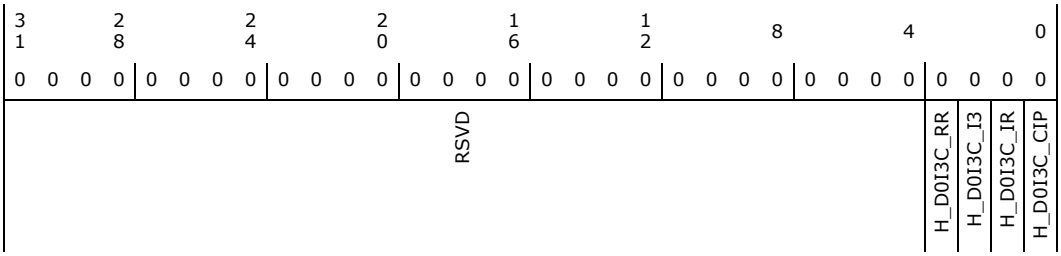
This register allows host to configure the power mode using D0i0/D0i3 support.

Access Method

Type: MEM Register
(Size: 32 bits)

Device: 22
Function: 0

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:4	0h RO	Reserved.
3	0h RO	**Restore Required (H_D0I3C_RR):** SW sets this bit to 1 in order to move the interface into the D0i3 state. Clearing this bit will return the interface into the fully active D0 state (D0i0).
2	0h RW	**D0i3 (H_D0I3C_I3):** SW sets this bit to 1 in order to move the IP into the D0i3 state. Clearing this bit will return the IP into the fully active D0 state (D0i0). When this bit changes state, mIA may be interrupted (see H_PCI_CSR.D0I3C_IS register's description)
1	0h RW	**Interrupt Request (H_D0I3C_IR):** SW sets this bit to 1 to ask for an interrupt to be generated on completion of the command. SW must clear or set this bit on each write to this register.
0	0h RO/V	**Command-in-Progress (H_D0I3C_CIP):** HW sets this bit on a 0->1 or 1->0 transition of D0i3 bit (bit [2]). While set, the other bits in this register are not valid and it is illegal for SW to write to any of them. While clear, all other bits in this register are valid and SW may write to them. If Interrupt Request (bit [1]) was set for the current command, HW may clear this bit before the interrupt has been made visible to SW, since when SW actually handles a particular interrupt is not visible to HW. SW writes to this bit have no effect. This bit is auto-cleared by HW one cycle after it is set.

§

31 IDE Redirect (IDE-R) (D22:F2)

31.1 IDE Redirect PCI Configuration (D22:F2) Registers Summary

Table 31-1. Summary of IDE Redirect PCI Configuration (D22:F2) Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	Device ID and Vendor ID (IDE_HOST_DID_VID)—Offset 0h	X_8086h
4h	7h	Status and Command (IDE_HOST_STS_CMD)—Offset 4h	B00000h
8h	Bh	Class Code and Revision ID (IDE_HOST_CC_RID)—Offset 8h	10185xxh
Ch	Fh	BIST, Header Type, Latency Timer, and Cache Line Size (IDE_HOST_BIST_HTYPE_LT_CLS)—Offset Ch	800000h
10h	13h	IDE Primary Command Block IO BAR (IDE_HOST_PCMDIOBAR)—Offset 10h	1h
14h	17h	IDE Primary Control Block IO BAR (IDE_HOST_PCTLIOBAR)—Offset 14h	1h
18h	1Bh	IDE Secondary Command Block IO BAR (IDE_HOST_SCMDIOBAR)—Offset 18h	1h
1Ch	1Fh	IDE Secondary Control Block IO BAR (IDE_HOST_SCTLIOBAR)—Offset 1Ch	1h
20h	23h	IDE Bus Master Block IO BAR (IDE_HOST_BMIOBAR)—Offset 20h	1h
2Ch	2Fh	Subsystem ID and Subsystem Vendor ID (IDE_HOST_SID_SVID)—Offset 2Ch	8086h
34h	37h	Capabilities List Pointer (IDE_HOST_CAPP)—Offset 34h	40h
3Ch	3Fh	Maximum Latency, Minimum Grant, Interrupt Pin and Interrupt Line (IDE_HOST_MAXL_MING_INTP_INTL)—Offset 3Ch	0h
40h	43h	MSI Message Control, Next Pointer and Capability ID (IDE_HOST_MSIMC_MSINP_MSICID)—Offset 40h	805005h
44h	47h	MSI Message Address (IDE_HOST_MSIMA)—Offset 44h	0h
48h	4Bh	MSI Message Upper Address (IDE_HOST_MSIMUA)—Offset 48h	0h
4Ch	4Fh	MSI Message Data (IDE_HOST_MSIMD)—Offset 4Ch	0h
50h	53h	Power Management Capabilities, Next Pointer and Capability ID (IDE_HOST_PMCAP_PMNP_PMCID)—Offset 50h	230001h
54h	57h	Power Management Data, Control/Status Register Bridge Support Extensions, Control and Status (IDE_HOST_PMD_PMCSRBSE_PMCSR)—Offset 54h	8h

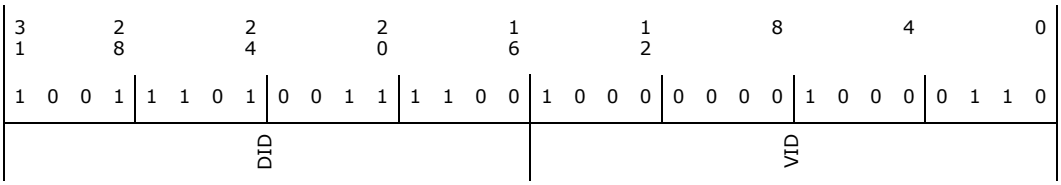
31.1.1 Device ID and Vendor ID (IDE_HOST_DID_VID)—Offset 0h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 2

Default: xxxx8086h



Bit Range	Default and Access	Field Name (ID): Description
31:16	-- RO/V	**Device ID (DID):** This field identifies the particular device. See the Device and Version ID Table in Volume 1 for the default value.
15:0	8086h RO	**Vendor ID (VID):** This field identifies the manufacturer of the device. The value of 0x8086 indicates Intel.

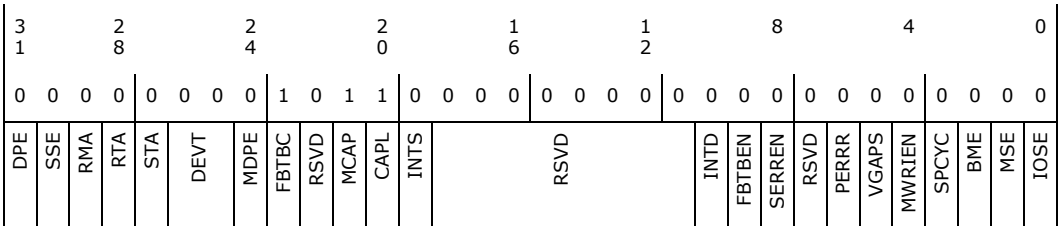
31.1.2 Status and Command (IDE_HOST_STS_CMD)—Offset 4h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 2

Default: B00000h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RO	**Detected Parity Error (DPE):** Not implemented. Hardwired to 0.
30	0h RO	**Signaled System Error (SSE):** Not implemented. Hardwired to 0.
29	0h RW/1C/V	**Received Master Abort (RMA):** This bit must be set by a master device whenever its transaction (except for Special Cycle) is completed with Unsupported Request Completion Status (a.k.a. Master-Abort). All master devices must implement this bit.
28	0h RW/1C/V	**Received Target Abort (RTA):** This bit must be set by a master device whenever its transaction is completed with Completer Abort Completion Status (a.k.a. Target-Abort). All master devices must implement this bit.

Bit Range	Default and Access	Field Name (ID): Description
27	0h RW/1C/V	**Signaled Target Abort (STA):** This bit must be set by a target device whenever it completes a Posted or Non-Posted transaction with a Completer Abort (a.k.a. Target-Abort) error. Devices that will never signal Completer Abort (a.k.a. Target-Abort) do not need to implement this bit.
26:25	0h RO	**Devsel Timing (DEVT):** These bits encode the timing of DEVSEL#. There are three allowable timings for assertion of DEVSEL# as described below: 00b: fast; 01b: medium; 10b: slow; 11b: reserved. These bits are read-only and must indicate the slowest time that a device asserts DEVSEL# for any bus command except Configuration Read and Configuration Write. Hardwired to 00b.
24	0h RO	**Master Data Parity Error (MDPE):** Not implemented. Hardwired to 0.
23	1h RO	**Fast Back to Back Capable (FBTBC):** This bit indicates whether or not the target is capable of accepting fast back-to-back transactions when the transactions are not to the same agent. This bit can be set to 1 if the device can accept these transactions and must be set to 0 otherwise. Hardwired to 1.
22	0h RO	Reserved.
21	1h RO	**66 MHz Capable (MCAP):** This bit indicates whether or not this device is capable of running at 66 MHz. A value of 0 indicates 33 MHz. A value of 1 indicates that the device is 66 MHz capable. Hardwired to 1.
20	1h RO	**Capabilities List (CAPL):** This optional read-only bit indicates whether or not this device implements the pointer for a New Capabilities linked list at offset 34h. A value of zero indicates that no New Capabilities linked list is available. A value of one indicates that the value read at offset 34h is a pointer in Configuration Space to a linked list of new capabilities.
19	0h RO	**Interrupt Status (INTS):** This read-only bit reflects the state of the interrupt in the device/function. Only when the Interrupt Disable bit in the command register is a 0 and this Interrupt Status bit is a 1, will the device's/function's INTx# signal be asserted. Setting the Interrupt Disable bit to a 1 has no effect on the state of this bit. Read-only and hardwired to 0 for a device that does NOT support pin-based interrupt.
18:11	0h RO	Reserved.
10	0h RW	**Interrupt Disable (INTD):** This bit disables the device/function from asserting INTx#. A value of 0 enables the assertion of its INTx# signal. A value of 1 disables the assertion of its INTx# signal. This bit's state after RST# is 0.
9	0h RO	**Fast Back to Back Enable (FBTBEN):** Not implemented. Hardwired to 0.
8	0h RO	**System Error Enable (SERREN):** Not implemented. Hardwired to 0.
7	0h RO	Reserved.
6	0h RO	**Parity Error Response (PERRR):** Not implemented. Hardwired to 0.
5	0h RO	**VGA Palette Snoop (VGAPS):** Not implemented. Hardwired to 0.
4	0h RO	**Memory Write and Invalidate Enable (MWRIEN):** Not implemented. Hardwired to 0.
3	0h RO	**Special Cycles (SPCYC):** Not implemented. Hardwired to 0.
2	0h RW	**Bus Master Enable (BME):** This bit controls the PCI device's ability to act as a master for data transfers. This bit does not impact the generation of completions for split transaction commands.

Bit Range	Default and Access	Field Name (ID): Description
1	0h RO	**Memory Space Enable (MSE):** Read-only and hardwired to 0 because IDE does NOT support Memory Space accesses.
0	0h RW	**IO Space Enable (IOSE):** Controls a device's response to I/O Space accesses. A value of 0 disables the device response. A value of 1 allows the device to respond to I/O Space accesses. State after RST# is 0.

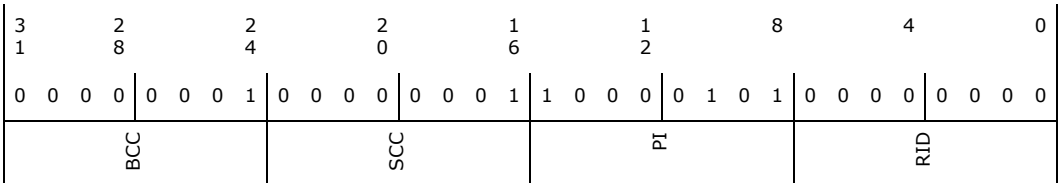
31.1.3 Class Code and Revision ID (IDE_HOST_CC_RID)—Offset 8h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 2

Default: 10185xxh



Bit Range	Default and Access	Field Name (ID): Description
31:24	1h RO	**Base Class Code (BCC):** Identifies the Base Class Code of an external IDE controller device driver.
23:16	1h RO	**Sub-Class Code (SCC):** Identifies the Sub-Class Code of an external IDE controller device driver.
15:8	85h RO	**Programming Interface (PI):** Identifies the Programming Interface of an IDE controller device driver.
7:0	-- RO/V	**Revision ID (RID):** This register specifies a device specific revision identifier. Refer to Device and Revision ID table in Vol1 for specific value.

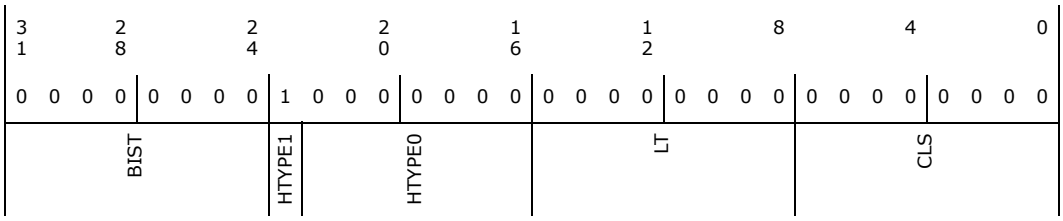
31.1.4 BIST, Header Type, Latency Timer, and Cache Line Size (IDE_HOST_BIST_HTYPE_LT_CLS)—Offset Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 2

Default: 800000h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	**Built In Self Test (BIST):** Not implemented. Hardwired to 0.
23	1h RO/V	**Header Type 1 (HTYPE1):** This bit identifies whether or not the device contains multiple functions. - If the bit is 0, then the device is single function. - If the bit is 1, then the device has multiple functions.
22:16	0h RO	**Header Type 0 (HTYPE0):** Hardwired to 0 to identify the non-bridge Configuration Space Header.
15:8	0h RO	**Latency Timer (LT):** Not implemented. Hardwired to 0.
7:0	0h RO	**Cache Line Size (CLS):** Not implemented. Hardwired to 0.

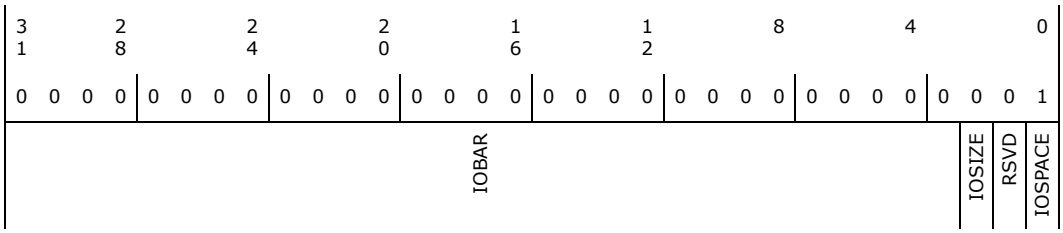
31.1.5 IDE Primary Command Block IO BAR (IDE_HOST_PCMDIOBAR)—Offset 10h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 2

Default: 1h



Bit Range	Default and Access	Field Name (ID): Description
31:3	0h RW	**IO BAR (IOBAR):** Software programs this space with the base address of the device's I/O region
2	0h RO	**IO Size (IOSIZE):** Hardwired to 0 to indicate 8B of I/O space
1	0h RO	Reserved.
0	1h RO	**IO Space Indicator (IOSPACE):** Hardwired to 1 to identify an I/O BAR.

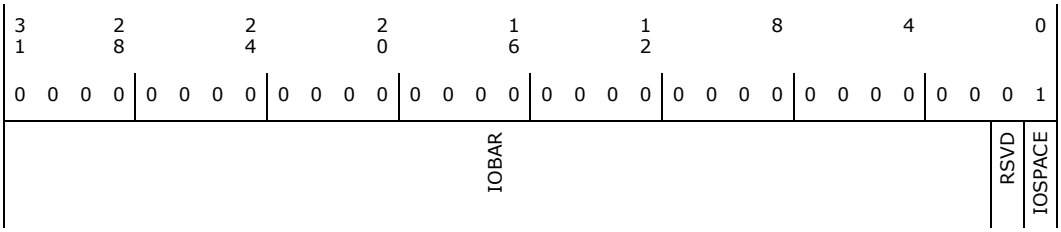
31.1.6 IDE Primary Control Block IO BAR (IDE_HOST_PCTLIOBAR)—Offset 14h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 2

Default: 1h



Bit Range	Default and Access	Field Name (ID): Description
31:2	0h RW	**IO BAR (IOBAR):** Software programs this space with the base address of the device's I/O region
1	0h RO	Reserved.
0	1h RO	**IO Space Indicator (IOSPACE):** Hardwired to 1 to identify an I/O BAR.

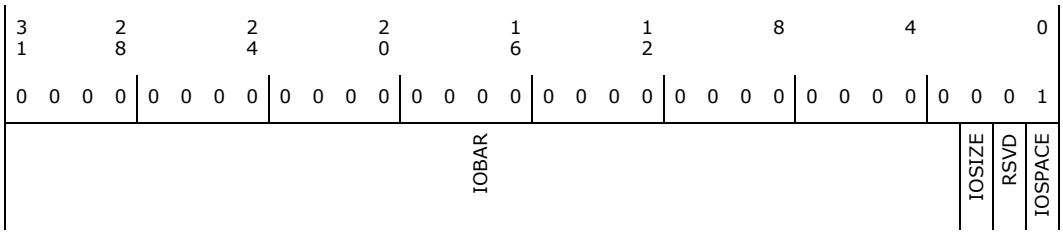
31.1.7 IDE Secondary Command Block IO BAR (IDE_HOST_SCMDIOBAR)—Offset 18h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 2

Default: 1h



Bit Range	Default and Access	Field Name (ID): Description
31:3	0h RW	**IO BAR (IOBAR):** Software programs this space with the base address of the device's I/O region
2	0h RO	**IO Size (IOSIZE):** Hardwired to 0 to indicate 8B of I/O space
1	0h RO	Reserved.
0	1h RO	**IO Space Indicator (IOSPACE):** Hardwired to 1 to identify an I/O BAR.

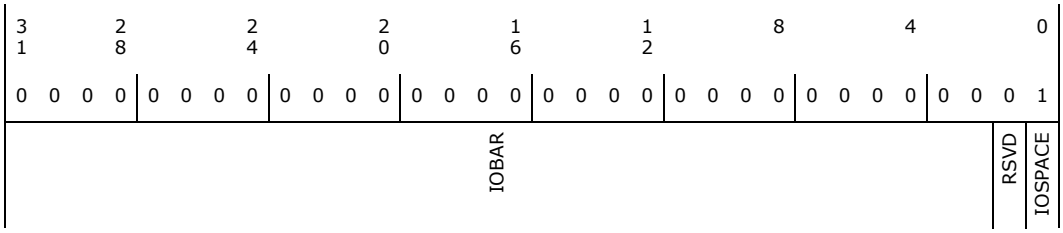
31.1.8 IDE Secondary Control Block IO BAR (IDE_HOST_SCTLIOBAR)—Offset 1Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 2

Default: 1h



Bit Range	Default and Access	Field Name (ID): Description
31:2	0h RW	**IO BAR (IOBAR):** Software programs this space with the base address of the device's I/O region

Bit Range	Default and Access	Field Name (ID): Description
1	0h RO	Reserved.
0	1h RO	**IO Space Indicator (IOSPACE):** Hardwired to 1 to identify an I/O BAR.

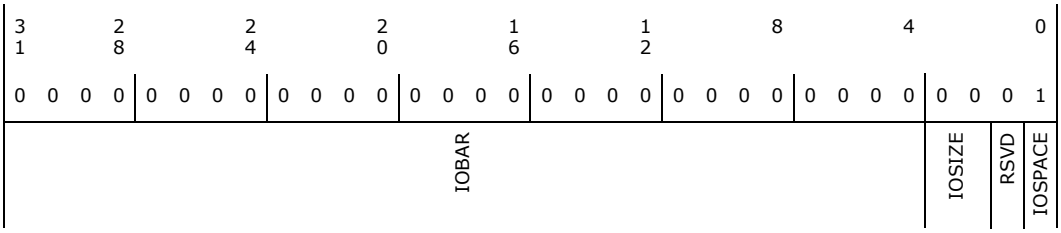
31.1.9 IDE Bus Master Block IO BAR (IDE_HOST_BMIOBAR)—Offset 20h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 2

Default: 1h



Bit Range	Default and Access	Field Name (ID): Description
31:4	0h RW	**IO BAR (IOBAR):** Software programs this space with the base address of the device's I/O region
3:2	0h RO	**IO Size (IOSIZE):** Hardwired to 0 to indicate 16B of I/O space
1	0h RO	Reserved.
0	1h RO	**IO Space Indicator (IOSPACE):** Hardwired to 1 to identify an I/O BAR.

31.1.10 Subsystem ID and Subsystem Vendor ID (IDE_HOST_SID_SVID)—Offset 2Ch

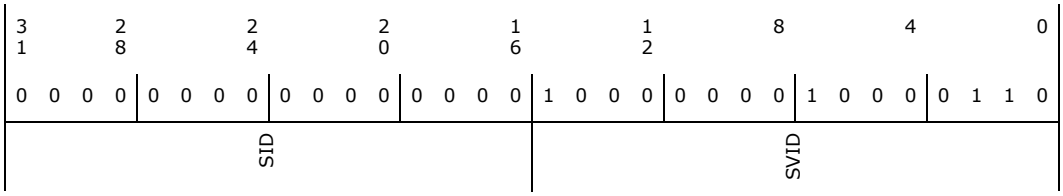
These registers are used to uniquely identify the add-in card or subsystem where the PCI device resides. They provide a mechanism for add-in card vendors to distinguish their add-in cards from one another even though the add-in cards may have the same PCI controller on them (and, therefore, the same Vendor ID and Device ID).

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 2

Default: 8086h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RW/O	**Subsystem ID (SID):** Value of this field is vendor-specific. This is written by BIOS. No hardware action will be taken on this value.
15:0	8086h RW/O	**Subsystem Vendor ID (SVID):** This field identifies the vendor of the add-in card or subsystem. This is written by BIOS. No hardware action will be taken on this value.

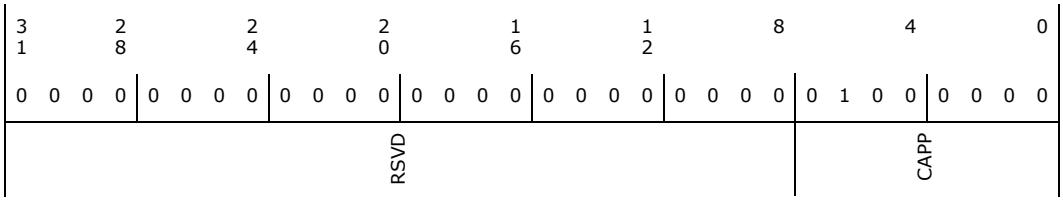
31.1.11 Capabilities List Pointer (IDE_HOST_CAPP)—Offset 34h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 2

Default: 40h



Bit Range	Default and Access	Field Name (ID): Description
31:8	0h RO	Reserved.
7:0	40h RO	**Capabilities Pointer (CAPP):** Indicates the pointer for the first entry in the capabilities list which is the MSI Capability.

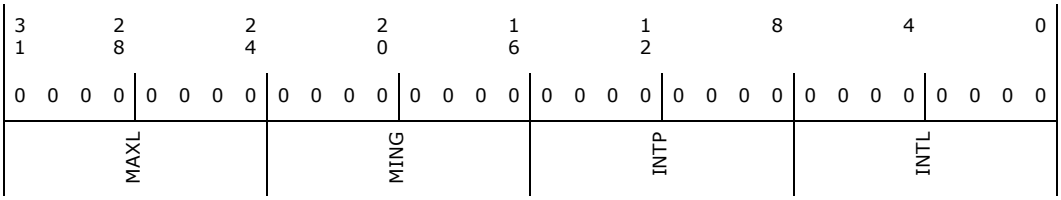
31.1.12 Maximum Latency, Minimum Grant, Interrupt Pin and Interrupt Line (IDE_HOST_MAXL_MING_INTP_INTL)—Offset 3Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	**Maximum Latency (MAXL):** Not implemented. Hardwired to 0.
23:16	0h RO	**Minimum Grant (MING):** Not implemented. Hardwired to 0.
15:8	0h RO/V	**Interrupt Pin (INTP):** This register specifies which interrupt pin IDE uses in PCI interrupt mode. Value Decoding 00h The function does NOT use an interrupt pin. 01h INTA 02h INTB 03h INTC 04h INTD 05h - FFh Reserved.
7:0	0h RW	**Interrupt Line (INTL):** The value written in this register indicates which input of the system interrupt controller, the device's interrupt pin is connected to. This value is used by the operating system and the device driver, and has no affect on the hardware

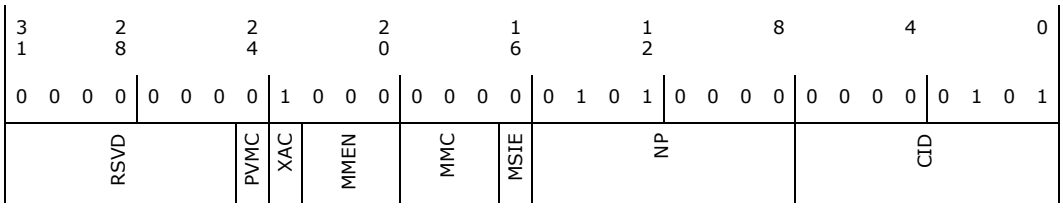
31.1.13 MSI Message Control, Next Pointer and Capability ID (IDE_HOST_MSIMC_MSINP_MSICID)—Offset 40h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 2

Default: 805005h



Bit Range	Default and Access	Field Name (ID): Description
31:25	0h RO	Reserved.
24	0h RO	**Per Vector Masking Capable (PVMC):** Not implemented. Hardwired to 0 to indicate the function does NOT support MSI per-vector masking.
23	1h RO	**64 bit address capable (XAC):** Hardwired to 1 to indicate the function is capable of sending a 64-bit message address.
22:20	0h RW	**Multiple Message Enable (MMEN):** Encoded number of interrupt vectors allocated by SW. This field is RW for software compatibility, but only one interrupt vector is ever sent by the function.
19:17	0h RO	**Multiple Message Capable (MMC):** Encoded number of interrupt vectors requested by a device. Hardwired to 0 to indicate one requested interrupt vector.
16	0h RW	**MSI Enable (MSIE):** If set, MSI interrupt delivery is enabled whereas pin-based interrupt delivery SHALL be disabled. If cleared, prior to returning the configuration write (that clears this field) completion, the function must send any pending MSI(s).
15:8	50h RO	**Next Item Pointer (NP):** Indicates the pointer for the next entry in the capabilities list. Hardwired to 50h to point to the PM Capability list
7:0	5h RO	**Capability ID (CID):** Hardwired to 05h to indicate the linked list item as the MSI Capability registers

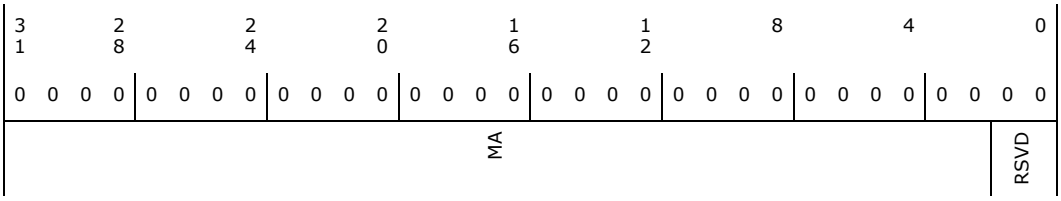
31.1.14 MSI Message Address (IDE_HOST_MSIMA)—Offset 44h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:2	0h RW	**Message Address (MA):** 32-bit DW-aligned MSI message address.
1:0	0h RO	Reserved.

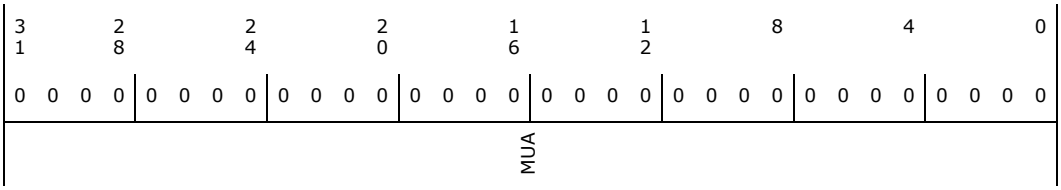
31.1.15 MSI Message Upper Address (IDE_HOST_MSIMUA)—Offset 48h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Message Upper Address (MUA):** Upper 32-bit of a 64-bit DW-aligned MSI message address.

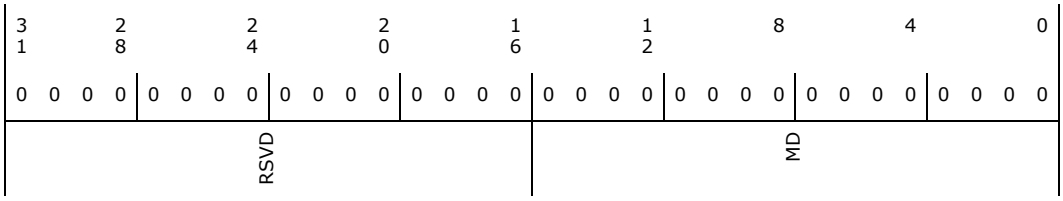
31.1.16 MSI Message Data (IDE_HOST_MSIMD)—Offset 4Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 2

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:0	0h RW	**Message Data (MD):** MSI Message Data

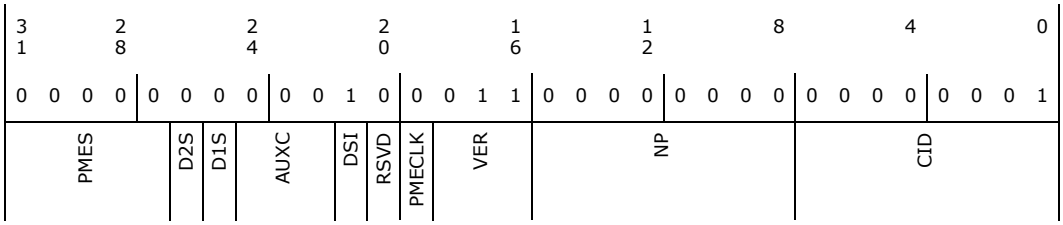
31.1.17 Power Management Capabilities, Next Pointer and Capability ID (IDE_HOST_PMCAP_PMNP_PMCID)—Offset 50h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 2

Default: 230001h



Bit Range	Default and Access	Field Name (ID): Description
31:27	0h RO	**PME Support (PMES):** This 5-bit field indicates the power states in which the function may assert PME#. A value of 0b for any bit indicates that the function is not capable of asserting the PME# signal while in that power state. bit(27) X XXX1b - PME# can be asserted from D0 bit(28) X XX1Xb - PME# can be asserted from D1 bit(29) X X1XXb - PME# can be asserted from D2 bit(30) X 1XXXb - PME# can be asserted from $D3_{HOT}$ bit(31) 1 XXXXb - PME# can be asserted from $D3_{COLD}$
26	0h RO	**D2 Support (D2S):** Hardwired to 0 to indicate that this device does not support D2

Bit Range	Default and Access	Field Name (ID): Description
25	0h RO	**D1 Support (D1S):** Hardwired to 0 to indicate that this device does not support D1
24:22	0h RO	**Aux Current (AUXC):** Not implemented. Hardwired to 0.
21	1h RO	**Device Specific Initialization (DSI):** indicates whether special initialization of this function is required (beyond the standard PCI configuration header) before the generic class device driver is able to use it. A 1 indicates that the function requires a device specific initialization sequence following transition to the D0 uninitialized state. Hardwired to 1 to indicate Device Specific Initialization is required.
20	0h RO	Reserved.
19	0h RO	**PME Clock (PMECLK):** Not implemented. Hardwired to 0.
18:16	3h RO	**Version (VER):** Hardwired to value of 011b indicates that this function complies with Rev. 1.2 of *PCI Power Management Interface Specification*
15:8	0h RO	**Next Item Pointer (NP):** Indicates the pointer for the next entry in the capabilities list. Hardwired to 0 to indicate no more linked list item.
7:0	1h RO	**Capability ID (CID):** Hardwired to 01h to indicate the linked list item as the PCI Power Management registers

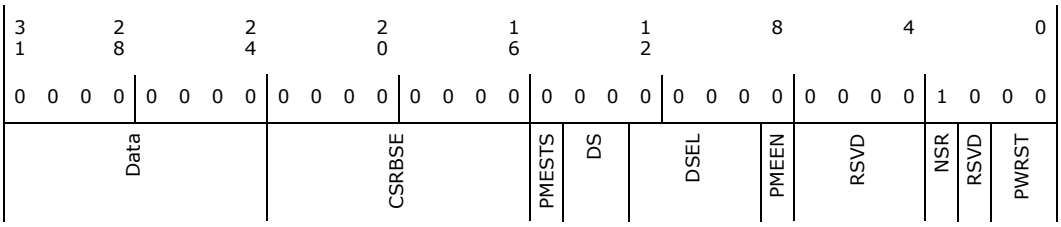
31.1.18 Power Management Data, Control/Status Register Bridge Support Extensions, Control and Status (IDE_HOST_PMD_PMCSRBSE_PMCSR)—Offset 54h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 2

Default: 8h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	**Data (Data):** Not implemented. Hardwired to 0.
23:16	0h RO	**Control/Status Register Bridge Support Extensions (CSRBSE):** Not implemented. Hardwired to 0.
15	0h RO	**PME Status (PMESTS):** Not implemented. Hardwired to 0.
14:13	0h RO	**Data Scale (DS):** Not implemented. Hardwired to 0.

Bit Range	Default and Access	Field Name (ID): Description
12:9	0h RO	**Data Select (DSEL):** Not implemented. Hardwired to 0.
8	0h RO	**PME Enable (PMEEN):** Not implemented. Hardwired to 0.
7:4	0h RO	Reserved.
3	1h RO	**No Soft Reset (NSR):** When set to 1, this bit indicates that devices transitioning from $D3_{HOT}$ to D0 because of PowerState commands do not perform an internal reset. Configuration Context is preserved. Upon transition from the $D3_{HOT}$ to the D0 Initialized state, no additional operating system intervention is required to preserve Configuration Context beyond writing the PowerState bits.
2	0h RO	Reserved.
1:0	0h RW	**Power State (PWRST):** This 2-bit field is used both to determine the current power state of a function and to set the function into a new power state. The definition of the field values is given below: 00b - D0 01b - D1 10b - D2 11b - $D3_{HOT}$ If software attempts to write an unsupported, optional state to this field, the write operation must complete normally on the bus; however, the data is discarded and no state change occurs.

§

32 Keyboard and Text (KT) (D22:F3)

32.1 Serial Port for Remote Keyboard and Text (KT) (D22:F3) Registers Summary

Table 32-1. Summary of Serial Port for Remote Keyboard and Text (KT) (D22:F3) Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	Device ID and Vendor ID (KT_HOST_DID_VID)—Offset 0h	X_8086h
4h	7h	Status and Command (KT_HOST_STS_CMD)—Offset 4h	B00000h
8h	Bh	Class Code and Revision ID (KT_HOST_CC_RID)—Offset 8h	70002XXh
Ch	Fh	BIST, Header Type, Latency Timer, and Cache Line Size (KT_HOST_BIST_HTYPE_LT_CLS)—Offset Ch	800000h
10h	13h	KT IO BAR (KT_HOST_IOBAR)—Offset 10h	1h
14h	17h	KT Memory BAR (KT_HOST_MEMBAR)—Offset 14h	0h
28h	2Bh	Cardbus CIS Pointer (KT_HOST_CCP)—Offset 28h	0h
2Ch	2Fh	Subsystem ID and Subsystem Vendor ID (KT_HOST_SID_SVID)—Offset 2Ch	8086h
30h	33h	Expansion ROM Base Address (KT_HOST_XRBAR)—Offset 30h	0h
34h	37h	Capabilities List Pointer (KT_HOST_CAPP)—Offset 34h	40h
3Ch	3Fh	Maximum Latency, Minimum Grant, Interrupt Pin and Interrupt Line (KT_HOST_MAXL_MING_INTP_INTL)—Offset 3Ch	0h
40h	43h	MSI Message Control, Next Pointer and Capability ID (KT_HOST_MSIMC_MSINP_MSICID)—Offset 40h	805005h
44h	47h	MSI Message Address (KT_HOST_MSIMA)—Offset 44h	0h
48h	4Bh	MSI Message Upper Address (KT_HOST_MSIMUA)—Offset 48h	0h
4Ch	4Fh	MSI Message Data (KT_HOST_MSIMD)—Offset 4Ch	0h
50h	53h	Power Management Capabilities, Next Pointer and Capability ID (KT_HOST_PMCAP_PMNP_PMCID)—Offset 50h	230001h
54h	57h	Power Management Data, Control/Status Register Bridge Support Extensions, Control and Status (KT_HOST_PMD_PMCSRBSE_PMCSR)—Offset 54h	8h

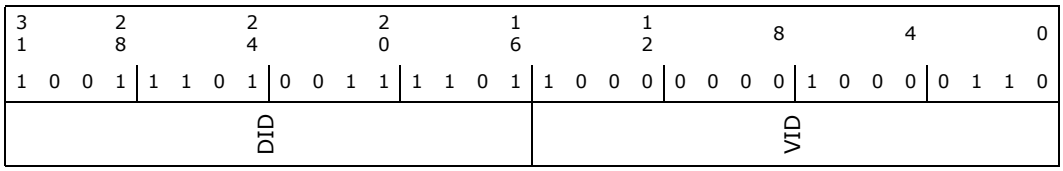
32.1.1 Device ID and Vendor ID (KT_HOST_DID_VID)—Offset 0h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 3

Default: X_8086h



Bit Range	Default and Access	Field Name (ID): Description
31:16	---- RO/V	**Device ID (DID):** This field identifies the particular device. See the Device and Version ID Table in Volume 1 for the default value.
15:0	8086h RO	**Vendor ID (VID):** This field identifies the manufacturer of the device. The value of 0x8086 indicates Intel.

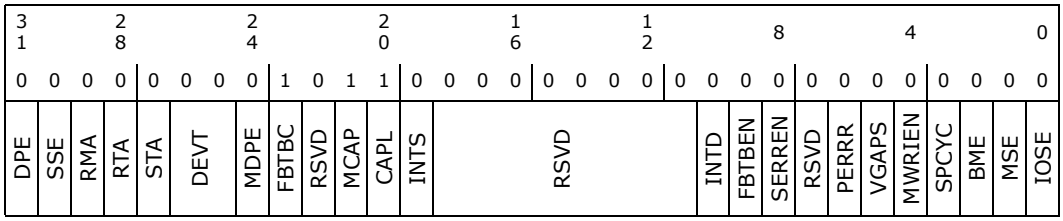
32.1.2 Status and Command (KT_HOST_STS_CMD)—Offset 4h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 3

Default: B00000h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RO	**Detected Parity Error (DPE):** Not implemented. Hardwired to 0.
30	0h RO	**Signaled System Error (SSE):** Not implemented. Hardwired to 0.
29	0h RW/1C/V	**Received Master Abort (RMA):** This bit must be set by a master device whenever its transaction (except for Special Cycle) is completed with Unsupported Request Completion Status (a.k.a. Master-Abort). All master devices must implement this bit.
28	0h RW/1C/V	**Received Target Abort (RTA):** This bit must be set by a master device whenever its transaction is completed with Completer Abort Completion Status (a.k.a. Target-Abort). All master devices must implement this bit.

Bit Range	Default and Access	Field Name (ID): Description
27	0h RW/1C/V	**Signaled Target Abort (STA):** This bit must be set by a target device whenever it completes a Posted or Non-Posted transaction with a Completer Abort (a.k.a. Target-Abort) error. Devices that will never signal Completer Abort (a.k.a. Target-Abort) do not need to implement this bit.
26:25	0h RO	**Devsel Timing (DEVT):** These bits encode the timing of DEVSEL#. There are three allowable timings for assertion of DEVSEL# as described below: 00b: fast; 01b: medium; 10b: slow; 11b: reserved. These bits are read-only and must indicate the slowest time that a device asserts DEVSEL# for any bus command except Configuration Read and Configuration Write. Hardwired to 00b.
24	0h RO	**Master Data Parity Error (MDPE):** Not implemented. Hardwired to 0.
23	1h RO	**Fast Back to Back Capable (FBTBC):** This bit indicates whether or not the target is capable of accepting fast back-to-back transactions when the transactions are not to the same agent. This bit can be set to 1 if the device can accept these transactions and must be set to 0 otherwise. Hardwired to 1.
22	0h RO	Reserved.
21	1h RO	**66 MHz Capable (MCAP):** This bit indicates whether or not this device is capable of running at 66 MHz. A value of 0 indicates 33 MHz. A value of 1 indicates that the device is 66 MHz capable. Hardwired to 1.
20	1h RO	**Capabilities List (CAPL):** This optional read-only bit indicates whether or not this device implements the pointer for a New Capabilities linked list at offset 34h. A value of zero indicates that no New Capabilities linked list is available. A value of one indicates that the value read at offset 34h is a pointer in Configuration Space to a linked list of new capabilities.
19	0h RO	**Interrupt Status (INTS):** This read-only bit reflects the state of the interrupt in the device/function. Only when the Interrupt Disable bit in the command register is a 0 and this Interrupt Status bit is a 1, will the device's/function's INTx# signal be asserted. Setting the Interrupt Disable bit to a 1 has no effect on the state of this bit. Read-only and hardwired to 0 for a device that does NOT support pin-based interrupt.
18:11	0h RO	Reserved.
10	0h RW	**Interrupt Disable (INTD):** This bit disables the device/function from asserting INTx#. A value of 0 enables the assertion of its INTx# signal. A value of 1 disables the assertion of its INTx# signal. This bit's state after RST# is 0.
9	0h RO	**Fast Back to Back Enable (FBTBEN):** Not implemented. Hardwired to 0.
8	0h RO	**System Error Enable (SERREN):** Not implemented. Hardwired to 0.
7	0h RO	Reserved.
6	0h RO	**Parity Error Response (PERRR):** Not implemented. Hardwired to 0.
5	0h RO	**VGA Palette Snoop (VGAPS):** Not implemented. Hardwired to 0.
4	0h RO	**Memory Write and Invalidate Enable (MWRIEN):** Not implemented. Hardwired to 0.
3	0h RO	**Special Cycles (SPCYC):** Not implemented. Hardwired to 0.

Bit Range	Default and Access	Field Name (ID): Description
2	0h RW	**Bus Master Enable (BME):** Controls the ability of a PCI device to issue Memory and I/O Read/Write Requests, and the ability of a PCI bridge to forward Memory and I/O Read/Write Requests in the Upstream direction. Devices: When this bit is Set, the PCI device function is allowed to issue Memory or I/O Requests. When this bit is Clear, the PCI device function is not allowed to issue any Memory or I/O Requests. Note that as MSI/MSI-X interrupt Messages are in-band memory writes, setting the Bus Master Enable bit to 0b disables MSI/MSI-X interrupt Messages as well. Requests other than Memory or I/O Requests are not controlled by this bit. Default value of this bit is 0b. This bit is hardwired to 0b if a PCI device function does not generate Memory or I/O Requests. Bridges: This bit controls forwarding of Memory or I/O Requests by a bridge in the Upstream direction. When this bit is 0b, Memory and I/O Requests received at the Downstream side of a bridge must be handled as Unsupported Requests (UR), and for Non-Posted Requests a Completion with UR completion status must be returned. This bit does not affect forwarding of Completions in either the Upstream or Downstream direction. The forwarding of Requests other than Memory or I/O Requests is not controlled by this bit. Default value of this bit is 0b.
1	0h RW	**Memory Space Enable (MSE):** Controls a device's response to Memory Space accesses. A value of 0 disables the device response. A value of 1 allows the device to respond to Memory Space accesses. State after RST# is 0.
0	0h RW	**IO Space Enable (IOSE):** Controls a device's response to I/O Space accesses. A value of 0 disables the device response. A value of 1 allows the device to respond to I/O Space accesses. State after RST# is 0.

32.1.3 Class Code and Revision ID (KT_HOST_CC_RID)—Offset 8h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 3

Default: 70002xxh

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
BCC								SCC								PI								RID							

Bit Range	Default and Access	Field Name (ID): Description
31:24	7h RO	**Base Class Code (BCC):** Identifies the Base Class Code of an external 16550-compatible serial controller device driver.
23:16	0h RO	**Sub-Class Code (SCC):** Identifies the Sub-Class Code of an external 16550-compatible serial controller device driver.
15:8	2h RO	**Programming Interface (PI):** Identifies the Programming Interface of an external 16550-compatible serial controller device driver.
7:0	---- RO/V	**Revision ID (RID):** Indicates stepping of the host controller. Refer to Device and Revision ID table in Vol1 for specific value.

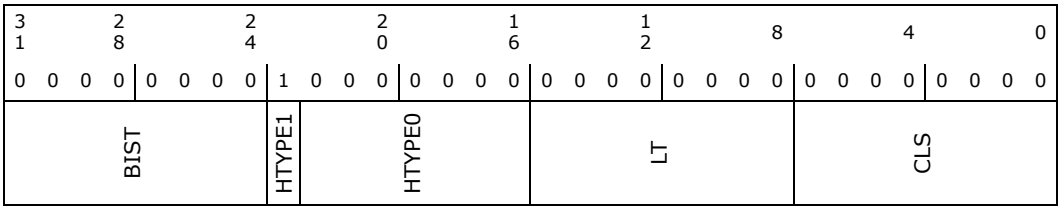
32.1.4 BIST, Header Type, Latency Timer, and Cache Line Size (KT_HOST_BIST_HTYPE_LT_CLS)—Offset Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 3

Default: 800000h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	**Built In Self Test (BIST):** Not implemented. Hardwired to 0.
23	1h RO/V	**Header Type 1 (HTYPE1):** This bit identifies whether or not the device contains multiple functions. - If the bit is 0, then the device is single function. - If the bit is 1, then the device has multiple functions.
22:16	0h RO	**Header Type 0 (HTYPE0):** This field identifies the layout of the second part of the predefined header (beginning at byte 10h in Configuration Space). - The encoding 00h specifies the non-bridge Configuration Space Header. - The encoding 01h specifies the PCI-to-PCI bridge Configuration Space Header. - The encoding 02h specifies the CardBus bridge Configuration Space Header. - All other encodings are reserved. Hardwired to 0 to identify the non-bridge Configuration Space Header.
15:8	0h RO	**Latency Timer (LT):** Not implemented. Hardwired to 0.
7:0	0h RO	**Cache Line Size (CLS):** Not implemented. Hardwired to 0.

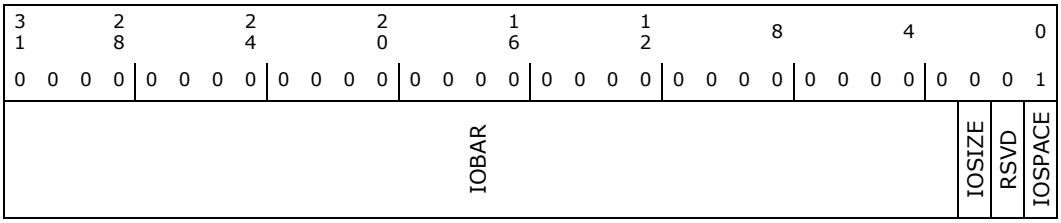
32.1.5 KT IO BAR (KT_HOST_IOBAR)—Offset 10h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 3

Default: 1h



Bit Range	Default and Access	Field Name (ID): Description
31:3	0h RW	**IO BAR (IOBAR):** Software programs this space with the base address of the device's IO region
2	0h RO	**IO Size (IOSIZE):** Hardwired to 0 to indicate 8B of I/O space
1	0h RO	Reserved.
0	1h RO	**IO Space Indicator (IOSPACE):** Hardwired to 1 to identify an I/O BAR.

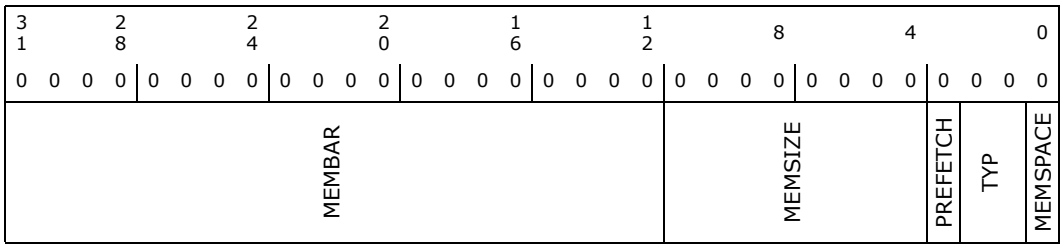
32.1.6 KT Memory BAR (KT_HOST_MEMBAR)—Offset 14h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:12	0h RW	**Memory BAR (MEMBAR):** Software programs this register with the base address of the device's memory region
11:4	0h RO	**Memory Size (MEMSIZE):** Hardwired to 0 to indicate 4KB of memory space
3	0h RO	**Prefetchable (PREFETCH):** A device can mark a range as prefetchable if there are no side effects on reads, the device returns all bytes on reads regardless of the byte enables. Hardwired to 0 to indicate the device's memory space as non-prefetchable.
2:1	0h RO	**Type (TYP):** Hardwired to 0 to indicate that Base register is 32 bits wide and mapping can be done anywhere in the 32-bit Memory Space.
0	0h RO	**Memory Space Indicator (MEMSPACE):** Hardwired to 0 to identify a Memory BAR.

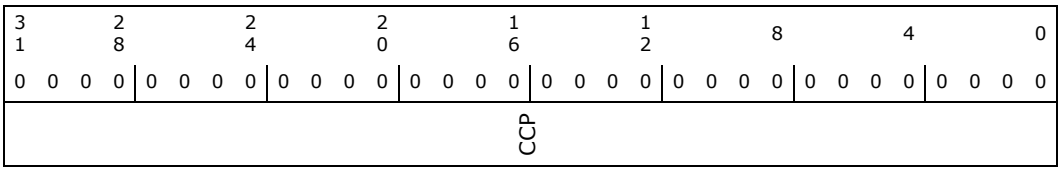
32.1.7 Cardbus CIS Pointer (KT_HOST_CCP)—Offset 28h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RO	**Cardbus CIS Pointer (CCP):** Not implemented. Hardwired to 0.

32.1.8 Subsystem ID and Subsystem Vendor ID (KT_HOST_SID_SVID)—Offset 2Ch

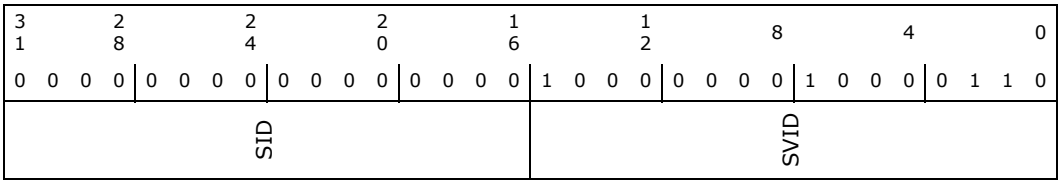
These registers are used to uniquely identify the add-in card or subsystem where the PCI device resides. They provide a mechanism for add-in card vendors to distinguish their add-in cards from one another even though the add-in cards may have the same PCI controller on them (and, therefore, the same Vendor ID and Device ID).

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 3

Default: 8086h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RW/O	**Subsystem ID (SID):** Value of this field is vendor-specific. This is written by BIOS. No hardware action will be taken on this value. **Implementation Note:** The Write-once lock must be implemented per field. SID should have its own lock.
15:0	8086h RW/O	**Subsystem Vendor ID (SVID):** This field identifies the vendor of the add-in card or subsystem. This is written by BIOS. No hardware action will be taken on this value. **Implementation Note:** The Write-once lock must be implemented per field. SVID should have its own lock.

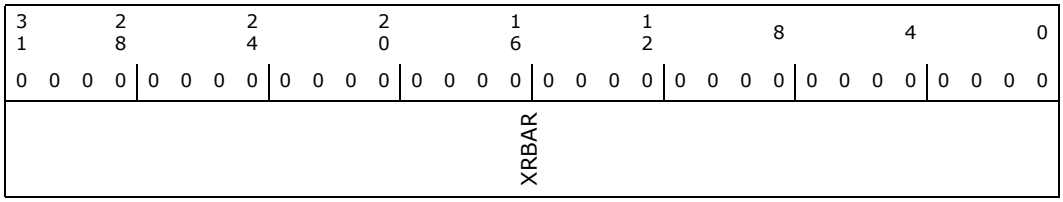
32.1.9 Expansion ROM Base Address (KT_HOST_XRBAR)—Offset 30h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RO	**Expansion ROM Base Address (XRBAR):** Not implemented. Hardwired to 0.

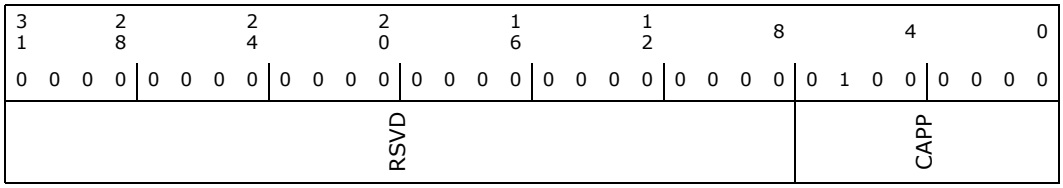
32.1.10 Capabilities List Pointer (KT_HOST_CAPP)—Offset 34h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 3

Default: 40h



Bit Range	Default and Access	Field Name (ID): Description
31:8	0h RO	Reserved.
7:0	40h RO	**Capabilities Pointer (CAPP):** Indicates the pointer for the first entry in the capabilities list which is the MSI Capability.

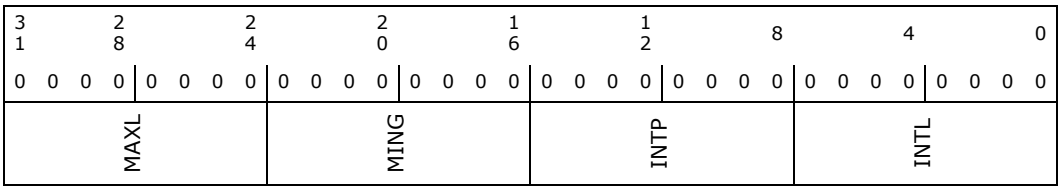
32.1.11 Maximum Latency, Minimum Grant, Interrupt Pin and Interrupt Line (KT_HOST_MAXL_MING_INTP_INTL)—Offset 3Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	**Maximum Latency (MAXL):** Not implemented. Hardwired to 0.
23:16	0h RO	**Minimum Grant (MING):** Not implemented. Hardwired to 0.
15:8	0h RO/V	**Interrupt Pin (INTP):** This register specifies which interrupt pin KT uses in PCI interrupt mode. Value Decoding 00h The function does NOT use an interrupt pin. 01h INTA 02h INTB 03h INTC 04h INTD 05h - FFh Reserved. ***Note:*** This field shadows the KTHIPINR.IPIN field in the PTIO Host private CR space which is configured by BIOS over IOSF SB.
7:0	0h RW	**Interrupt Line (INTL):** This register is used to communicate interrupt line routing information. POST software will write the routing information into this register as it initializes and configures the system. The value in this register tells which input of the system interrupt controller(s) the device's interrupt pin is connected to. The device itself does not use this value, rather it is used by device drivers and operating systems. Device drivers and operating systems can use this information to determine priority and vector information. Values in this register are system architecture specific.

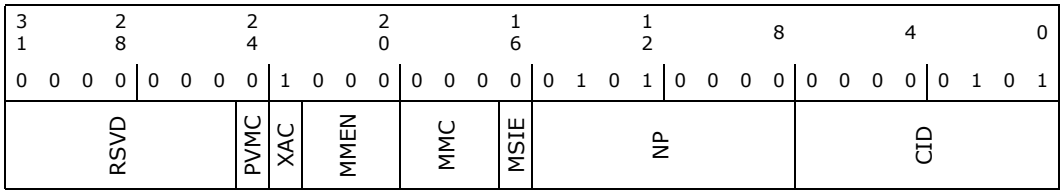
32.1.12 MSI Message Control, Next Pointer and Capability ID (KT_HOST_MSIMC_MSINP_MSICID)—Offset 40h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 3

Default: 805005h



Bit Range	Default and Access	Field Name (ID): Description
31:25	0h RO	Reserved.
24	0h RO	**Per Vector Masking Capable (PVMC):** Not implemented. Hardwired to 0 to indicate the function does NOT support MSI per-vector masking.
23	1h RO	**64 bit address capable (XAC):** Hardwired to 1 to indicate the function is capable of sending a 64-bit message address.
22:20	0h RW	**Multiple Message Enable (MMEN):** Encoded number of interrupt vectors allocated by SW. This field is RW for software compatibility, but only one interrupt vector is ever sent by the function.
19:17	0h RO	**Multiple Message Capable (MMC):** Encoded number of interrupt vectors requested by a device. Hardwired to 0 to indicate one requested interrupt vector.
16	0h RW	**MSI Enable (MSIE):** If set, MSI interrupt delivery is enabled whereas pin-based interrupt delivery SHALL be disabled. If cleared, prior to returning the configuration write (that clears this field) completion, the function must send any pending MSI(s).
15:8	50h RO	**Next Item Pointer (NP):** Indicates the pointer for the next entry in the capabilities list. Hardwired to 50h to point to the PM Capability list
7:0	5h RO	**Capability ID (CID):** Hardwired to 05h to indicate the linked list item as the MSI Capability registers

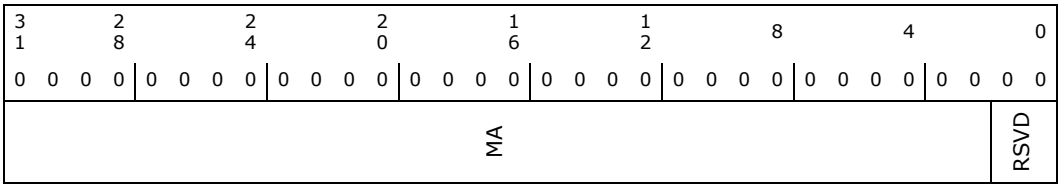
32.1.13 MSI Message Address (KT_HOST_MSIMA)—Offset 44h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:2	0h RW	**Message Address (MA):** 32-bit DW-aligned MSI message address.
1:0	0h RO	Reserved.

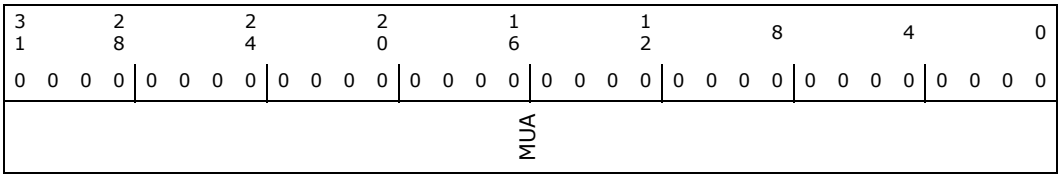
32.1.14 MSI Message Upper Address (KT_HOST_MSIMUA)—Offset 48h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Message Upper Address (MUA):** Upper 32-bit of a 64-bit DW-aligned MSI message address.

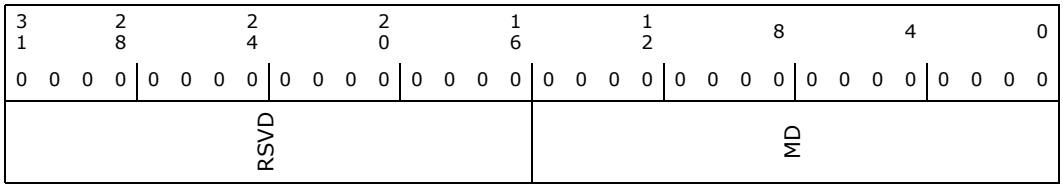
32.1.15 MSI Message Data (KT_HOST_MSIMD)—Offset 4Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 3

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:0	0h RW	**Message Data (MD):** MSI Message Data

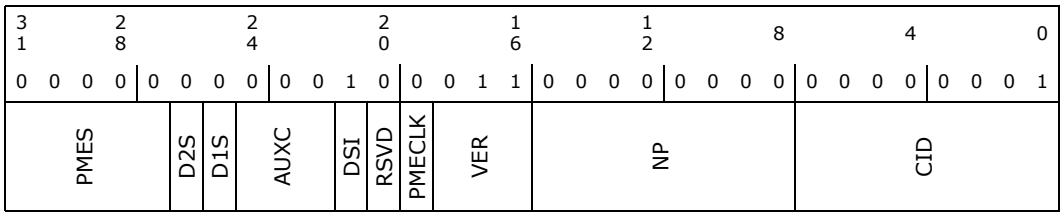
32.1.16 Power Management Capabilities, Next Pointer and Capability ID (KT_HOST_PMCAP_PMNP_PMCID)—Offset 50h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 3

Default: 230001h



Bit Range	Default and Access	Field Name (ID): Description
31:27	0h RO	**PME Support (PMES):** This 5-bit field indicates the power states in which the function may assert PME#. A value of 0b for any bit indicates that the function is not capable of asserting the PME# signal while in that power state. bit(27) X XXX1b - PME# can be asserted from D0 bit(28) X XX1Xb - PME# can be asserted from D1 bit(29) X X1XXb - PME# can be asserted from D2 bit(30) X 1XXXb - PME# can be asserted from D3HOT bit(31) 1 XXXXb - PME# can be asserted from $D3_{COLD}$
26	0h RO	**D2 Support (D2S):** Hardwired to 0 to indicate that this device does not support D2
25	0h RO	**D1 Support (D1S):** Hardwired to 0 to indicate that this device does not support D1
24:22	0h RO	**Aux Current (AUXC):** Not implemented. Hardwired to 0.
21	1h RO	**Device Specific Initialization (DSI):** indicates whether special initialization of this function is required (beyond the standard PCI configuration header) before the generic class device driver is able to use it. A 1 indicates that the function requires a device specific initialization sequence following transition to the D0 uninitialized state. Hardwired to 1 to indicate Device Specific Initialization is required.
20	0h RO	Reserved.
19	0h RO	**PME Clock (PMECLK):** Not implemented. Hardwired to 0.
18:16	3h RO	**Version (VER):** Hardwired to value of 011b indicates that this function complies with Rev 1.2 of the *PCI Power Management Interface Specification*
15:8	0h RO	**Next Item Pointer (NP):** Indicates the pointer for the next entry in the capabilities list. Hardwired to 0 to indicate no more linked list item.
7:0	1h RO	**Capability ID (CID):** Hardwired to 01h to indicate the linked list item as the PCI Power Management registers

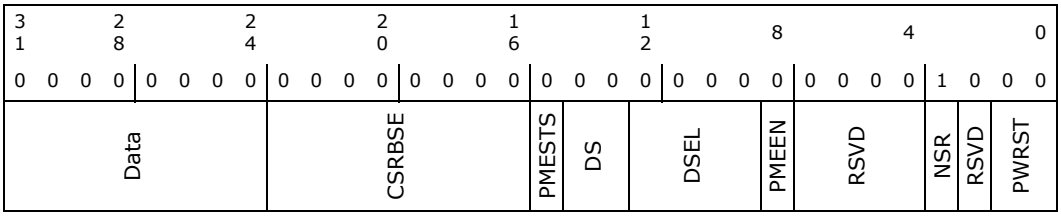
32.1.17 Power Management Data, Control/Status Register Bridge Support Extensions, Control and Status (KT_HOST_PMD_PMCSRBSE_PMCSR)—Offset 54h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 22
Function: 3

Default: 8h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	**Data (Data):** Not implemented. Hardwired to 0.
23:16	0h RO	**Control/Status Register Bridge Support Extensions (CSRBSE):** Not implemented. Hardwired to 0.
15	0h RO	**PME Status (PMESTS):** This bit is set when the function would normally assert the PME signal independent of the state of the PME_En bit. Writing a 1 to this bit will clear it and cause the function to stop asserting a PME (if enabled). Writing a 0 has no effect. If the function supports PME from $D3_{COLD}$, then this bit is sticky and must be explicitly cleared by the operating system each time the operating system is initially loaded. Not implemented. Hardwired to 0.
14:13	0h RO	**Data Scale (DS):** Not implemented. Hardwired to 0.
12:9	0h RO	**Data Select (DSEL):** Not implemented. Hardwired to 0.
8	0h RO	**PME Enable (PMEEN):** A 1 enables the function to assert PME. When 0, PME assertion is disabled. This bit defaults to 0 if the function does not support PME generation from $D3_{COLD}$. If the function supports PME from $D3_{COLD}$ then this bit is sticky and must be explicitly cleared by the operating system each time it is initially loaded. Not implemented. Hardwired to 0.
7:4	0h RO	Reserved.
3	1h RO	**No Soft Reset (NSR):** When set to 1, this bit indicates that devices transitioning from $D3_{HOT}$ to D0 because of PowerState commands do not perform an internal reset. Configuration Context is preserved. Upon transition from the $D3_{HOT}$ to the D0 Initialized state, no additional operating system intervention is required to preserve Configuration Context beyond writing the PowerState bits.
2	0h RO	Reserved.
1:0	0h RW	**Power State (PWRST):** This 2-bit field is used both to determine the current power state of a function and to set the function into a new power state. The definition of the field values is given below: 00b - D0 01b - D1 10b - D2 11b - $D3_{HOT}$ If software attempts to write an unsupported, optional state to this field, the write operation must complete normally on the bus; however, the data is discarded and no state change occurs.

32.2 Keyboard and Text (KT) Additional Configuration Registers Summary

Table 32-2. Summary of Keyboard and Text (KT) Additional Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
54h	57h	Power Management Control and Status (KT_CSXE_PMD_PMCSRBSE_PMCSR)—Offset 54h	8h

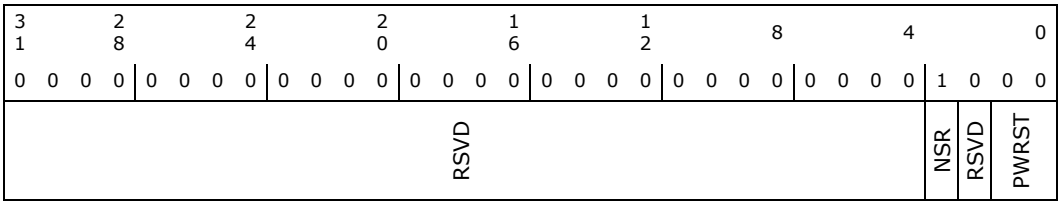
32.2.1 Power Management Control and Status (KT_CSXE_PMD_PMCSRBSE_PMCSR)—Offset 54h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 13
Function: 1

Default: 8h



Bit Range	Default and Access	Field Name (ID): Description
31:4	0h RO	Reserved.
3	1h RO	**No Soft Reset (NSR):** When set to 1, this bit indicates that devices transitioning from $D3_{HOT}$ to D0 because of PowerState commands do not perform an internal reset. Configuration Context is preserved. Upon transition from the $D3_{HOT}$ to the D0 Initialized state, no additional operating system intervention is required to preserve Configuration Context beyond writing the PowerState bits.
2	0h RO	Reserved.
1:0	0h RW	**Power State (PWRST):** This 2-bit field is used both to determine the current power state of a function and to set the function into a new power state. The definition of the field values is given below: 00b - D0 01b - D1 10b - D2 11b - $D3_{HOT}$ If software attempts to write an unsupported, optional state to this field, the write operation will complete normally on the bus; however, the data is discarded and no state change occurs.

§

33 Innovation Engine (IE)

33.1 Acronyms

Acronyms	Description
IE	Innovation Engine
Intel® ME	Intel® Management Engine
DRNG	Digital Random Number Generator
ITH	Intel Trace Hub

33.2 References

None.

33.3 Overview

The PCH integrates a small embedded core referred to as the Innovation Engine. The IE promotes customer innovation. The IE is a small, 32-bit Intel® Architecture core. It can be enabled by system builders and OEM's to run signed firmware created by the system builder or a third party software vendor. The IE can operate independently or cooperatively with the Intel® Management Engine (Intel® ME), which also resides in the PCH. This subsystem is connected to the rest of the system via an internal fabric, and has its own system address space, exposing some of its interfaces to the host.

33.4 Signal Description

Name	Type	Description
GPP_C3_SML0CLK_IE	I/OD	**IE System Management Link [0] Clock**; Muxed with Intel ME SML0.
GPP_C4_SML0DATA_IE	I/OD	**IE System Management Link [0] Data**; Muxed with Intel ME SML0.
GPP_C5_SML0ALERT_IE_N	I/OD	**IE System Management Link [0] Alert**; Muxed with Intel ME SML0.
GPP_D13_SML0BCLK_IE	I/OD	**IE 2nd instance of Intel ME's SML0CLK used for Comms Hub**; Muxed with Intel ME SML0B.
GPP_D14_SML0BDATA_IE	I/OD	**IE 2nd instance of Intel ME's SML0DATA used for Comms Hub**; Muxed with Intel ME SML0B.
GPP_D16_SML0BALERT_IE_N	I/OD	**IE 2nd instance of Intel ME's SML0ALERTB used for Comms Hub**; Muxed with Intel ME SML0B.
GPP_C6_SML1CLK_IE	I/OD	**IE System Management Link [1] Clock**; Muxed with Intel ME SML1.
GPP_C7_SML1DATA_IE	I/OD	**IE System Management Link [1] Data**; Muxed with Intel ME SML1.
GPP_B23_MEIE_SML1ALRT_N_PHOT_N	I/OD	**IE SML1ALERTB: System Management Link [1] Alert**; Muxed with Intel ME SML1.
GPP_H10_SML2CLK_IE	I/OD	**IE System Management Link [2] Clock**; Muxed with Intel ME SML2.
GPP_H11_SML2DATA_IE	I/OD	**IE System Management Link [2] Data**; Muxed with Intel ME SML2.
GPP_H12_SML2ALERT_N_IE_N	I/OD	**IE System Management Link [2] Alert**; Muxed with Intel ME SML2.
GPP_H13_SML3CLK_IE	I/OD	**IE System Management Link [3] Clock**; Muxed with Intel ME SML3.
GPP_H14_SML3DATA_IE	I/OD	**IE System Management Link [3] Data**; Muxed with Intel ME SML3.

Name	Type	Description
GPP_H15_SML3ALERT_N_IE_N	I/OD	**IE System Management Link [3] Alert**; Muxed with Intel ME SML3.
GPP_H16_SML4CLK_IE	I/OD	**IE System Management Link [4] Clock**; Muxed with Intel ME SML4.
GPP_H17_SML4DATA_IE	I/OD	**IE System Management Link [4] Data**; Muxed with Intel ME SML4.
GPP_H18_SML4ALERT_N_IE_N	I/OD	**IE System Management Link [4] Alert**; Muxed with Intel ME SML4.
GPP_G0_FANTACH0_FANTACH0IE	I	**IE Tachometer pulse input signal 0**; It is used to measure fan speed via the fan's SENSE signal.
GPP_G1_FANTACH1_FANTACH1IE	I	**IE Tachometer pulse input signal 1**; It is used to measure fan speed via the fan's SENSE signal.
GPP_G2_FANTACH2_FANTACH2IE	I	**IE Tachometer pulse input signal 2**; It is used to measure fan speed via the fan's SENSE signal.
GPP_G3_FANTACH3_FANTACH3IE	I	**IE Tachometer pulse input signal 3**; It is used to measure fan speed via the fan's SENSE signal.
GPP_G4_FANTACH4_FANTACH4IE	I	**IE Tachometer pulse input signal 4**; It is used to measure fan speed via the fan's SENSE signal.
GPP_G5_FANTACH5_FANTACH5IE	I	**IE Tachometer pulse input signal 5**; It is used to measure fan speed via the fan's SENSE signal.
GPP_G6_FANTACH6_FANTACH6IE	I	**IE Tachometer pulse input signal 6**; It is used to measure fan speed via the fan's SENSE signal.
GPP_G7_FANTACH7_FANTACH7IE	I	**IE Tachometer pulse input signal 7**; It is used to measure fan speed via the fan's SENSE signal.
GPP_G8_FANPWM0_FANPWM0IE	OD	**IE Pulse Width Modulated duty cycle output signal 0;** It is used for fan speed control.
GPP_G9_FANPWM1_FAMPWM1IE	OD	**IE Pulse Width Modulated duty cycle output signal 1;** It is used for fan speed control.
GPP_G10_FANPWM2_FANPWM2IE	OD	**IE Pulse Width Modulated duty cycle output signal 2;** It is used for fan speed control.
GPP_G11_FANPWM3_FANPMW3IE	OD	**IE Pulse Width Modulated duty cycle output signal 3;** It is used for fan speed control.
GPP_D22_IE_UART_TX	O	**IE High Speed UART Transmit Data**
GPP_D21_IE_UART_RX	I	**IE High Speed UART Receive Data**

33.5 Integrated Pull-Ups and Pull-Downs

Signal	Resistor Type	Value	Notes
GPP_C5_SML0ALERT_IE_N	Pull-Down		1
GPP_B23_MEIE_SML1ALRT_N_PHOT_N	Pull-Down		1
GPP_H12_SML2ALERT_N_IE_N	Pull-Down		1
GPP_G0_FANTACH0_FANTACH0IE	Pull-Up		
GPP_G1_FANTACH1_FANTACH1IE	Pull-Up		
GPP_G2_FANTACH2_FANTACH2IE	Pull-Up		
GPP_G3_FANTACH3_FANTACH3IE	Pull-Up		
GPP_G4_FANTACH4_FANTACH4IE	Pull-Up		
GPP_G5_FANTACH5_FANTACH5IE	Pull-Up		
GPP_G6_FANTACH6_FANTACH6IE	Pull-Up		
GPP_G7_FANTACH7_FANTACH7IE	Pull-Up		

Notes:
1. The internal pull-down is only applied during the strap sampling window and is then disabled.
2. Internal pull-ups and pull-downs are only valid once all the voltages are valid.

33.6 I/O Signal Planes and States

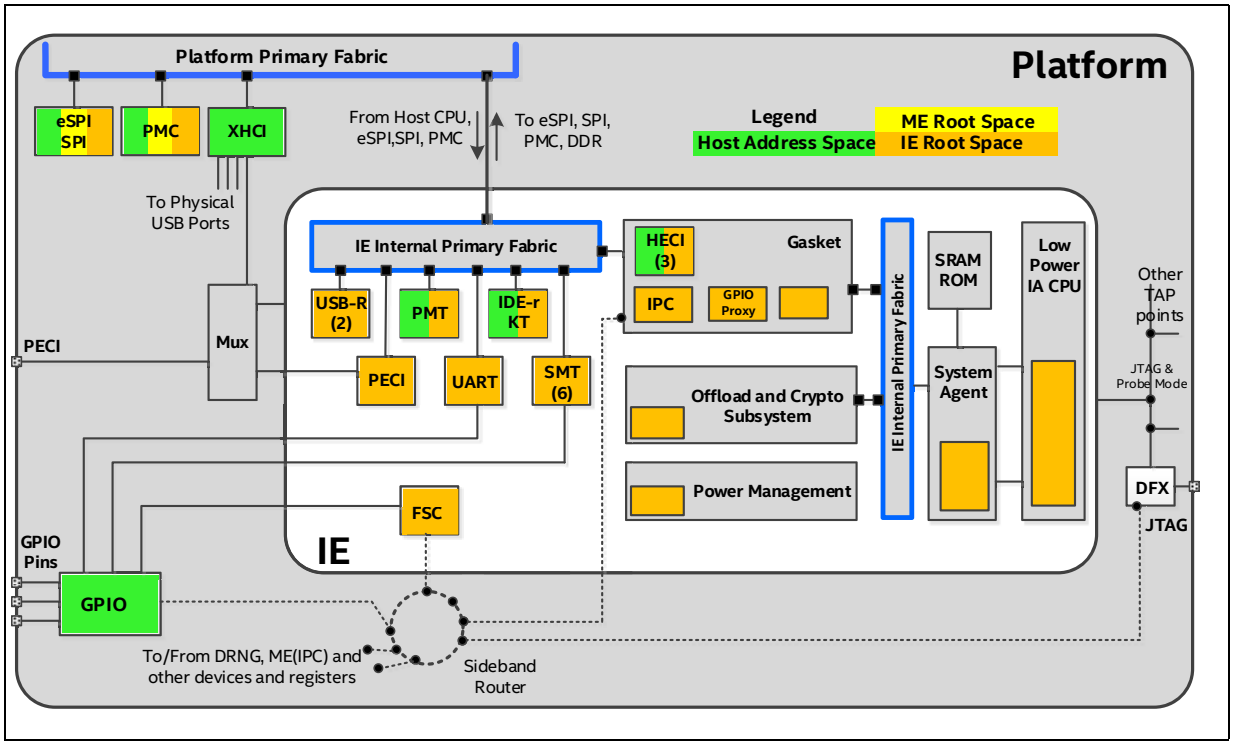
Signal Name	Power Plane	During Reset	Immediately After Reset	S3/S4/S5	Deep Sx
GPP_C3_SML0CLK_IE	Primary	Z	Z	Z	OFF
GPP_C4_SML0DATA_IE	Primary	Z	Z	Z	OFF
GPP_C5_SML0ALERT_IE_N	Primary	L	0	0	OFF
GPP_D13_SML0BCLK_IE	Primary	Z	Z	Z	OFF
GPP_D14_SML0BDATA_IE	Primary	Z	Z	Z	OFF
GPP_D16_SML0BALERT_IE_N	Primary	Z	Z	Z	OFF
GPP_C6_SML1CLK_IE	Primary	Z	Z	Z	OFF
GPP_C7_SML1DATA_IE	Primary	Z	Z	Z	OFF
GPP_B23_MEIE_SML1ALRT_N_PHOT_N	Primary	Z	Z	Z	OFF
GPP_H10_SML2CLK_IE	Primary	Z	Z	Z	OFF
GPP_H11_SML2DATA_IE	Primary	Z	Z	Z	OFF
GPP_H12_SML2ALERT_N_IE_N	Primary	Z	Z	Z	OFF
GPP_H13_SML3CLK_IE	Primary	Z	Z	Z	OFF
GPP_H14_SML3DATA_IE	Primary	Z	Z	Z	OFF
GPP_H15_SML3ALERT_N_IE_N	Primary	Z	Z	Z	OFF
GPP_H16_SML4CLK_IE	Primary	Z	Z	Z	OFF
GPP_H17_SML4DATA_IE	Primary	Z	Z	Z	OFF
GPP_H18_SML4ALERT_N_IE_N	Primary	Z	Z	Z	OFF
GPP_G0_FANTACH0_FANTACH0IE	Primary	Z	Z	Z	OFF
GPP_G1_FANTACH1_FANTACH1IE	Primary	Z	Z	Z	OFF
GPP_G2_FANTACH2_FANTACH2IE	Primary	Z	Z	Z	OFF
GPP_G3_FANTACH3_FANTACH3IE	Primary	Z	Z	Z	OFF
GPP_G4_FANTACH4_FANTACH4IE	Primary	Z	Z	Z	OFF
GPP_G5_FANTACH5_FANTACH5IE	Primary	Z	Z	Z	OFF
GPP_G6_FANTACH6_FANTACH6IE	Primary	Z	Z	Z	OFF
GPP_G7_FANTACH7_FANTACH7IE	Primary	Z	Z	Z	OFF
GPP_G8_FANPWM0_FANPWM0IE	Primary	Z	Z	Z	OFF
GPP_G9_FANPWM1_FAMPWM1IE	Primary	Z	Z	Z	OFF
GPP_G10_FANPWM2_FANPWM2IE	Primary	Z	Z	Z	OFF
GPP_G11_FANPWM3_FANPMW3IE	Primary	Z	Z	Z	OFF

33.7 Functional Description

The IE sub-system includes a very small, low power 32-bit IA core plus a system agent, ROM/SRAM and I/O. The IE is designed for Original Equipment Manufacturers and system builders to develop product differentiating features built on Intel Architecture.

33.7.1 Block Diagram

Figure 33-1. IE Block Diagram



IE has the following main blocks:

- Low Power IA CPU
- System Agent
- SRAM/ROM
- Offload and Crypto Subsystem
- Gasket
- Converged Manageability

IE is integrated to the PCH via internal primary and sideband fabric connections to PCH primary fabric and sideband router as shown in the figure above.

Other main interfaces (signals) include:

- IA CPU JTAG and Probe mode interface to PCH DFx
- SMT/FSC/UART interface to GPIO
- PECI interface to PECI pin
- UTMI interface to XHCI controller

33.7.2 IE Devices

Refer to Table 33-1 for IE host root space device list.

Table 33-1. IE Host Root Space Devices

Host Visible Functions B:D:F		
Device	**B:D:F**	**Comments**
HECI1	M:16:0	
HECI2	M:16:1	
PTIO:IDER	M:16:2	
PTIO:KT	M:16:3	
HECI3	M:16:4	
PMT	M:16:5	

In addition to host root space devices, IE implements several IE root space PCI functions which have PCI configure header and are enumerate-able to the IE FW. IE also implements several fixed address (a.k.a ACPI) IE root space devices and some PCI-like devices on sideband fabric. IE FW directly interfaces with these devices. IE root space device's Device/Function number and Device ID are listed in Table 33-2. HECI1, HECI2, HECI3, PTIO:IDER, PTIO:KT and PMT are dual root space devices.

Table 33-2. IE Root Space Devices (Sheet 1 of 2)

IE Root Space Device	Device Type	IE Root Space Device BDF	IE Root Space Device ID	Comment
Gasket				
HECI1*	ACPI	D2:F0	0xFFF0	
HECI2*	ACPI	D1:F1	0xFFF1	
HECI3*	ACPI	D1:F11	0xFFFB	
GPIO Proxy	ACPI	D1:F7	0xFFF7	
CM Devices				
USB-r1	PCI	D9:F0	0x0110	
USB-r2	PCI	D9-F1	0x0111	
SMT1	PCI	D10:F0	0x0120	
SMT2	PCI	D10:F1	0x0121	
SMT3	PCI	D10:F2	0x0122	
SMT4	PCI	D10:F3	0x0123	
SMT5	PCI	D10:F4	0x0124	
SMT6	PCI	D10:F5	0x0125	
PECI	PCI	D12:F0	0x0104	
PTIO:IDER*	PCI	D13:F0	0x0105	
PTIO:KT*	PCI	D13:F1	0x0106	
PMT*	PCI	D14:F0	0x0107	
UART	PCI	D15:F0	0x0108	
Outside IE				
SPI	PCI	D25:F0	0x0204	Intel ME Root Space SPI flash has a different DID of 0x0201.

Table 33-2. IE Root Space Devices (Sheet 2 of 2)

IE Root Space Device	Device Type	IE Root Space Device BDF	IE Root Space Device ID	Comment
eSPI	PCI	D25:F1	0x0202	
PMC	PCI	D26:F0	0x0203	
DRNG	Sideband Fabric	NA	NA	
ITH	Sideband Fabric	NA	NA	
Note: * These are dual root-space devices.				

33.7.3 IE Features

The IE features are listed in Table 33-3.

Table 33-3. IE Features

Feature	Comment
Access to System Memory	Yes
HECI	Yes (x3)
GPIO Proxy	Yes
Application / Kernel Timer	Yes
IPC	Intel ME
USB-r	Yes
SMT	Yes (x6)
PECI	Yes
IDE-r	Yes
KT	Yes
PMT	Yes
UART	Yes
Hotham	Yes
FSC	Yes
DMA-AES (AES-128, AES-256)	2
DMA-Hash (SHA-1, SHA256)	1
DMA-RC4	1
GPDMA	1
Firmware Tracing via ITH	Yes
JTAG interface	Yes
Chassis compliant power gating	Yes
Sx Support	Yes
Clock gating	Yes
Dynamic clock throttling	Yes
Partition Reset	Yes
The number of Internal primary fabric channels	3
Sideband fabric	Yes

Refer to IE development guide for detail operation of each feature, and device programming model.

33.7.4 IE Power Management

This section summarizes the IE Power Management functionality in the PCH. From the PCI configuration (IE root space) perspective, this chapter contains function 0 representing the interface for the Power Management Controller and some other IE Power Management registers.

Summary of differences between IE and Intel ME PM space.

- IE does not have access to RTC configure registers control.
- IE can not control the sleep pins for sleep stretching.
- IE does not have access to the power state control register.
- IE does not have control of the configuration of timeouts for the PCH
  - PROC_PWRGD to CPURST time out value
  - PROC_PWRGD stuck low timeout
  - Host_Prim_Rst entry time
- Does not have access to the integrated 1 GbE LAN control registers. The definitions still exist in the register bits but the fields have been set to reserve.

33.7.4.1 PROC_PWRGD Support of IE

The IE is capable of delaying the wake Host platform wake sequence. In order to avoid violating the PEROC_PWRGD-to-RST# specification, this is done by delaying the rising edge of the PROC_PWRGD pin. Additionally, the PROC_PWRGD pin must be driven low in all of the cases in which the traditional SLP_S3# pin was driven low. This prevents the PCH from violating the CPU's max PROC_PWRGD-to-CPURST# de-assertion specification in the event that the IE overrides the SLP_S3# pin high. For example, in the boundary case that a reset-with power-cycle event occurs immediately after reset-without-power-cycle is started. In that case, the PCH may proceed with a long reset sequence in which the reset logic attempts to power-cycle the platform. However, the IE is overriding the SLP_Sx# signals high. Therefore, the processor would not see its PROC_PWRGD pin de-assert while its reset input is asserted for seconds without this new functional change. Adding SLP_S3# to the conditions that de-assert PROC_PWRGD is also important for the case of a short residency in an S3/4/5 state. If the platform power discharges slowly, it is possible that a platform could keep PCH_PWROK and SYS_PWROK asserted for extended periods of time thereby either violating the PROC_PWRGD-to-RST# specification or preventing the IE from delaying the wake from S3.

The PCH contains Hardware Watchdog timers to protect against Firmware hangs or other failures that can cause the PROC_PWRGD-related specs to be violated at the processor.

33.7.4.2 IE Reset Handling

The IE platform is capable of operating through a Intel ME and a host partitioned reset and vice versa. There are also cases where either platform can cause a “Global Reset” condition that forces a reset on all platforms.

33.8 Registers

Table 33-1 lists host visible IE devices. Host software accesses these IE devices register through their bus, device, and function number in the host space. For the register detail, refer to the IE developers guide included in Section 33.2.

33.8.1 IE HECI Devices In Host Space (D16:F0, D16:F1, D16:F4)

There are three host visible IE HECI devices for host software access to communicate with IE firmware. The device ID of IE HECI devices are listed in Chapter 2.

Refer to the Intel ME HECI Interface Chapter for the three IE HECI devices register description in host space. They are identical to Intel ME HECI devices except the device number is 16.

§

34 Fan Control

34.1 Acronyms

Acronyms	Description
PWM	Pulse Width Modulation
FSC	Fan Speed Controller

34.2 References

None.

34.3 Overview

The Fan Speed Controller is used to control the fans in the system.

34.4 Signal Description

Name	Type	Description
FAN_TACH_0 / GPP_G0	I	Tachometer pulse input signal
FAN_TACH_1 / GPP_G1	I	Tachometer pulse input signal
FAN_TACH_2 / GPP_G2	I	Tachometer pulse input signal
FAN_TACH_3 / GPP_G3	I	Tachometer pulse input signal
FAN_TACH_4 / GPP_G4	I	Tachometer pulse input signal
FAN_TACH_5 / GPP_G5	I	Tachometer pulse input signal
FAN_TACH_6 / GPP_G6	I	Tachometer pulse input signal
FAN_TACH_7 / GPP_G7	I	Tachometer pulse input signal
FAN_PWM_0 / GPP_G8	OD	Pulse Width Modulated duty cycle output
FAN_PWM_1 / GPP_G9	OD	Pulse Width Modulated duty cycle output
FAN_PWM_2 / GPP_G10	OD	Pulse Width Modulated duty cycle output
FAN_PWM_3 / GPP_G11	OD	Pulse Width Modulated duty cycle output

34.5 Integrated Pull-ups and Pull-downs

Signal	Resistor Type	Value	Notes
FAN_TACH_[7:0]	Internal Pull-up	14K - 26K	
Note: 1. The internal pull-down is only applied during the strap sampling window and is then disabled.			

34.6 I/O Signal Planes and States

Signal Name	Power Plane	During Reset	Immediately after Reset	S3/S4/S5	Deep Sx
FAN_TACH_[7:0]	Primary	Undriven	Undriven	Undriven	Off
FAN_PWM_[3:0]	Primary	Undriven	Undriven	Undriven	Off

34.7 Functional Description

The Fan Speed Controller (FSC) is used to control fans in the system. FSC drives four pulse width modulated signals to control the speed of the fans. FSC samples eight tachometer inputs to measure the speed of the fans. FSC also has an interface to the on-die thermal sensor to know the current temperature.

PWM signal is driven as open-drain. An external pull-up resistor should be implemented to provide the rising edge of the PWM output signal. The PWM output is driven low during reset, which represents 0% duty cycle to the majority of fans.

Note that if a PWM output is programmed to inverted polarity for a particular fan, then the low voltage driven during reset represents 100% duty cycle to the fan.

§

35 Enterprise Value Add

35.1 Acronyms

Acronyms	Description
EVA	Enterprise Value Add
MS Unit	Management Subsystem Unit
IE	Innovation Engine
Intel® ME	Intel® Management Engine

35.2 References

None

35.3 Overview

The Intel® C620 Series Chipset adds new features above the standard PCH functions that we share with client for the purpose of providing extra support for the workstation and enterprise customers. Higher level enabling/control of these new features are done via the MS Unit. This section describes what encompasses the MS Unit. These functions are part of the base feature set of the Intel® C620 Series Chipset and are available with every SKU. The EVA functions are all accessible via DMI.

35.4 Functional Description

The MS Unit is a Multifunction device (device 17), consisting of MROM0 (Function 0), MROM 1(Function 1) and SSATA (Function 5).

The SSATA controller is a replication of the SATA controller. All the features and functions are the same with the exception that the SSATA controller has six ports while the SATA controller has eight. The register PCIe and Register description of the SSATA controller is in the SATA/SSATA chapter since they are the same.

MROM1 provides the configuration control of the IE via PCIe configuration registers. It provides the standard DID, VID< PCISTS, PCICMD, CC, RID, CLK, MLT and HDR registers from 00h to 0Fh The Expansion ROM base address is defined at address 30h. The maximum size of the ROM bar is 16 MB.

MROM0 is the base function for the MS Unit at function 0. The PCI configuration registers describe the expansion ROM region for the Intel ME. MROM0 also provides control and enables various EVA functions. The MS Unit Device Function Hide Control register provides the ability to hide SSATA and MROM1 from BIOS and the OS. MROM0 is never hidden. MROM0 provides the control for the ADR Timer. Through the ADR Timer Control register, the ADR timer can be enabled or disabled, and the value of the timer is programmed. The length of the time is combination of the ADR timer value and multiplier value. The ADR timer is programmed in steps of 25, 50 or 100 us, and then a multiplier of 1-96 is then applied to the timer value to give the final timeout.

Inside the logic of the MS Unit is the mux control of the PECI and USBr signals between IE and Intel ME. Both the Intel ME and IE have PECI control functions and interfaces to the XHCI host controller for USB redirection. The MS unit does the physical muxing of those signals. As it is muxing, it's not possible for the IE and Intel ME to both be able to control PECI and the USB redirection.

35.5 MROM0 Configuration Registers Summary

Table 35-1. Summary of xHCI Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	1h	Vendor ID (VID)—Offset 0h	8086h
2h	3h	Device ID (DID)—Offset 2h	(TBA)h
4h	5h	Command (CMD)—Offset 4h	0h
6h	7h	Device Status (STS)—Offset 6h	000h
8h	8h	Revision ID (RID)—Offset 8h	0h
9h	9h	Programming Interface (PI)—Offset 9h	30h
Ah	Ah	Sub Class Code (SCC)—Offset Ah	3h
Bh	Bh	Base Class Code (BCC)—Offset Bh	Ch
Dh	Dh	Master Latency Timer (MLT)—Offset Dh	0h
Eh	Eh	Header Type (HT)—Offset Eh	0h
2Ch	2Dh	Subsystem Vendor ID (SSVID)—Offset 2Ch	0h
2Eh	2Fh	Subsystem ID (SSID)—Offset 2Eh	0h
30h	33h	Expansion ROM Base Address Register (EROMBAR)—Offset 30h	70h
34	34	Capabilities Pointer (CAP_PTR) - Offset 30h	80h
3Ch	3Ch	Interrupt Line (ILINE)—Offset 3Ch	0h
3Dh	3Dh	Interrupt Pin (IPIN)—Offset 3Dh	0h
80h	81h	Power Management Capability List Register ID (PMCAPLST) - Offset 80h	01h
82h	83h	Power Management Capabilities (PM_CAP) - offset 82h	C0E2h
84h	85h	Power Management Control/Status (PM_CS) Offset 84h	0h
D4h	D7h	Device Function Hide Control Register (MSDEVFUNCHIDE) - Offset D4h	0h
DCh	DFh	Soft Strap Public Register (SSPUB) - Offset DCh	0h
E0h	E3h	Private Block Control Register (PRIBLKCTL) - Offset E0h	0h
E4	E7	Device Clock Gate Control Register (DEVCLKGCTL) - offset E4h	0h
E8	E9	Personality Lock Key Control Register (PLKCTL) - Offset E8h	0h
180	183	ADR Timer Control Register (ADRTIMERCTRL) - Offset 180h	10000000h

35.5.1 Vendor ID (VID)—Offset 0h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 0

Default: 8086h

15			12				8				4				0
1	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0
VID															

Bit Range	Default and Access	Field Name (ID): Description
15:0	8086h RO	**Vendor ID (VID)**

35.5.2 Device ID (DID)—Offset 2h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 20
Function: 0

Default: 8C31h

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DID															

Bit Range	Default and Access	Field Name (ID): Description
15:0	TBAV	**Device ID (DID):** See Global Device ID table in Chap. 2 for value

35.5.3 Command (CMD)—Offset 4h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 0

Default: 0h

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD					ID	FBE	SERR	WCC	PER	VPS	MWI	SCE	BME	MSE	IOSE

Bit Range	Default and Access	Field Name (ID): Description
15:11	0h RO	Reserved.
10	0h RO	**Interrupt Disable (ID):** MROM does not have an interrupt to disable, so this bit is hardwired to 0.
9	0h RO	**Fast Back to Back Enable (FBE)**
8	0h RW	**SERR# Enable (SERR):** When set to 1, the XHC is capable of generating (internally) SERR#. See section on error handling.
7	0h RO	**Wait Cycle Control (WCC)**
6	0h RW	**Parity Error Response (PER):** This bit controls the setting of the master data parity error bit in the status register in response to a parity error received.
5	0h RO	**VGA Palette Snoop (VPS)**
4	0h RO	**Memory Write Invalidate (MWI)**
3	0h RO	**Special Cycle Enable (SCE)**
2	0h RW	**Bus Master Enable (BME):** This bit controls the ability of the function to issue memory and I/O read or write requests.
1	0h RW	**Memory Space Enable (MSE):** This bit controls the response to memory space accesses. When 0b, the function will handle memory transactions targeting it as an Unsupported request (UR). For Type 1 configuration space headers, this bit controls the primary side response to memory space accesses targeting the secondary side. When this bis is 0b, every memory transaction targeting a secondary interface is handled as an Unsupported Request (UR).
0	0h RO	**I/O Space Enable (IOSE):** Reserved as 0. Read-Only.

35.5.4 Device Status (STS)—Offset 6h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 0

Default: 290h

15			12				8				4				0
0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0
DPE	SSE	RMA	RTA	STA	DEVT		MDPED	FBBC	UDF	MC	CL	IS	RSVD		

Bit Range	Default and Access	Field Name (ID): Description
15	0h RW/1C	**Detected Parity Error (DPE):** This bit is set by the Intel PCH whenever a parity error is seen on the internal interface, regardless of the setting of bit 6 or bit 8 in the Command register or any other conditions. Software clears this bit by writing a 1 to this bit location.
14	0h RW/1C	**Signaled System Error (SSE):** This bit is set by the Intel PCH whenever it signals SERR# (internally). The SERR_EN bit (bit 8 in the Command Register) must be 1 for this bit to be set. See error handling section for complete list of conditions handled. Software clears this bit by writing a 1 to this bit location.
13	0h RW/1C	**Received Master-Abort Status (RMA):** This bit is set when the requester receives a completion with a UR completion status. Software clears this bit by writing a 1 to this bit location.
12	0h RW/1C	**Received Target Abort Status (RTA):** This bit is set upon receiving a CA completion status. Software clears this bit by writing a 1 to this bit location.
11	0h RW/1C	**Signaled Target-Abort Status (STA):** This bit is used to indicate when the function generates a completion packet with Completer Abort (CA) status.
10:9	1h RO	**DEVSEL# Timing Status (DEVT):** This 2-bit field defines the timing for DEVSEL# assertion. Read-Only.
8	0h RW/1C	**Master Data Parity Error Detected (MDPED):** This bit is set by the Intel PCH whenever a data parity error is detected in a read completion packet on the internal interface and bit 6 of the Command register is set to 1. Software clears this bit by writing a 1 to this bit location.
7	0h RO	**Fast Back-to-Back Capable (FBBC):** Reserved as 0 Read-Only.
6	0h RO	**User Definable Features (UDF):** Reserved as 0. Read-Only.
5	0h RO	**66 MHz Capable (MC):** Reserved as 0. Read-Only.
4	1h RO	**Capabilities List (CL):** Hardwired to 1 indicating that offset 34h contains a valid capabilities pointer.
3	0h RO/V	**Interrupt Status (IS):** Interrupts are not supported so this bit always returns "0" when read.
2:0	0h RO	Reserved.

35.5.5 Revision ID (RID)—Offset 8h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 0

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
RID							

Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO/V	**Revision ID (RID):** RIDs are identified in the datasheet specification updates.

35.5.6 Programming Interface (PI)—Offset 9h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 0

Default: 30h

7			4				0
0	0	1	1	0	0	0	0
PI							

Bit Range	Default and Access	Field Name (ID): Description
7:0	00h RO	**Programming Interface (PI):** Default value of 00h.

35.5.7 Sub Class Code (SCC)—Offset Ah

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 0

Default: 3h

7			4				0
0	0	0	0	0	0	0	0
SCC							

Bit Range	Default and Access	Field Name (ID): Description
7:0	00h RO	**Sub Class Code (SCC):** default value of 00h

35.5.8 Base Class Code (BCC)—Offset Bh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 0

Default: Ch

7			4				0
0	0	0	0	1	1	0	0
BCC							

Bit Range	Default and Access	Field Name (ID): Description
7:0	FFh RO	**Base Class Code (BCC):** A value of FFh indicates that this does not fit into any defined class.

35.5.9 Cachleline Size Register (CLS)—Offset Ch

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 0

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
CLS							

Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW	**Cacheline Size Register (CLR):** This register has no meaning for MROM.

35.5.10 Master Latency Timer (MLT)—Offset Dh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 0

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
MLT							

Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO	**Master Latency Timer (MLT):** The bits will be fixed at 0 as this register has no meaning for MROM.

35.5.11 Header Type (HT)—Offset Eh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 0

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
MFB	CL						

Bit Range	Default and Access	Field Name (ID): Description
7	0h RO	**Multi-Function Bit (MFB):** Defaults to 0, but if after reset MROM1 or SSATA is enabled then this bit will become a “1”.
6:0	0h RO	**Configuration layout (CL):** Hardwired to 0 to indicate a standard PCI configuration layout.

Bit Range	Default and Access	Field Name (ID): Description
63:16	0h RW	**Base Address (BA):** Bits (63:16) correspond to memory address signals (63:16), respectively. This gives 64 KB of relocatable memory space aligned to 64 KB boundaries.
15:4	0h RO	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
3	0h RO	**Prefetchable (Prefetchable):** This bit is hardwired to 0 indicating that this range should not be prefetched.
2:1	2h RO	**Type (Type):** If this field is hardwired to 00 it indicates that this range can be mapped anywhere within 32-bit address space. If this field is hardwired to 10 it indicates that this range can be mapped anywhere within 64-bit address space.
0	0h RO	**Resource Type Indicator (RTE):** This bit is hardwired to 0 indicating that the base address field in this register maps to memory space

35.5.12 Subsystem Vendor ID (SSVID)—Offset 2Ch

This register is modified and maintained by BIOS.

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 0

Default: 8086h

15			12				8				4				0
1	0	0	0	1	0	0	0	1	0	0	0	0	1	1	0
SSVID															

Bit Range	Default and Access	Field Name (ID): Description
15:0	8086h RW/L	**Subsystem Vendor ID (SSVID):** This field identifies Intel as the manufacturer of the device.

35.5.13 Subsystem ID (SSID)—Offset 2Eh

This register is modified and maintained by BIOS

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 0

Default: 0h

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SSID															

Bit Range	Default and Access	Field Name (ID): Description
15:0	0h RW/L	**Subsystem ID (SSID):**

35.5.14 Expansion ROM Base Address Register (EROMBAR)—Offset 30h

Access Method

Type: CFG Register (Size: 32bits)

Device: 17
Function: 0

Default: XXh

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	1
RBAUB								RBAB23	RBAB22	RBAB21	RBAB20	RBAB19	RBAB18	RBAB17	RBAB16	RBAB15	RSVD														ERE

Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RW-LV	**ROM Base Address Upper Bits (RBAUB):** The ROM base address includes both RBAUB and RBAB (23:15) bits below. The RBAUB bits correspond to bits 31:24 0f the ROM base address value.
23	0h RW-LV	**ROM Base Address Bit 23(RBAB23):** Bit 23 becomes RO and locked with ROM is 16M, size = 0h. All MSIZE bits are RW if MROMCTL[size]>=9h, indicating 32K, unless otherwise locked as described.
22	0h RW-LV	**ROM Base Address Bit 22(RBAB22):** Bit 22 becomes RO and locked with ROM is 8M or larger, size <= 1h. All MSIZE bits are RW if MROMCTL[size]>=9h, indicating 32K, unless otherwise locked as described
21	0h RW-LV	**ROM Base Address Bit 21(RBAB21):** Bit 21 becomes RO and locked with ROM is 4M or larger, size <= 2h. All MSIZE bits are RW if MROMCTL[size]>=9h, indicating 32K, unless otherwise locked as described
20	0h RW-LV	**ROM Base Address Bit 20(RBAB20):** Bit 20 becomes RO and locked with ROM is 2M or larger, size <= 3h. All MSIZE bits are RW if MROMCTL[size]>=9h, indicating 32K, unless otherwise locked as described
19	0h RW-LV	**ROM Base Address Bit 19(RBAB19):** Bit 19 becomes RO and locked with ROM is 1M, size <= 4h. All MSIZE bits are RW if MROMCTL[size]>=9h, indicating 32K, unless otherwise locked as described
18	0h RW-LV	**ROM Base Address Bit 18(RBAB18):** Bit 18 becomes RO and locked with ROM is 512K or larger, size< = 5h. All MSIZE bits are RW if MROMCTL[size]>=9h, indicating 32K, unless otherwise locked as described
17	0h RW-LV	**ROM Base Address Bit 17(RBAB17):** Bit 17 becomes RO and locked with ROM is 256K or larger, size <= 6h. All MSIZE bits are RW if MROMCTL[size]>=9h, indicating 32K, unless otherwise locked as described
16	0h RW-LV	**ROM Base Address Bit 16(RBAB16):** Bit 16 becomes RO and locked with ROM is 128K or larger, size <= 7h. All MSIZE bits are RW if MROMCTL[size]>=9h, indicating 32K, unless otherwise locked as described

Bit Range	Default and Access	Field Name (ID): Description
15	0h RW-LV	**ROM Base Address Bit 15(RBAB15):** Bit 15 becomes RO and locked with ROM is 64K or larger, size <= 8h. All MSIZE bits are RW if MROMCTL[size]>=9h, indicating 32K, unless otherwise locked as described
14:1	0;RO	Reserved
0	0h RW-LV	**Expansion ROM Enable (ERE):** 0b: ROM Disabled 1b: Expansion ROM enabled Not writable if MROMCTL[EMROM] = 0 or PRIBLKCTL{ERBD] =1 as described above. If ERE = 1b, but PCICMD[MSE] = 0b, then this address region will no be enabled, as PCICMD[MSE] takes precedence. Both bits need to be 1 in order to enabled address decoding. BIOS must disable ROMBAR to prevent further MROM access prior to the transition to OS. This register can be locked down by PRIBLKCTL[ERBD]=1

35.5.15 Capabilities Pointer (CAP_PTR)—Offset 34h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 0

Default: 80h

7			4				0
1	0	0	0	0	0	0	0
CAP_PTR							

Bit Range	Default and Access	Field Name (ID): Description
7:0	80h RO	**Capabilities Pointer (CAP_PTR):** This register points to the starting offset of the capabilities ranges.

35.5.16 Interrupt Line (ILINE)—Offset 3Ch

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 0

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
ILINE							

Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW	**Interrupt Line (ILINE):** This data is not used by the Intel PCH. It is used as a scratchpad register to communicate to software the interrupt line that the interrupt pin is connected to. A value of 0xFF defines no connection. BIOS should write FFh to this field

35.5.17 Interrupt Pin (IPIN)—Offset 3Dh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 20
Function: 0

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
IPIN							

Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW/L	**Interrupt pin (IPIN):** Bits 7:0 reflect the Interrupt Pin assigned to the host controller by the platform (and are hardwired).

35.5.18 Power Management Capability List Register ID (PMCAPLST)—Offset 80h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 0

Default: 1h

15			12				8				4				0
0	0	0	0	0	0	0	0	0I	0	0	0	0	0	0	1
NP								CPID							

Bit Range	Default and Access	Field Name (ID): Description
7:0	1h RO	**PCI Power Management Capability ID (PM_CID):** A value of 01h indicates that this is a PCI Power Management capabilities field.
15:8	00h RW-L	**Next Pointer (NP):** Contains the offset of the next item in the capabilities list. A null value is used to indicate that this is the last capability.

35.5.19 Power Management Capabilities (PM_CAP)—Offset 82h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 0

Default: 0000h

15			12				8				4				0
1	1	0	0	0	0	0	1	1	1	0	0	0	0	1	0
PME_Support					D2_Support	D1_Support	Aux_Current			DSI	RSVD	PMEClock	Version		

Bit Range	Default and Access	Field Name (ID): Description
15:11	00h RO	**PME_Support (PME_Support):** This 5-bit field indicates the power states in which the function may assert PME#. A value of 00h implies that this device does not generate PME events.
10	0h RO	**D2_Support (D2_Support):** The D2 state is not supported.
9	0h RO	**D1_Support (D1_Support):** The D1 state is not supported.
8:6	0h RO	**Aux_Current (Aux_Current):** Auxiliary power is not supported.
5	0h Ro	**DSI (DSI):** The Intel PCH reports 0, indicating that no device-specific initialization is required.
4	0h RO	Reserved.
3	0h RW/L	**PME Clock (PMEClock):** The Intel PCH reports 0, indicating that no PCI clock is required to generate PME#.
2:0	3h RW/L	**Version (Version):** Complies with Revision 1.2 of the PCI Power Management Specification.

35.5.20 Power Management Control/Status (PM_CS)—Offset 84h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 0

Default: 8h

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
PME_Status	Data_Scale		Data_Select				PME_En	RSVD				NSR	RSVD	PowerState	

Bit Range	Default and Access	Field Name (ID): Description
15	0h RW/1C	**PME_Status (PME_Status):** This bit is set when the function would normally assert the PME# signal independent of the state of the PME_En bit. Writing a 1 to this bit will clear it and cause the internal PME to deassert (if enabled). Writing a 0 has no effect. This bit must be explicitly cleared by the operating system each time the operating system is loaded.
14:13	0h RO	**Data_Scale (Data_Scale):** Not supported so hardwired to 00.
12:9	0h RO	**Data_Select (Data_Select):** Not supported so hardwired to 00.
8	0h RW	**PME_En (PME_En):** A 1 enables the function to generate an internal PME signal when PME_Status is 1. This bit must be explicitly cleared by the operating system each time it is initially loaded.
7:4	0h RO	Reserved.
3	1h RO	**No Soft Reset (NSR):** This bit indicates that devices transitioning from $D3_{HOT}$ to D0 because of PowerState commands do not perform an internal reset. Configuration Context is preserved. Upon transition from the $D3_{HOT}$ to the D0 Initialized state, no additional operating system intervention is required to preserve Configuration Context beyond writing the PowerState bits.
2	0h RO	Reserved.
1:0	0h RW	**PowerState (PowerState):** This 2-bit field is used both to determine the current power state and to set a new power state. The definition of the field values are: 00b - D0 state 11b - $D3_{HOT}$ state If software attempts to write a value of 10b or 01b in to this field, the write operation must complete normally, however, the data is discarded and no state change occurs.

35.5.21 Device Function Hide Control Register (MSDEVFUNCHIDE)—Offset D4h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 17
Function: 0

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
LOCK	RSVD																								SSATA	RSVD			MROM1	RSVD	

Bit Range	Default and Access	Field Name (ID): Description
31	0h RW/l	**LOCK (Addr):** locked =1b, unlocked = 0b. This bit locks the device function hide register. Once this bit is set to 1b, it can not be unlocked. Writing a 0b has no effect. This gets reset to 0b on a power cycle.
30:6	0h RO	Reserved.
5	0b RW/L	**Function 5 sSATA (SSATA):** 0b = enabled for use by BIOS/OS, 1b = hidden from BIOS/OS.
4:2	0h RO	Reserved.
1	1b RW/L	**Function 1 MROM1(MROM1**) 0b = enabled for use by BIOS/OS, 1b = hidden from BIOS/OS. By default MROM1 is hidden from BIOS and the OS
0	0 RO	**Function 0 MROM0 (MROM0):** MROM0 is never hidden

35.5.22 Soft Strap Public Register (SSPUB)—Offset DCh

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 17
Function: 0

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																														DISSSATA	DISMROM1

Bit Range	Default and Access	Field Name (ID): Description
31:2	0h RO	**RSVD**
1	0h RO	Disable SSATA (DISSSATA): reflects the status of soft straps to enable or disable the SSATA. 0 = SSATA enabled. 1 = SSATA disabled
0	0b RO	Disab.e MROM1 (DISMROM1): reflects the status of soft straps to enable or disable the MROM1. 0 = MROM1enabled. 1 = MROM1disabled

35.5.23 Private Block Control Register (PRIBLKCTL)—Offset E0h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 17
Function: 0

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ERBD	RSVD																														DNEERB

Bit Range	Default and Access	Field Name (ID): Description
31	0h RO	**Expansion ROM BAR Disable (ERBD):** This bit is self locking. Once set to one can not be cleared by writing a "0" to it. Setting this to "1" makes all bits in the EROMBAR register RO
30:1	0h RW	These bits are reserved and must remain "0"
0	0b RO	**Do Not Enable Expansion ROM Bar (DNEERB);** This is a read-only bit that reflects the state of a fuse bit. The fuse does not directly control anything but is used as a signal for BIOS and FW to enable the expansion ROM BAR or not.

35.5.24 Device Clock Gate Control Register (DEVCLKGCTL)—Offset E4h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 0

Default: 0010h

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ICGE	RSVD							ICT							

Bit Range	Default and Access	Field Name (ID): Description
15	0h Rw	**Idle Clock Gate Enable (ICGE):** When set this bit enables the IP block to enter clock gating mode, where the clocks are turned off to save power.
14:8	0h RW	These bits are reserved and must remain "0".
7:0	0b Rw	**Idle Clock Timer (ICT):** This field indicates the number of clocks that the IP block must be idle for before the clock disable process begins. 01h = 1 clock and FFh = 255. The minimum number recommended is 16 (default).

35.5.25 Personality Lock Key Control Register (PLKCTL) —Offset E8h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 0

Default: 0000h

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD															CL

Bit Range	Default and Access	Field Name (ID): Description
15:1	0h RW	These bits are reserved and must remain "0".
0	0b Rw-L	**Capability Lock (CL):** Lock bit for all RW-L bits in all registers for this function. 0b= unlocked 1b = locked. Once a "1" is written to this bit it is only reset to 0 with a global reset.

35.5.26 ADR Timer Control Register (ADRTIMERCTRL)—Offset 180h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 17
Function: 0

Default: 10000000h

31			28				24				20				16				12				8				4				0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ADR_TMR			RSVD				ADR_MUL				RSVD																					

Bit Range	Default and Access	Field Name (ID): Description
31:30	00b RW-L	**ADR Timer (ADR_TMR)**: The value programmed into this field is part of the equation for determining how long a global reset included in the ADR flow will be delayed before triggering a reset. The time must be long enough to along the even to be communicated to the CPU via PM Sync and for the CPU to complete it's required steps. The actual time is a combination of the time determined by this register multiplied by the value in the ADR timer multiplier (bits 27:24 of this same register). The lock key bit is located in the Personality Lock Key Control Register (PLKCTL) 00: 25 uS 01: 50 uS 10: 100 uS 11: 0 uS (off)
29:28	01b RSVD	**RSVD**
27:24	0h, RW-L	**ADR Timer Multiplier (ADR_MULT):** This field contains 9 encoded values, that when used as multiplies with ADR_TMR produce the final length of time that a global reset is held off for in the ADR flow. 0000: 1 0001: 8 0010: 24 0011: 40 0100: 56 0110: 72 0111: 80 1000: 88 1001: 96 1010 - 1111: Reserved, defaults to a multiplier of 96
23:0	0h RV	**RSVD**

35.6 MROM1 Configuration Registers Summary

Table 35-2. Summary of MROM1 Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	1h	Vendor ID (VID)—Offset 0h	8086h
2h	3h	Device ID (DID)—Offset 2h	(TBA)h
4h	5h	Command (CMD)—Offset 4h	0h
6h	7h	Device Status (STS)—Offset 6h	000h
8h	8h	Revision ID (RID)—Offset 8h	0h
9h	9h	Programming Interface (PI)—Offset 9h	30h
Ah	Ah	Sub Class Code (SCC)—Offset Ah	3h
Bh	Bh	Base Class Code (BCC)—Offset Bh	Ch
Dh	Dh	Master Latency Timer (MLT)—Offset Dh	0h
Eh	Eh	Header Type (HT)—Offset Eh	0h
2Ch	2Dh	Subsystem Vendor ID (SSVID)—Offset 2Ch	0h
2Eh	2Fh	Subsystem ID (SSID)—Offset 2Eh	0h
30h	33	Expansion ROM Base Address Register (EROMBAR)—Offset 30h	0h
34h	34h	Capabilities Pointer (CAP_PTR) Offset 34h	80h

Table 35-2. Summary of MROM1 Configuration Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
3Ch	3Ch	Interrupt Line (ILINE)—Offset 3Ch	0h
3Dh	3Dh	Interrupt Pin (IPIN)—Offset 3Dh	0h
80h	81h	Power management Capability List Register ID (PMCAPLST) - Offset 80h	0h
82	83h	Power Management Capabilities (PM_CAP) - Offset 82h	0h
84h	87h	Power Management Control/Status (PM_CS) Offset 84h	0008h
E0	E3	Private Block Control Register (PRIBLKCTL) - Offset E0h	0h
E4	E5	Device Clock Gate Control Register (DEVCLKGCTL) - Offset E4h	0h
E8h	E9h	Personality Lock Key control Register (PLKCTL) - Offset E8h	0h

35.6.1 Vendor ID (VID)—Offset 0h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 1

Default: 8086h

15			12				8				4				0
1	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0
VID															

Bit Range	Default and Access	Field Name (ID): Description
15:0	8086h RO	**Vendor ID (VID)**

35.6.2 Device ID (DID)—Offset 2h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 1

Default: h

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DID															

Bit Range	Default and Access	Field Name (ID): Description
15:0	TBAV	**Device ID (DID):** See Global Device ID table in Chapter 2 for value.

35.6.3 Command (CMD)—Offset 4h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 1

Default: 0h

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD					ID	FBE	SERR	WCC	PER	VPS	MWI	SCE	BME	MSE	IOSE

Bit Range	Default and Access	Field Name (ID): Description
15:11	0h RO	Reserved.
10	0h RO	**Interrupt Disable (ID):** MROM does not have an interrupt to disable, so this bit is hardwired to 0.
9	0h RO	**Fast Back to Back Enable (FBE)**
8	0h RW	**SERR# Enable (SERR):** When set to 1, the XHC is capable of generating (internally) SERR#. See section on error handling.
7	0h RO	**Wait Cycle Control (WCC)**
6	0h RW	**Parity Error Response (PER):** This bit controls the setting of the master data parity error bit in the status register in response to a parity error received.
5	0h RO	**VGA Palette Snoop (VPS)**
4	0h RO	**Memory Write Invalidate (MWI)**
3	0h RO	**Special Cycle Enable (SCE)**
2	0h RW	**Bus Master Enable (BME):** This bit controls the ability of the function to issue memory and I/O read or write requests.
1	0h RW	**Memory Space Enable (MSE):** This bit controls the response to memory space accesses. When 0b, the function will handle memory transactions targeting it as an Unsupported request (UR). For Type 1 configuration space headers, this bit controls the primary side response to memory space accesses targeting the secondary side. When this bis is 0b, every memory transaction targeting a secondary interface is handled as an Unsupported Request (UR).
0	0h RO	**I/O Space Enable (IOSE):** Reserved as 0. Read-Only.

35.6.4 Device Status (STS)—Offset 6h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 1

Default: 290h

15			12				8				4				0
0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0
DPE	SSE	RMA	RTA	STA	DEVT		MDPED	FBBC	UDF	MC	CL	IS	RSVD		

Bit Range	Default and Access	Field Name (ID): Description
15	0h RW/1C	**Detected Parity Error (DPE):** This bit is set by the Intel PCH whenever a parity error is seen on the internal interface, regardless of the setting of bit 6 or bit 8 in the Command register or any other conditions. Software clears this bit by writing a 1 to this bit location.
14	0h RW/1C	**Signaled System Error (SSE):** This bit is set by the Intel PCH whenever it signals SERR# (internally). The SERR_EN bit (bit 8 in the Command Register) must be 1 for this bit to be set. See error handling section for complete list of conditions handled. Software clears this bit by writing a 1 to this bit location.
13	0h RW/1C	**Received Master-Abort Status (RMA):** This bit is set when the requester receives a completion with a UR completion status. Software clears this bit by writing a 1 to this bit location.
12	0h RW/1C	**Received Target Abort Status (RTA):** This bit is set upon receiving a CA completion status. Software clears this bit by writing a 1 to this bit location.
11	0h RW/1C	**Signaled Target-Abort Status (STA):** This bit is used to indicate when the function generates a completion packet with Completer Abort (CA) status.
10:9	1h RO	**DEVSEL# Timing Status (DEVT):** This 2-bit field defines the timing for DEVSEL# assertion. Read-Only.
8	0h RW/1C	**Master Data Parity Error Detected (MDPED):** This bit is set by the Intel PCH whenever a data parity error is detected in a read completion packet on the internal interface and bit 6 of the Command register is set to 1. Software clears this bit by writing a 1 to this bit location.
7	0h RO	**Fast Back-to-Back Capable (FBBC):** Reserved as 0 Read-Only.
6	0h RO	**User Definable Features (UDF):** Reserved as 0. Read-Only.
5	0h RO	**66 MHz Capable (MC):** Reserved as 0. Read-Only.
4	1h RO	**Capabilities List (CL):** Hardwired to 1 indicating that offset 34h contains a valid capabilities pointer.
3	0h RO/V	**Interrupt Status (IS):** Interrupts are not supported so this bit always returns "0" when read.
2:0	0h RO	Reserved.

35.6.5 Revision ID (RID)—Offset 8h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 1

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
RID							

Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO/V	**Revision ID (RID):** RIDs are identified in the datasheet specification updates.

35.6.6 Programming Interface (PI)—Offset 9h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 1

Default: 30h

7			4				0
0	0	1	1	0	0	0	0
PI							

Bit Range	Default and Access	Field Name (ID): Description
7:0	00h RO	**Programming Interface (PI):** Default value of 00h.

35.6.7 Sub Class Code (SCC)—Offset Ah

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 1

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
SCC							

Bit Range	Default and Access	Field Name (ID): Description
7:0	00h RO	**Sub Class Code (SCC):** Default value of 00h

35.6.8 Base Class Code (BCC)—Offset Bh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 1

Default: Ch

7			4				0
0	0	0	0	1	1	0	0
BCC							

Bit Range	Default and Access	Field Name (ID): Description
7:0	FFh RO	**Base Class Code (BCC):** A value of FFh indicates that this does not fit into any defined class.

35.6.9 Cachleline Size Register (CLS)—Offset Ch

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 1

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
CLS							

Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW	**Cacheline Size Register (CLR):** This register has no meaning for MROM.

35.6.10 Master Latency Timer (MLT)—Offset Dh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 1

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
MLT							

Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO	**Master Latency Timer (MLT):** The bits will be fixed at 0 as this register has no meaning for MROM.

35.6.11 Header Type (HT)—Offset Eh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 1

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
MFB	CL						

Bit Range	Default and Access	Field Name (ID): Description
7	0h RO	**Multi-Function Bit (MFB):** Defaults to 0, but if after reset MROM1 or SSATA is enabled then this bit will become a “1”.
6:0	0h RO	**Configuration layout (CL):** Hardwired to 0 to indicate a standard PCI configuration layout.

Bit Range	Default and Access	Field Name (ID): Description
63:16	0h RW	**Base Address (BA):** Bits [63:16] correspond to memory address signals [63:16], respectively. This gives 64 KB of relocatable memory space aligned to 64 KB boundaries.
15:4	0h RO	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
3	0h RO	**Prefetchable (Prefetchable):** This bit is hardwired to 0 indicating that this range should not be prefetched.
2:1	2h RO	**Type (Type):** If this field is hardwired to 00 it indicates that this range can be mapped anywhere within 32-bit address space. If this field is hardwired to 10 it indicates that this range can be mapped anywhere within 64-bit address space.
0	0h RO	**Resource Type Indicator (RTE):** This bit is hardwired to 0 indicating that the base address field in this register maps to memory space

35.6.12 Subsystem Vendor ID (SSVID)—Offset 2Ch

This register is modified and maintained by BIOS

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 1

Default: 8086h

15			12				8				4				0
1	0	0	0	1	0	0	0	1	0	0	0	0	1	1	0
SSVID															

Bit Range	Default and Access	Field Name (ID): Description
15:0	8086h RW/L	**Subsystem Vendor ID (SSVID):** This field identifies Intel as the manufacturer of the device.

35.6.13 Subsystem ID (SSID)—Offset 2Eh

This register is modified and maintained by BIOS.

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 1

Default: 0h

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SSID															

Bit Range	Default and Access	Field Name (ID): Description
15:0	0h RW/L	**Subsystem ID (SSID):**

35.6.14 Expansion ROM Base Address Register (EROMBAR)—Offset 30h

Access Method

Type: CFG Register
(Size: 32bits)

Device: 17
Function: 1

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RBAUB								RBAB23	RBAB22	RBAB21	RBAB20	RBAB19	RBAB18	RBAB17	RBAB16	RBAB15	RSVD														ERE

Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RW-LV	**ROM Base Address Upper Bits (RBAUB):** The ROM base address includes both RBAUB and RBAB (23:15) bits below. The RBAUB bits correspond to bits 31:24 0f the ROM base address value.
23	0h RW-LV	**ROM Base Address Bit 23(RBAB23):** Bit 23 becomes RO and locked with ROM is 16M, size = 0h. All MSIZE bits are RW if MROMCTL[size]>=9h, indicating 32K, unless otherwise locked as described.
22	0h RW-LV	**ROM Base Address Bit 22(RBAB22):** Bit 22 becomes RO and locked with ROM is 8M or larger, size <= 1h. All MSIZE bits are RW if MROMCTL[size]>=9h, indicating 32K, unless otherwise locked as described.
21	0h RW-LV	**ROM Base Address Bit 21(RBAB21):** Bit 21 becomes RO and locked with ROM is 4M or larger, size <= 2h. All MSIZE bits are RW if MROMCTL[size]>=9h, indicating 32K, unless otherwise locked as described.
20	0h RW-LV	**ROM Base Address Bit 20(RBAB20):** Bit 20 becomes RO and locked with ROM is 2M or larger, size <= 3h. All MSIZE bits are RW if MROMCTL[size]>=9h, indicating 32K, unless otherwise locked as described.
19	0h RW-LV	**ROM Base Address Bit 19(RBAB19):** Bit 19 becomes RO and locked with ROM is 1M, size <= 4h. All MSIZE bits are RW if MROMCTL[size]>=9h, indicating 32K, unless otherwise locked as described.
18	0h RW-LV	**ROM Base Address Bit 18(RBAB18):** Bit 18 becomes RO and locked with ROM is 512K or larger, size< = 5h. All MSIZE bits are RW if MROMCTL[size]>=9h, indicating 32K, unless otherwise locked as described.
17	0h RW-LV	**ROM Base Address Bit 17(RBAB17):** Bit 17 becomes RO and locked with ROM is 256K or larger, size <= 6h. All MSIZE bits are RW if MROMCTL[size]>=9h, indicating 32K, unless otherwise locked as described.
16	0h RW-LV	**ROM Base Address Bit 16(RBAB16):** Bit 16 becomes RO and locked with ROM is 128K or larger, size <= 7h. All MSIZE bits are RW if MROMCTL[size]>=9h, indicating 32K, unless otherwise locked as described.

Bit Range	Default and Access	Field Name (ID): Description
15	0h RW-LV	**ROM Base Address Bit 15(RBAB15):** Bit 15 becomes RO and locked with ROM is 64K or larger, size <= 8h. All MSIZE bits are RW if MROMCTL[size]>=9h, indicating 32K, unless otherwise locked as described.
14:1	0;RO	Reserved
0	0h RW-LV	**Expansion ROM Enable (ERE):** 0b: ROM Disabled 1b: Expansion ROM enabled Not writable if MROMCTL[EMROM] = 0 or PRIBLKCTL{ERBD] =1 as described above. If ERE = 1b, but PCICMD[MSE] = 0b, then this address region will no be enabled, as PCICMD[MSE] takes precedence. Both bits need to be 1 in order to enabled address decoding. BIOS must disable ROMBAR to prevent further MROM access prior to the transition to OS. This register can be locked down by PRIBLKCTL[ERBD]=1.

35.6.15 Capabilities Pointer (CAP_PTR)—Offset 34h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 1

Default: 80h

7			4				0
1	0	0	0	0	0	0	0
CAP_PTR							

Bit Range	Default and Access	Field Name (ID): Description
7:0	80h RO	**Capabilities Pointer (CAP_PTR):** This register points to the starting offset of the capabilities ranges.

35.6.16 Interrupt Line (ILINE)—Offset 3Ch

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 1

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
ILINE							

Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW	**Interrupt Line (ILINE):** This data is not used by the Intel PCH. It is used as a scratchpad register to communicate to software the interrupt line that the interrupt pin is connected to. A value of 0xFF defines no connection. BIOS should write FFh to this field.

35.6.17 Interrupt Pin (IPIN)—Offset 3Dh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 17
Function: 1

Default: 0h

7			4				0
0	0	0	0	0	0	0	0
IPIN							

Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RW/L	**Interrupt pin (IPIN):** Bits [7:0] reflect the Interrupt Pin assigned to the host controller by the platform (and are hardwired).

35.6.18 Power Management Capability List Register ID (PMCAPLST)—Offset 80h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 1

Default: 1h

15			12				8				4				0
0	0	0	0	0	0	0	0	0I	0	0	0	0	0	0	1
NP								CPID							

Bit Range	Default and Access	Field Name (ID): Description
7:0	1h RO	**PCI Power Management Capability ID (PM_CID):** A value of 01h indicates that this is a PCI Power Management capabilities field.
15:8	00h RW-L	**Next Pointer (NP):** Contains the offset of the next item in the capabilities list. A null value is used to indicate that this is the last capability.

35.6.19 Power Management Capabilities (PM_CAP)—Offset 82h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 1

Default: 3h

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
PME_Support					D2_Support	D1_Support	Aux_Current			DSI	RSVD	PMEClock	Version		

Bit Range	Default and Access	Field Name (ID): Description
15:11	00h RO	**PME_Support (PME_Support):** This 5-bit field indicates the power states in which the function may assert PME#. A value of 00h implies that this device does not generate PME events.
10	0h RO	**D2_Support (D2_Support):** The D2 state is not supported.
9	0h RO	**D1_Support (D1_Support):** The D1 state is not supported.
8:6	0h RO	**Aux_Current (Aux_Current):** Auxiliary power is not supported.
5	0h Ro	**DSI (DSI):** The Intel PCH reports 0, indicating that no device-specific initialization is required.
4	0h RO	Reserved.
3	0h RW/L	**PME Clock (PMEClock):** The Intel PCH reports 0, indicating that no PCI clock is required to generate PME#.
2:0	3h RW/L	**Version (Version):** Complies with Revision 1.2 of the PCI Power Management Specification.

35.6.20 Power Management Control/Status (PM_CS)—Offset 84h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 1

Default: 8h

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
PME_Status	Data_Scale		Data_Select				PME_En	RSVD				NSR	RSVD	PowerState	

Bit Range	Default and Access	Field Name (ID): Description
15	0h RW/1C	**PME_Status (PME_Status):** This bit is set when the function would normally assert the PME# signal independent of the state of the PME_En bit. Writing a 1 to this bit will clear it and cause the internal PME to deassert (if enabled). Writing a 0 has no effect. This bit must be explicitly cleared by the operating system each time the operating system is loaded.
14:13	0h RO	**Data_Scale (Data_Scale):** Not supported so hardwired to 00.
12:9	0h RO	**Data_Select (Data_Select):** Not supported so hardwired to 00.
8	0h RW	**PME_En (PME_En):** A 1 enables the function to generate an internal PME signal when PME_Status is 1. This bit must be explicitly cleared by the operating system each time it is initially loaded.
7:4	0h RO	Reserved.
3	1h RO	**No Soft Reset (NSR):** this bit indicates that devices transitioning from $D3_{HOT}$ to D0 because of PowerState commands do not perform an internal reset. Configuration Context is preserved. Upon transition from the $D3_{HOT}$ to the D0 Initialized state, no additional operating system intervention is required to preserve Configuration Context beyond writing the PowerState bits.
2	0h RO	Reserved.
1:0	0h RW	**PowerState (PowerState):** This 2-bit field is used both to determine the current power state and to set a new power state. The definition of the field values are: 00b - D0 state 11b - $D3_{HOT}$ state If software attempts to write a value of 10b or 01b in to this field, the write operation must complete normally, however, the data is discarded and no state change occurs.

35.6.21 Private Block Control Register (PRIBLKCTL)—Offset E0h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 17
Function: 1

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ERBD	RSVD																														DNEERB

Bit Range	Default and Access	Field Name (ID): Description
31	0h RO	**Expansion ROM BAR Disable (ERBD):** This bit is self locking. Once set to one can not be cleared by writing a "0" to it. Setting this to "1" makes all bits in the EROMBAR register RO
30:1	0h RW	These bits are reserved and must remain "0".
0	0b RO	**Do Not Enable Expansion ROM Bar (DNEERB);** This is a read-only bit that reflects the state of a fuse bit. The fuse does not directly control anything but is used as a signal for BIOS and FW to enable the expansion ROM BAR or not.

35.6.22 Device Clock Gate Control Register (DEVCLKGCTL)—Offset E4h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 1

Default: 0h

15			12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ICGE	RSVD							ICT							

Bit Range	Default and Access	Field Name (ID): Description
15	0h Rw	**Idle Clock Gate Enable (ICGE):** When set this bit enables the IP block to enter clock gating mode, where the clocks are turned off to save power
14:8	0h RW	These bits are reserved and must remain "0".
7:0	0b Rw	**Idle Clock Timer (ICT):** This field indicates the number of clocks that the IP block must be idle for before the clock disable process begins. 01h = 1 clock and FFh = 255. The minimum number recommended is 16 (default)

35.6.23 Personality Lock Key Control Register (PLKCTL) —Offset E8h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 17
Function: 1

Default: 0000h

15			12				8				4				0
0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0
RSVD															CL

Bit Range	Default and Access	Field Name (ID): Description
15:1	0h RW	These bits are reserved and must remain "0".
0	0b Rw-L	**Capability Lock (CL):** Lock bit for all RW-L bits in all registers for this function. 0b= unlocked 1b = locked. Once a "1" is written to this bit it is only reset to 0 with a global reset.

§

36 Additional Configuration Registers

36.1 FIA Configuration PCR Registers Summary

These registers are within the PCH Private Configuration Space which is accessible through the PCH Sideband Interface. They can be accessed via (SBREG_BAR + PortID + Register Offset).

Table 36-1. Summary of FIA Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	Common Control (CC)—Offset 0h	0h
100h	103h	Device Reference Clock Request Mapping 1 (DRCRM1)—Offset 100h	76543210h
104h	107h	Device Reference Clock Request Mapping 2 (DRCRM2)—Offset 104h	FEDCBA98h
108h	10Bh	Device Reference Clock Request Mapping 3 (DRCRM3)—Offset 108h	F000FFFFh
224	227	PCIe Uplink and Downlink Strap Configuration (PCIEUDLSTRPCFG)	0h
250h	253h	Lane Owner Status 1 (LOS1)—Offset 250h	0h
254h	257h	Lane Owner Status 2 (LOS2)—Offset 254h	0h
258h	25Bh	Lane Owner Status 3 (LOS3)—Offset 258h	0h
25Ch	25Fh	Lane Owner Status 4 (LOS4)—Offset 25Ch	0h

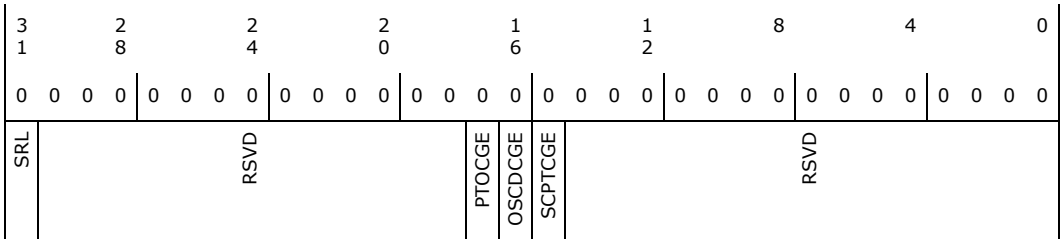
36.1.1 Common Control (CC)—Offset 0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RW/O	**Secured Register Lock (SRL):** When this bit is set, all the secured registers will be locked and will be Read-Only.
30:18	0h RO	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
17	0h RW	**Partition/Trunk Oscillator Clock Gating Enable (PTOCGE):** When set, the oscillator and side clock will be dynamically clock gated when the conditions to clock gate are met. When clear, the oscillator and side clock will never be dynamically clock gated.
16	0h RW	**Oscillator/Side Clock Dynamic Clock Gating Enable (OSCDCGE):** When set, the oscillator and side clock will be dynamically clock gated when the conditions to clock gate are met. When clear, the oscillator and side clock will never be dynamically clock gated.
15	0h RW	**Side Clock Partition/Trunk Clock Gating Enable (SCPTCGE):** When set, the Side Clock will be clock gated at the partition/trunk level when the conditions to clock gate are met. When clear, the Side Clock will never be clock gated at the partition/trunk level.
14:0	0h RO	Reserved.

36.1.2 Device Reference Clock Request Mapping 1 (DRCRM1)—Offset 100h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 76543210h

31			28				24				20				16				12				8				4				0
0	1	1	1	0	1	1	0	0	1	0	1	0	1	0	0	0	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
P8CKRQM				P7CKRQM				P6CKRQM				P5CKRQM				P4CKRQM				P3CKRQM				P2CKRQM				P1CKRQM				

Bit Range	Default and Access	Field Name (ID): Description
31:28	7h RW/L	**PCI Express Port 7 CLKREQ Mapping (P7CKRQM):** Same description as bit [3:0], except that this field applies to PCIe Port 8.
27:24	6h RW/L	**PCI Express Port 6 CLKREQ Mapping (P6CKRQM):** Same description as bit [3:0], except that this field applies to PCIe Port 7.
23:20	5h RW/L	**PCI Express Port 5 CLKREQ Mapping (P5CKRQM):** Same description as bit [3:0], except that this field applies to PCIe Port 6.
19:16	4h RW/L	**PCIe Express Port 4 CLKREQ Mapping (P4CKRQM):** Same description as bit [3:0], except that this field applies to PCIe Port 5.
15:12	3h RW/L	**PCI Express Port 3 CLKREQ Mapping (P3CKRQM):** Same description as bit [3:0], except that this field applies to PCIe Port 4.

Bit Range	Default and Access	Field Name (ID): Description
11:8	2h RW/L	**PCI Express Port 2 CLKREQ Mapping (P2CKRQM):** Same description as bit [3:0], except that this field applies to PCIe Port 3.
7:4	1h RW/L	**PCI Express Port 1 CLKREQ Mapping (P1CKRQM):** Same description as bit [3:0], except that this field applies to PCIe Port 2.
3:0	0h RW/L	**PCI Express Port 0 CLKREQ Mapping (P0CKRQM):** The mapping of PCI Express Port 0 to the corresponding CLKREQ# pin is configured by this field. 0000b: Port 0 maps to CLKREQ0# pin. 0001b: Port 0 maps to CLKREQ1# pin. 0010b: Port 0 maps to CLKREQ2# pin. 0011b: Port 0 maps to CLKREQ3# pin. 0100b: Port 0 maps to CLKREQ4# pin. 0101b: Express Port 0 maps to CLKREQ5# pin. ....[br 1111b: Port 0 maps to CLKREQ15# pin. Software must never map multiple Express Ports to the same CLKREQ# pin. The hardware behavior is undefined if this ever happens. ***Notes:*** This field must be configured prior to enabling any power management features. This register bit is Read-Only when the CC.SRL bit is set. Register Attribute: Static.

36.1.3 Device Reference Clock Request Mapping 2 (DRCRM2)—Offset 104h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: FEDCBA98h

31			28				24				20				16				12				8				4				0
1	1	1	1	1	1	1	0	1	1	0	1	1	1	0	0	1	0	1	1	1	0	1	0	1	0	0	1	1	0	0	0
P16CKRQM				P15CKRQM				P14CKRQM				P13CKRQM				P12CKRQM				P11CKRQM				P10CKRQM				P9CKRQM			

Bit Range	Default and Access	Field Name (ID): Description
31:28	Fh RW/L	**PCI Express Port 15 CLKREQ Mapping (P15CKRQM):** Same description as bit [3:0], except that this field applies to PCIe Port 16.
27:24	Eh RW/L	**Express Port 14 CLKREQ Mapping (P14CKRQM):** Same description as bit [3:0], except that this field applies to PCIe Port 15.
23:20	Dh RW/L	**PCI Express Port 13 CLKREQ Mapping (P13CKRQM):** Same description as bit [3:0], except that this field applies to PCIe Port 14.
19:16	Ch RW/L	**PCI Express Port 12 CLKREQ Mapping (P12CKRQM):** Same description as bit [3:0], except that this field applies to PCIe Port 13.
15:12	Bh RW/L	**PCI Express Port 11 CLKREQ Mapping (P11CKRQM):** Same description as bit [3:0], except that this field applies to PCIe Port 12.

Bit Range	Default and Access	Field Name (ID): Description
11:8	Ah RW/L	**PCI Express Port 10 CLKREQ Mapping (P10CKRQM):** Same description as bit [3:0], except that this field applies to PCIe Port 11.
7:4	9h RW/L	**PCI Express Port 9 CLKREQ Mapping (P9CKRQM):** Same description as bit [3:0], except that this field applies to PCIe Port 10.
3:0	8h RW/L	**PCI Express Port 8 CLKREQ Mapping (P8CKRQM):** The mapping of PCI Express Port 9 to the corresponding CLKREQ# pin is configured by this field. 0000b: Port 8 maps to CLKREQ0# pin. 0001b: Port 8 maps to CLKREQ1# pin. 0010b: Port 8 maps to CLKREQ2# pin. 0011b: Port 8 maps to CLKREQ3# pin. 0100b: Port 8 maps to CLKREQ4# pin. 0101b: Express Port 1 maps to CLKREQ5# pin. ....[br 1111b: Port 8 maps to CLKREQ15# pin. Software must never map multiple Express Ports to the same CLKREQ# pin. The hardware behavior is undefined if this ever happens. ***Notes:*** This field must be configured prior to enabling any power management features. This register bit is Read-Only when the CC.SRL bit is set. Register Attribute: Static.

36.1.4 Device Reference Clock Request Mapping 3 (DRCRM3)—Offset 108h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: F000FFFFh

31			28				24				20				16				12				8				4				0
1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
GBEPCKRQM				RSVD												P20CKRQM				P19CKRQM				P18CKRQM				P17CKRQM				

Bit Range	Default and Access	Field Name (ID): Description
31:28	Fh RW/L	**GbE Port CLKREQ Mapping (GBEPCKRQM):** The mapping of GbE port to the corresponding CLKREQ# pin is configured by this field. 0000b: GbE port maps to CLKREQ0# pin. 0001b: GbE port maps to CLKREQ1# pin. 0010b: GbE port maps to CLKREQ2# pin. 0011b: GbE port maps to CLKREQ3# pin. 0100b: GbE port maps to CLKREQ4# pin. 0101b: GbE port maps to CLKREQ5# pin. ..... 1111b: GbE port maps to CLKREQ15# pin. Software must never map multiple Express Ports to the same CLKREQ# pin. The hardware behavior is undefined if this ever happens. ***Notes:*** This field must be configured prior to enabling any power management features. This register bit is Read-Only when the CC.SRL bit is set. Register Attribute: Static.
27:16	0h RO	Reserved.
15:12	Fh RW/L	**PCI Express Port 19 CLKREQ Mapping (P19CKRQM):** Same description as bit [3:0], except that this field applies to PCIe Port 20.
11:8	Fh RW/L	**PCI Express Port 18 CLKREQ Mapping (P18CKRQM):** Same description as bit [3:0], except that this field applies to PCIe Port 19.
7:4	Fh RW/L	**PCI Express Port 17 CLKREQ Mapping (P17CKRQM):** Same description as bit [3:0], except that this field applies to PCIe Port 18.
3:0	Fh RW/L	**PCI Express Port 16 CLKREQ Mapping (P16CKRQM):** The mapping of PCI Express Port 16 to the corresponding CLKREQ# pin is configured by this field. 0000b: Port 16 maps to CLKREQ0# pin. 0001b: Port 16 maps to CLKREQ1# pin. 0010b: Port 16 maps to CLKREQ2# pin. 0011b: Port 16 maps to CLKREQ3# pin. 0100b: Port 16 maps to CLKREQ4# pin. 0101b: Express Port 16 maps to CLKREQ5# pin. ....[br 1111b: Port 16 maps to CLKREQ15# pin. Software must never map multiple Express Ports to the same CLKREQ# pin. The hardware behavior is undefined if this ever happens. ***Note:*** This field must be configured prior to enabling any power management features. This register bit is Read-Only when the CC.SRL bit is set. Register Attribute: Static.

36.1.5 PCIe Uplink and Downlink Strap Configuration (PCIEUDLSTRPCFG)—Offset 224h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVDd																															pcieuds

Bit Range	Default and Access	Field Name (ID): Description
31:1	00000000h RO	**RSVD:**
0	RO/S	**PCIe Uplink and Downlink Strap (PCIEUDS):** 0: Statically assigned Lanes 25:18 to PCIe Downlink 1: Statically assigned Lanes 25:18 to PCIe Uplink

36.1.6 Lane Owner Status 1 (LOS1)—Offset 250h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
L7O				L6O				L5O				L4O				L3O				L2O				L1O				L0O			

Bit Range	Default and Access	Field Name (ID): Description
31:28	0h RO/V	**Lane 7 Owner (L7O):** Same description as bits [3:0].
27:24	0h RO/V	**Lane 6 Owner (L6O):** Same description as bits [3:0].
23:20	0h RO/V	**Lane 5 Owner (L5O):** Same description as bits [3:0].
19:16	0h RO/V	**Lane 4 Owner (L4O):** Same description as bits [3:0].

Bit Range	Default and Access	Field Name (ID): Description
15:12	0h RO/V	**Lane 3 Owner (L3O):** Same description as bits [3:0].
11:8	0h RO/V	**Lane 2 Owner (L2O):** Same description as bits [3:0].
7:4	0h RO/V	**Lane 1 Owner (L1O):** Same description as bits [3:0].
3:0	0h RO/V	**Lane 0 Owner (L0O):** Lane 0 Owner (L0O): This register indicates the lane owner for Lane 0. 0000: PCIe/DMI. 0001: USB3. 0010: SATA. 0011: GbE. Others: Reserved.

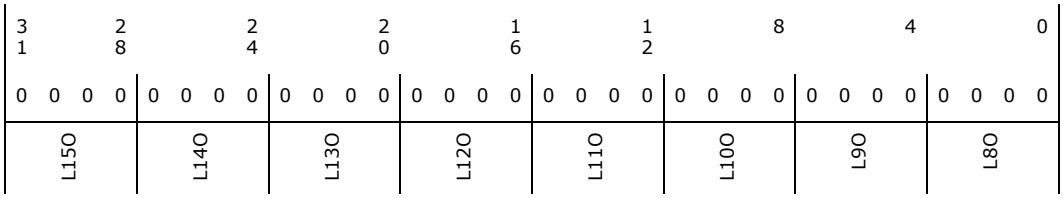
36.1.7 Lane Owner Status 2 (LOS2)—Offset 254h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:28	0h RO/V	**Lane 15 Owner (L15O):** Same description as bits [3:0].
27:24	0h RO/V	**Lane 14 Owner (L14O):** Same description as bits [3:0].
23:20	0h RO/V	**Lane 13 Owner (L13O):** Same description as bits [3:0].
19:16	0h RO/V	**Lane 12 Owner (L12O):** Same description as bits [3:0].
15:12	0h RO/V	**Lane 11 Owner (L11O):** Same description as bits [3:0].
11:8	0h RO/V	**Lane 10 Owner (L10O):** Same description as bits [3:0].
7:4	0h RO/V	**Lane 9 Owner (L9O):** Same description as bits [3:0].

Bit Range	Default and Access	Field Name (ID): Description
3:0	0h RO/V	**Lane 8 Owner (L8O):** Lane 8 Owner (L8O): This register indicates the lane owner for Lane 0. 0000: PCIe/DMI. 0001: USB3. 0010: SATA. 0011: GbE. Others: Reserved.

36.1.8 Lane Owner Status 3 (LOS3)—Offset 258h

Size:32 bits
FIA Private Configuration Register: Offset 258h: LOS3 - Lane Owner Status 3

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
L23O				L22O				L21O				L20O				L19O				L18O				L17O				L16O			

Bit Range	Default and Access	Field Name (ID): Description
31:28	0h RO/V	**Lane 23 Owner (L23O):** Same description as bits [11:8].
27:24	0h RO/V	**Lane 22 Owner (L22O):** Same description as bits [11:8].
23:20	0h RO/V	**Lane 21 Owner (L21O):** Same description as bits [11:8].
19:16	0h RO/V	**Lane 20 Owner (L20O):** Same description as bits [11:8].
15:12	0h RO/V	**Lane 19 Owner (L19O):** Same description as bits [11:8].
11:8	0h RO/V	**Lane 18 Owner (L18O):** S 0000: PCIe/DMI/PCIe Uplink 0001: USB3. 0010: SATA. 0011: GbE. Others: Reserved.
7:4	0h RO/V	**Lane 17 Owner (L17O):** Same description as bits [3:0].
3:0	0h RO/V	**Lane 16 Owner (L16O):** Lane 16 Owner (L16O): This register indicates the lane owner for Lane 0. 0000: PCIe/DMI. 0001: USB3. 0010: SATA. 0011: GbE. Others: Reserved.

36.1.9 Lane Owner Status 4 (LOS4)—Offset 25Ch

FIA Private Configuration Register: Offset 25Ch: LOS4 - Lane Owner Status 4

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD								L29O				L28O				L27O				L26O				L25O				L24O			

Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23:20	0h RO	Reserved.
19:16	0h RO	Reserved.
15:12	0h RO	Reserved.
11:8	0h RO	Reserved.
7:4	0h RO/V	**Lane 25 Owner (L25O):** Same description as bits [3:0].
3:0	0h RO/V	**Lane 24 Owner (L24O):** Lane 24 Owner (L24O): This register indicates the lane owner for Lane 0. 0000: PCIe/DMI/PCIe Uplink 0001: USB3. 0010: SATA. 0011: GbE. Others: Reserved.

36.2 DMI Configuration Registers Summary

Table 36-2. Summary of DMI Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
2014h	2017h	Virtual Channel 0 Resource Control (V0CTL)—Offset 2014h	80000010h
2018h	201Bh	Virtual Channel 0 Resource Status (V0STS)—Offset 2018h	0h
2020h	2023h	Virtual Channel 1 Resource Control (V1CTL)—Offset 2020h	0h
2024h	2027h	Virtual Channel 1 Resource Status (V1STS)—Offset 2024h	0h
2040h	2043h	ME Virtual Channel (VCm) Resource Control (VMCTL)—Offset 2040h	0h
2046h	2049h	ME Virtual Channel (VCm) Resource Status (VMSTS)—Offset 2046h	0h
2084h	2087h	Uncorrectable Error Status (UES)—Offset 2084h	0h

Table 36-2. Summary of DMI Configuration Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
2088h	208Bh	Uncorrectable Error Mask (UEM)—Offset 2088h	0h
208Ch	208Fh	Uncorrectable Error Severity (UEV)—Offset 208Ch	0h
2090h	2093h	Correctable Error Status (CES)—Offset 2090h	0h
2094h	2097h	Correctable Error Mask (CEM)—Offset 2094h	2000h
20ACh	20AFh	Root Error Command (REC)—Offset 20ACh	0h
20B0h	20B3h	Root Error Status (RES)—Offset 20B0h	0h
20B4h	20B7h	Error Source Identification (ESID)—Offset 20B4h	0h
21A4h	21A7h	Link Capabilities (LCAP)—Offset 21A4h	12C40h
21A8h	21ABh	Link Control and Link Status (LCTL_LSTS)—Offset 21A8h	410000h
21ACh	21AFh	Link Capabilities 2 (LCAP2)—Offset 21ACh	0h
21B0h	21B3h	Link Control 2 and Link Status 2 (LCTL2_LSTS2)—Offset 21B0h	1h
2234h	2237h	DMI Control Register (DMIC)—Offset 2234h	0h
223Ch	223Fh	IOSF Primary Control And Status (IPCS_IOSFSBCS)—Offset 223Ch	0h
2304h	2307h	DMI Port Link Control (DMILINKC)—Offset 2304h	0h
2310h	2313h	DMI Configuration (DMICFG)—Offset 2310h	0h
2320h	2323h	DMI PLL Shutdown (DMIPLLDOWN)—Offset 2320h	0h
2334h	2337h	DMI Power Management Control (DMIPMCTL)—Offset 2334h	0h
2338h	233Bh	DMI Additional Link Control (DMIALC)—Offset 2338h	0h
2340h	2343h	DMI NFTS (DMINFTS)—Offset 2340h	0h
2730h	2733h	LPC Generic I/O Range 1 (LPCLGIR1)—Offset 2730h	0h
2734h	2737h	LPC Generic I/O Range 2 (LPCLGIR2)—Offset 2734h	0h
2738h	273Bh	LPC Generic I/O Range 3 (LPCLGIR3)—Offset 2738h	0h
273Ch	273Fh	LPC Generic I/O Range 4 (LPCLGIR4)—Offset 273Ch	0h
2740h	2743h	LPC Generic Memory Range (LPCGMR)—Offset 2740h	0h
2744h	2747h	LPC BIOS Decode Enable (LPCBDE)—Offset 2744h	FFCFh
274Ch	274Fh	General Control and Status (GCS)—Offset 274Ch	0h
2750h	2753h	I/O Trap Register 1 low (IOT1_LOW)—Offset 2750h	0h
2754h	2757h	I/O Trap Register 1 high (IOT1_HIGH)—Offset 2754h	0h
2770h	2773h	LPC I/O Decode Range (LPCIOD)—Offset 2770h	0h
2774h	2777h	LPC I/O Enable (LPCIOE)—Offset 2774h	0h
2778h	277Bh	TCO Base Address (TCOBASE)—Offset 2778h	0h
27B0h	27B3h	PM Base Control (PMBASEC)—Offset 27B0h	0h
27B4h	27B7h	ACPI Base Address (ACPIBA)—Offset 27B4h	0h
27B8h	27BBh	ACPI Base Destination ID (ACPIBDID)—Offset 27B8h	0h

36.2.1 Virtual Channel 0 Resource Control (V0CTL)—Offset 2014h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 80000010h

31			28				24				20				16				12				8				4				0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
EN	RSVD							ID				RSVD								ETVM				RSVD					TVM				RSVD

Bit Range	Default and Access	Field Name (ID): Description
31	1h RO	**Virtual Channel Enable (EN):** Enables the VC when set. Disables the VC when cleared.
30:27	0h RO	Reserved.
26:24	0h RO	**Virtual Channel Identifier (ID):** Indicates the ID to use for this virtual channel.
23:16	0h RO	Reserved.
15:10	0h RW/L	**Extended TC/VC Map (ETVM):** Defines the upper 8-bits of the VC0 16-bit TC/VC mapping registers. These registers use the PCI Express reserved TC[3] traffic class bit. This register is Read-Only if DMIC.SRL field is set.
9:7	0h RO	Reserved.
6:1	8h RW/L	**Transaction Class / Virtual Channel Map (TVM):** Indicates which transaction classes are mapped to this virtual channel. When a bit is set, this transaction class is mapped to the virtual channel. This register is Read-Only if DMIC.SRL field is set.
0	0h RO	Reserved.

36.2.2 Virtual Channel 0 Resource Status (V0STS)—Offset 2018h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD															NP	RSVD																

Bit Range	Default and Access	Field Name (ID): Description
31:18	0h RO	Reserved.
17	0h RO/V	**VC Negotiation Pending (NP):** When set, indicates the virtual channel is still being negotiated with ingress ports.
16:0	0h RO	Reserved.

36.2.3 Virtual Channel 1 Resource Control (V1CTL)—Offset 2020h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
EN	RSVD			ID				RSVD								ETVM				RSVD				TVM								RSVD

Bit Range	Default and Access	Field Name (ID): Description
31	0h RW/L	**Virtual Channel Enable (EN):** Enables the VC when set. Disables the VC when cleared.
30:28	0h RO	Reserved.
27:24	0h RW/L	**Virtual Channel Identifier (ID):** Indicates the ID to use for this virtual channel. ***Note:*** BIOS is required to program VCID[3] to 0 when operating at DMI2.
23:16	0h RO	Reserved.
15:10	0h RW/L	**Extended TC/VC Map (ETVM):** Defines the upper 8-bits of the VC1 16-bit TC/VC mapping registers. These registers use the PCI Express reserved TC[3] traffic class bit. This register is Read-Only if the DMIC.SRL field is set. If VC1 is not enabled, the register output sent to the users of this register is forced to 0.
9:8	0h RO	Reserved.
7:1	0h RW/L	**Transaction Class / Virtual Channel Map (TVM):** Indicates which transaction classes are mapped to this virtual channel. When a bit is set, this transaction class is mapped to the virtual channel. This register is Read-Only if the DMIC.SRL field is set. If VC1 is not enabled, the register output sent to the users of this register is forced to 0.
0	0h RO	Reserved.

36.2.4 Virtual Channel 1 Resource Status (V1STS)—Offset 2024h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD															NP	RSVD															

Bit Range	Default and Access	Field Name (ID): Description
31:18	0h RO	Reserved.
17	0h RO/V	**VC Negotiation Pending (NP):** When set, indicates the virtual channel is still being negotiated with ingress ports.
16:0	0h RO	Reserved.

36.2.5 ME Virtual Channel (VCm) Resource Control (VMCTL)—Offset 2040h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
EN	RSVD			ID				RSVD								ETVM				RSVD				TVM							RSVD

Bit Range	Default and Access	Field Name (ID): Description
31	0h RW/L	**Virtual Channel Enable (EN):** Enables the VC when set. Disables the VC when cleared.
30:28	0h RO	Reserved.
27:24	0h RW/L	**Virtual Channel Identifier (ID):** Indicates the ID to use for this virtual channel. ***Note:*** BIOS is required to program VCID[3] to 0 when operating at DMI2.
23:16	0h RO	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
15:10	0h RW/L	**Extended TC/VC Map (ETVM):** Defines the upper 8-bits of the VCm 16-bit TC/VC mapping registers. These registers use the PCI Express reserved TC[3] traffic class bit. This register is Read-Only if the DMIC.SRL field is set. If VCm is not enabled, the register output sent to the users of this register is forced to 0.
9:8	0h RO	Reserved.
7:1	0h RW/L	**Transaction Class / Virtual Channel Map (TVM):** Indicates which transaction classes are mapped to this virtual channel. When a bit is set, this transaction class is mapped to the virtual channel. This register is Read-Only if the DMIC.SRL field is set. If VCm is not enabled, the register output sent to the users of this register is forced to 0.
0	0h RO	Reserved.

36.2.6 ME Virtual Channel (VCm) Resource Status (VMSTS)—Offset 2046h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD															NP	RSVD															

Bit Range	Default and Access	Field Name (ID): Description
31:18	0h RO	Reserved.
17	0h RO/V	**VC Negotiation Pending (NP):** When set, indicates the virtual channel is still being negotiated with ingress ports.
16:0	0h RO	Reserved.

36.2.7 Uncorrectable Error Status (UES)—Offset 2084h

These registers are reset by core PWROK

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD															RO	UC	CA	CT	FCPE	PT	RSVD							DLPE	RSVD			TE

Bit Range	Default and Access	Field Name (ID): Description
31:18	0h RO	Reserved.
17	0h RWC/P	**Receiver Overflow Status (RO):** Indicates a receiver overflow occurred.
16	0h RO	**Unexpected Completion Status (UC):** Reserved, not supported.
15	0h RWC/P	**Completer Abort Status (CA):** Indicates a completer abort was received.
14	0h RO	**Completion Timeout Status (CT):** Reserved, not supported.
13	0h RO	**Flow Control Protocol Error Status (FCPE):** Reserved, not supported.
12	0h RWC/P	**Poisoned TLP Status (PT):** Indicates a poisoned TLP was received.
11:5	0h RO	Reserved.
4	0h RWC/P	**Data Link Protocol Error Status (DLPE):** Indicates a data link protocol error occurred.
3:1	0h RO	Reserved.
0	0h RO	**Training Error Status (TE):** Not supported.

36.2.8 Uncorrectable Error Mask (UEM)—Offset 2088h

When set, the corresponding error in the UES register is masked, and the logged error will cause no action. When cleared, the corresponding error is enabled.
These registers are reset by core PWROK.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD											URE	EE	MT	RO	UC	CM	CT	FCPE	PT	RSVD								DLPE	RSVD			TE

Bit Range	Default and Access	Field Name (ID): Description
31:21	0h RO	Reserved.
20	0h RW/P	**Unsupported Request Error Mask (URE):** Mask for uncorrectable errors.
19	0h RO	**ECRC Error Mask (EE):** ECRC is not supported.
18	0h RW/P	**Malformed TLP Mask (MT):** Mask for malformed TLPs.
17	0h RW/P	**Receiver Overflow Mask (RO):** Mask for receiver overflows.
16	0h RO	**Unexpected Completion Mask (UC):** Reserved, Not supported.
15	0h RW/P	**Completer Abort Mask (CM):** Mask for completer abort.
14	0h RO	**Completion Timeout Mask (CT):** Reserved, not supported.
13	0h RO	**Flow Control Protocol Error Mask (FCPE):** Not supported.
12	0h RW/P	**Poisoned TLP Mask (PT):** Mask for poisoned TLPs.
11:5	0h RO	Reserved.
4	0h RW/P	**Data Link Protocol Error Mask (DLPE):** Mask for data link protocol errors.
3:1	0h RO	Reserved.
0	0h RO	**Training Error Mask (TE):** Not supported.

36.2.9 Uncorrectable Error Severity (UEV)—Offset 208Ch

These registers are reset by core PWROK

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD											URE	EE	MT	RO	UC	CA	CT	FCPE	PT	RSVD								DLPE	RSVD			TE

Bit Range	Default and Access	Field Name (ID): Description
31:21	0h RO	Reserved.
20	0h RW/P	**Unsupported Request Error Severity (URE):** Severity for unsupported request reception.
19	0h RO	**ECRC Error Severity (EE):** Not Supported.
18	0h RW/P	**Malformed TLP Severity (MT):** Severity for malformed TLP reception.
17	0h RW/P	**Receiver Overflow Severity (RO):** Severity for receiver overflow occurrences.
16	0h RO	**Unexpected Completion Severity (UC):** Reserved, Not supported.
15	0h RW/P	**Completer Abort Severity (CA):** Severity for completer.
14	0h RO	**Completion Timeout Severity (CT):** Reserved, Not supported.
13	0h RO	**Flow Control Protocol Error Severity (FCPE):** Not supported.
12	0h RW/P	**Poisoned TLP Severity (PT):** Severity for poisoned TLP reception.
11:5	0h RO	Reserved.
4	0h RW/P	**Data Link Protocol Error Severity (DLPE):** Severity for data link protocol errors.
3:1	0h RO	Reserved.
0	0h RW/P	**Training Error Severity (TE):** TE not supported. This bit is RW for ease of implementation.

36.2.10 Correctable Error Status (CES)—Offset 2090h

These registers are reset by core PWROK.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																		ANFES	RTT	RSVD			RNR	BD	BT	RSVD					RE	

Bit Range	Default and Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13	0h RWC/P	**Advisory Non-Fatal Error Status (ANFES):** When set, indicates that a Advisory Non-Fatal Error occurred.
12	0h RWC/P	**Replay Timer Timeout Status (RTT):** Indicates the replay timer timed out.
11:9	0h RO	Reserved.
8	0h RWC/P	**Replay Number Rollover Status (RNR):** Indicates the replay number rolled over.
7	0h RWC/P	**Bad DLLP Status (BD):** Indicates a bad DLLP was received.
6	0h RWC/P	**Bad TLP Status (BT):** Indicates a bad TLP was received.
5:1	0h RO	Reserved.
0	0h RWC/P	**Receiver Error Status (RE):** Indicates a receiver error occurred.

36.2.11 Correctable Error Mask (CEM)—Offset 2094h

When set, the corresponding error in the CES register is masked, and the logged error will cause no action. When cleared, the corresponding error is enabled.
These registers are reset by core PWROK.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 2000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																		ANFEM	RTT	RSVD			RNR	BD	BT	RSVD					RE

Bit Range	Default and Access	Field Name (ID): Description
31:14	0h RO	Reserved.
13	1h RW/P	**Advisory Non-Fatal Error Mask (ANFEM):** When set, masks Advisory Non-Fatal errors from (a) signaling ERR_COR to the device control register and (b) updating the Uncorrectable Error Status register. This register is set by default to enable compatibility with software that does not comprehend Role-Based Error Reporting.
12	0h RW/P	**Replay Timer Timeout Mask (RTT):** Mask for replay timer timeout.
11:9	0h RO	Reserved.
8	0h RW/P	**Replay Number Rollover Mask (RNR):** Mask for replay number rollover.
7	0h RW/P	**Bad DLLP Mask (BD):** Mask for bad DLLP reception.
6	0h RW/P	**Bad TLP Mask (BT):** Mask for bad TLP reception.
5:1	0h RO	Reserved.
0	0h RW/P	**Receiver Error Mask (RE):** Mask for receiver errors.

36.2.12 Root Error Command (REC)—Offset 20ACh

In an exposed AER capability, this register allows errors to generate interrupts. For this implementation, and for RCRBs in general, interrupts cannot be generated, so this register is reserved.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DPDP	UTPB	RSVD																													

Bit Range	Default and Access	Field Name (ID): Description
31	0h RW	**Drop Poisoned Downstream Packets (DPDP):** When set to a '1': if downstream packet on DMI is received with the EP bit set, this packet and all subsequent packets with data received on DMI for any VC will have their Unsupported Transaction (UT) field set causing them to be forwarded to the Error Handler. When cleared to a '0', downstream packets from DMI with the EP bit set are forwarded onto the downstream backbone normally.
30	0h RW	**Unsupported Transaction Policy Bit (UTPB):** When set to 1, the Unsupported Transactions detected on DMI will not set the UES.URE bit. This subsequently ensures that SERR will never be signaled in response to Unsupported Transactions regardless of UEV.URE. When set to 0, the Unsupported Transactions detected on DMI will set the UES.URE bit.
29:0	0h RO	Reserved.

36.2.13 Root Error Status (RES)—Offset 20B0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																													ENR	RSVD	CR

Bit Range	Default and Access	Field Name (ID): Description
31:3	0h RO	Reserved.
2	0h RWC	**ERR_FATAL/NONFATAL Received (ENR):** Set when either a fatal or a non-fatal error message is received or an internal fatal error is detected (all internal uncorrectable errors are fatal).
1	0h RO	Reserved.
0	0h RWC	**ERR_COR Received (CR):** Set when a correctable error message is received or an internal correctable error is detected.

36.2.14 Error Source Identification (ESID)—Offset 20B4h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
EFNFSID																ECSID															

Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO/V	**ERR_FATAL/NONFATAL Source Identification (EFNFSID):** Loaded with the requester ID indicated in the received ERR_FATAL or ERR_NONFATAL message when RES.ENR is first set, or the internal requestor ID if an internally detected error
15:0	0h RO/V	**ERR_COR Source Identification (ECSID):** Loaded with the requester ID indicated in the received ERR_COR message when RES.CR is first set, or the internal requester ID if an internally detected error

36.2.15 Link Capabilities (LCAP)—Offset 21A4h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 12C40h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	1	1	0	0	0	1	0	0	0	0	0	0
PN								RSVD							EL1		EL0		APMS		MLW						MLS				

Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	**Port Number (PN):** Indicates the port number for the DMI is 0.
23:18	0h RO	Reserved.
17:15	2h RWO	**L1 Exit Latency (EL1):** Indicates that the exit latency is 2s to 4s 000b - Less than 1 s 001b - 1 s to less than 2 s 010b - 2 s to less than 4 s 011b - 4 s to less than 8 s 100b - 8 s to less than 16 s 101b - 16 s to less than 32 s 110b - 32 s to 64 s 111b - More than 64 s
14:12	2h RW	**L0s Exit Latency (EL0):** This field is RW and updatable by BIOS. When BIOS sets this field, it must also update DMI's L0s Control.NFTS field.
11:10	3h RW	**Active State Link PM Support (APMS):** Indicates the level of ASPM support on DMI. 00: No ASPM Support 01: reserved 10: L1 Supported 11: Reserved
9:4	4h RO	**Maximum Link Width (MLW):** Indicates the maximum link width is 4 lanes.
3:0	0h RO/F	**Max Link Speed (MLS):** This field indicates the maximum Link speed of the associated Port. 0001b: 2.5 GT/s is supported. 0010b: 5.0 GT/s, is supported. 0011: 8.0 GT/s is supported

36.2.16 Link Control and Link Status (LCTL_LSTS)—Offset 21A8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 410000h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD							NLW					CLS				RSVD								ES	RSVD						ASPM	

Bit Range	Default and Access	Field Name (ID): Description
31:26	0h RO	Reserved.
25:20	4h RO/V	**Negotiated Link Width (NLW):** Negotiated link width Valid Encodings: 00_0001b: x1 00_0010b: x2 00_0100b: x4
19:16	1h RO/V	**Current Link Speed (CLS):** This field indicates the negotiated Link speed of the given link. 0001b = 2.5 GT/s 0010b = 5.0 GT/s 0010b = 8.0 GT/s
15:8	0h RO	Reserved.
7	0h RW	**Extended Synch (ES):** When set, forces extended transmission of FTS ordered sets when exiting L0s prior to entering L0 and extra TS1 sequences at exit from L1 prior to entering L0.
6:2	0h RO	Reserved.
1:0	0h RW	**Active State Link PM Control (ASPM):** Indicates whether DMI should enter L0s or L1 or both. 00 Disabled 01 reserved 10 L1 Entry Enabled 11 reserved

36.2.17 Link Capabilities 2 (LCAP2)—Offset 21ACh

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																									SLSV						RSVD

Bit Range	Default and Access	Field Name (ID): Description
31:8	0h RO	Reserved.
7:1	0h RO/F	**Support Link Speed Vector (SLSV):** This field indicates the supported Link speed of the associated Port. For each bit, a value of 1b indicates that the corresponding Link speed is supported, otherwise, the Link speed is not supported. Bit definitions within this field are: Bit 0: 2.5 GT/s. Bit 1: 5.0 GT/s. Bit 2: 8.0 GT/s. Bits 6:3: Reserved. This field reports a value of 0000001b if GEN1 data rate is supported but both GEN2 and GEN3 data rate support are not. This field reports a value of 0000011b if both GEN1 and GEN2 data rate are supported but GEN3 data rate support is not. Otherwise, this register reports 0000111b
0	0h RO	Reserved.

36.2.18 Link Control 2 and Link Status 2 (LCTL2_LSTS2)—Offset 21B0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 1h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
RSVD																													TLS			

Bit Range	Default and Access	Field Name (ID): Description
31:4	0h RO	Reserved.
3:0	1h RW/F	**Target Link Speed (TLS):** This field sets an upper limit on Link operational speed by restricting the values advertised by the upstream component in its training sequences. The encoded value specifies a bit location in the Supported Link Speeds Vector (in the Link Capabilities 2 register) that corresponds to the current Link speed. Defined encodings are: 0001b: Supported Link Speeds Vector field bit 0. 0010b: Supported Link Speeds Vector field bit 1. 0011b: Supported Link Speeds Vector field bit 2. 0100b: Supported Link Speeds Vector field bit 3. 0101b: Supported Link Speeds Vector field bit 4. 0110b: Supported Link Speeds Vector field bit 5. 0111b: Supported Link Speeds Vector field bit 6. All other encodings are reserved. If a value is written to this field that does not correspond to a supported speed, as indicated by the Supported Link Speeds Vector, the result is undefined. The default value of this field is 2.5 GT/s for survivability reason. ***Note:*** This register field could be used by REUT software to limit the link speed to 2.5 GT/s or 5 GT/s data rate. Register Attribute: Dynamic.

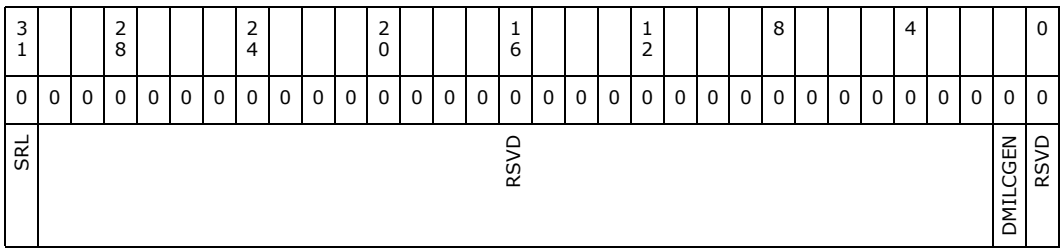
36.2.19 DMI Control Register (DMIC)—Offset 2234h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RWO	**Secured Register Lock (SRL):** When this bit is set, all the secured registers will be locked and will be Read-Only. The following fields are locked by DMIC.SRL: V0CTL.ETVM and V0CTL.TVM. V1CTL.ETVM and V1CTL.TVM.
30:2	0h RO	Reserved.
1	0h RW	**DMI Link Dynamic Clock Gate Enable (DMILCGEN):** When set, this bit enables dynamic clock gating on the DMI Link clock domain logic. When cleared, dynamic clock gating on the DMI Link clock domain is disabled.
0	0h RO	Reserved.

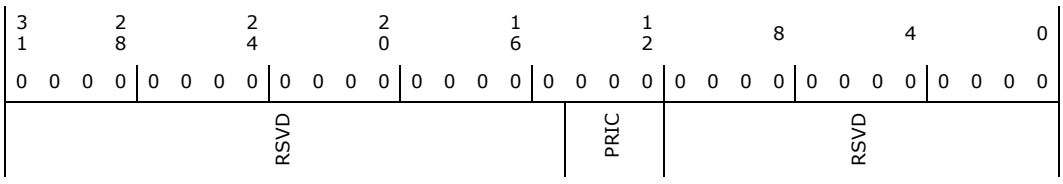
36.2.20 IOSF Primary Control And Status (IPCS_IOSFSBCS)—Offset 223Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:15	0h RO	Reserved.
14:12	0h RW	**IOSF Primary ISM Idle Counter (PRIC):** BIOS may need to program this register field.
11:0	0h RO	Reserved.

36.2.21 DMI Port Link Control (DMILINKC)—Offset 2304h

The BIOS may need to program this register.

36.2.22 DMI Configuration (DMICFG)—Offset 2310h

The BIOS may need to program this register.

36.2.23 DMI PLL Shutdown (DMIPLLDOWN)—Offset 2320h

The BIOS may need to program this register.

36.2.24 DMI Power Management Control (DMIPMCTL)—Offset 2334h

The BIOS may need to program this register.

36.2.25 DMI Additional Link Control (DMIALC)—Offset 2338h

The BIOS may need to program this register.

36.2.26 DMI NFTS (DMINFTS)—Offset 2340h

The BIOS may need to program this register.

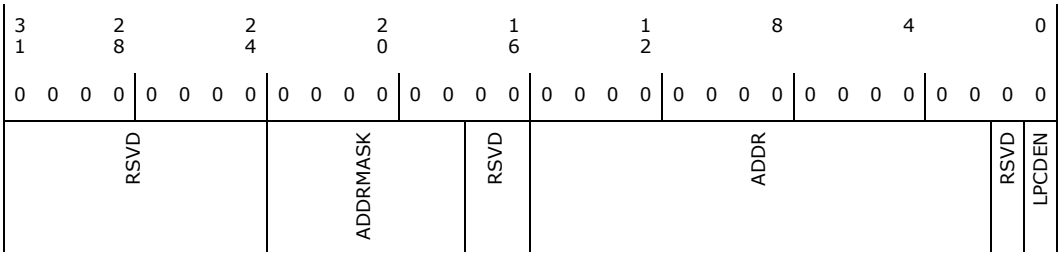
36.2.27 LPC Generic I/O Range 1 (LPCLGIR1)—Offset 2730h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23:18	0h RW/L	**Address [7:2] Mask (ADDRMASK):** A 1 in any bit position indicates that any value in the corresponding address bit in a received cycle will be treated as a match. The corresponding bit in the Address field, below, is ignored. The mask is only provided for the lower 6 bits of the DWord address, allowing for decoding blocks up to 256 bytes in size. This register is Read-Only if the DMIC.SRL field is set. Register Attribute: Static.
17:16	0h RO	Reserved.
15:2	0h RW/L	**Address [15:2] (ADDR):** DWord-aligned address. Note that PCH does not provide decode down to the word or byte level. This register is Read-Only if the DMIC.SRL field is set. Register Attribute: Static.
1	0h RO	Reserved.
0	0h RW/L	**LPC Decode Enable (LPCDEN):** When this bit is set to 1, then the range specified in this register is enabled for decoding to LPC. This register is Read-Only if the DMIC.SRL field is set. Register Attribute: Static.

36.2.28 LPC Generic I/O Range 2 (LPCLGIR2)—Offset 2734h

Same description as LPCLGIR1 register.

36.2.29 LPC Generic I/O Range 3 (LPCLGIR3)—Offset 2738h

Same description as LPCLGIR1 register.

36.2.30 LPC Generic I/O Range 4 (LPCLGIR4)—Offset 273Ch

Same description as LPCLGIR1 register.

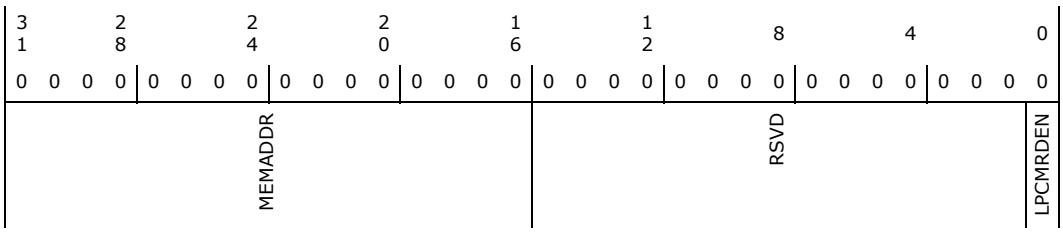
36.2.31 LPC Generic Memory Range (LPCGMR)—Offset 2740h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RW/L	**Memory Address (MEMADDR):** This field specifies a 64KB memory block anywhere in the 4GB memory space that will be decoded to LPC as standard LPC Memory Cycle if enabled. This register is Read-Only if the DMIC.SRL field is set. Register Attribute: Static.
15:1	0h RO	Reserved.
0	0h RW/L	**LPC Memory Range Decode Enable (LPCMRDEN):** When this bit is set to 1, then the range specified in this register is enabled for decoding to LPC. This register is Read-Only if the DMIC.SRL field is set. Register Attribute: Static.

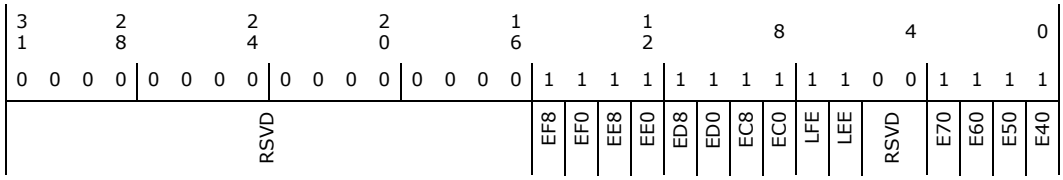
36.2.32 LPC BIOS Decode Enable (LPCBDE)—Offset 2744h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: FFCFh



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15	1h RO	**F8-FF Enable (EF8):** Enables decoding of 512K of the following BIOS range: Data space: FFF80000h – FFFFFFFFh Feature space: FFB80000h – FFBFFFFFh Register Attribute: Static.
14	1h RW/L	**F0-F8 Enable (EF0):** Enables decoding of 512K of the following BIOS range: Data space: FFF00000h – FFF7FFFFh Feature space: FFB00000h – FFB7FFFFh This register is Read-Only if the DMIC.SRL field is set. Register Attribute: Static.
13	1h RW/L	**E8-EF Enable (EE8):** Enables decoding of 512K of the following BIOS range: Data space: FFE80000h – FFEFFFFFh Feature space: FFA80000h – FFAFFFFFh This register is Read-Only if the DMIC.SRL field is set. Register Attribute: Static.
12	1h RW/L	**E0-E8 Enable (EE0):** Enables decoding of 512K of the following BIOS range: Data space: FFE00000h – FFE7FFFFh Feature Space: FFA00000h – FFA7FFFFh This register is Read-Only if the DMIC.SRL field is set. Register Attribute: Static.
11	1h RW/L	**D8-DF Enable (ED8):** Enables decoding of 512K of the following BIOS range: Data space: FFD80000h – FFDFFFFFh Feature space: FF980000h – FF9FFFFFh This register is Read-Only if the DMIC.SRL field is set. Register Attribute: Static.

Bit Range	Default and Access	Field Name (ID): Description
10	1h RW/L	**D0-D7 Enable (ED0):** Enables decoding of 512K of the following BIOS range: Data space: FFD00000h – FFD7FFFFh Feature space: FF900000h – FF97FFFFh This register is Read-Only if the DMIC.SRL field is set. Register Attribute: Static.
9	1h RW/L	**C8-CF Enable (EC8):** Enables decoding of 512K of the following BIOS range: Data space: FFC80000h – FFCFFFFFh Feature space: FF880000h – FF8FFFFFh This register is Read-Only if the DMIC.SRL field is set. Register Attribute: Static.
8	1h RW/L	**C0-C7 Enable (EC0):** Enables decoding of 512K of the following BIOS range: Data space: FFC00000h – FFC7FFFFh Feature space: FF800000h – FF87FFFFh This register is Read-Only if the DMIC.SRL field is set. Register Attribute: Static.
7	1h RW/L	**Legacy F Segment Enable (LFE):** This enables the decoding of the legacy 64KB range at F0000h – FFFFFh This register is Read-Only if the DMIC.SRL field is set. Register Attribute: Static.
6	1h RW/L	**Legacy E Segment Enable (LEE):** This enables the decoding of the legacy 64KB range at E0000h – EFFFFh This register is Read-Only if the DMIC.SRL field is set. Register Attribute: Static.
5:4	0h RO	Reserved.
3	1h RW/L	**70-7F Enable (E70):** Enables decoding of 1MB of the following BIOS range: Data space: FF700000h – FF7FFFFFh Feature space: FF300000h – FF3FFFFFh This register is Read-Only if the DMIC.SRL field is set. Register Attribute: Static.
2	1h RW/L	**60-6F Enable (E60):** Enables decoding of 1MB of the following BIOS range: Data space: FF600000h – FF6FFFFFh Feature Space: FF200000h – FF2FFFFFh This register is Read-Only if the DMIC.SRL field is set. Register Attribute: Static.
1	1h RW/L	**50-5F Enable (E50):** Enables decoding of 1MB of the following BIOS range: Data space: FF500000h – FF5FFFFFh Feature Space: FF100000h – FF1FFFFFh This register is Read-Only if the DMIC.SRL field is set. Register Attribute: Static.
0	1h RW/L	**40-4F Enable (E40):** Enables decoding of 1MB of the following BIOS range: Data space: FF400000h – FF4FFFFFh Feature space: FF000000h – FF0FFFFFh This register is Read-Only if the DMIC.SRL field is set. Register Attribute: Static.

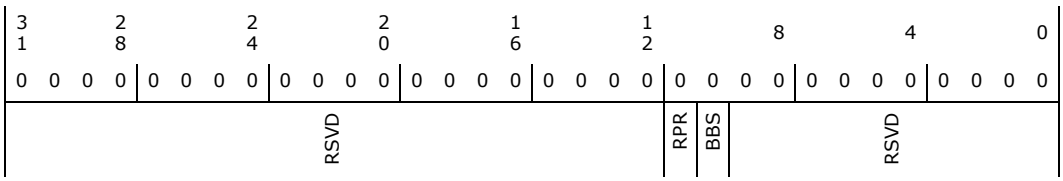
36.2.33 General Control and Status (GCS)—Offset 274Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:12	0h RO	Reserved.
11	0h RW/L	**Reserved Page Route (RPR):** Determines where to send the reserved page registers. These addresses are sent to PCIe Root Port or LPC/eSPI for the purpose of generating POST codes. The I/O addresses modified by this field are: 80h - 8Fh. When cleared, DMI will not perform source decode on the I/O ranges specified above. The cycles hitting these ranges will end up in P2SB which will then forward the cycle to LPC or eSPI through IOSF Sideband. When set, access to the I/O ranges specified above will be forwarded to PCIe Root Port with the destination ID specified in GCS.RPRDID using DMI source decode. The aliases for these registers, at 90h, 94h, 95h, 96h, 98h, 9Ch, 9Dh, and 9Eh, are never source decoded by DMI. This register is Read-Only if the DMIC.SRL field is set. Register Attribute: Static.
10	0h RW/L/S	**Boot BIOS Strap (BBS):** This field determines the destination of accesses to the BIOS memory range. Bits Description 0 SPI 1 LPC/eSPI When SPI or LPC/eSPI is selected, the range that is decoded is further qualified by other configuration bits described in the respective sections. The value in this field can be overwritten by software as long as the BIOS Interface Lock-Down (bit 0) and DMIC.SRL are not set. Register Attribute: Static.
9:0	0h RO	Reserved.

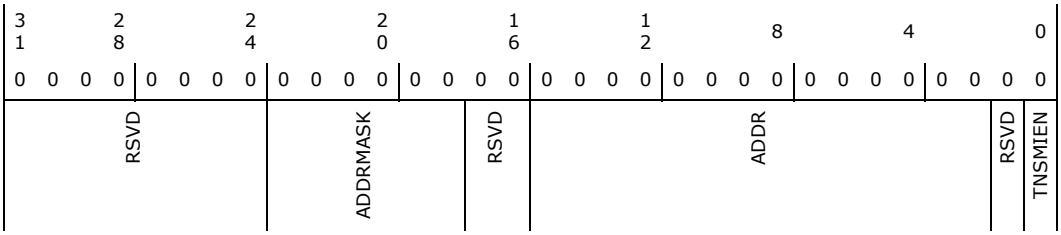
36.2.34 I/O Trap Register 1 low (IOT1_LOW)—Offset 2750h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23:18	0h RW	**Address Mask (ADDRMASK):** A 1 in any bit position indicates that any value in the corresponding address bit in a received cycle will be treated as a match. The corresponding bit in the Address field, below, is ignored. The mask is only provided for the lower 6 bits of the DWord address, allowing for traps on address ranges up to 256 bytes in size.
17:16	0h RO	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
15:2	0h RW	**Address (ADDR):** DWord-aligned address.
1	0h RO	Reserved.
0	0h RW	**Trap and SMI# Enable (TNSMIEN):** When this bit is set to "1?, then the trapping logic specified in this register is enabled.

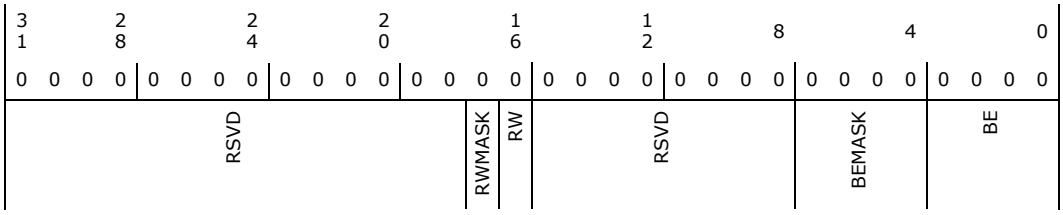
36.2.35 I/O Trap Register 1 high (IOT1_HIGH)—Offset 2754h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:18	0h RO	Reserved.
17	0h RW	**Read/Write Mask (RWMASK):** When this bit is 1, the trapping logic will operate on both read and write cycles. When this bit is 0, the cycle must match the type specified in bit 16.
16	0h RW	**Read/Write (RW):** 1 = Read 0 = Write The value in this field does not matter if bit 17 is set.
15:8	0h RO	Reserved.
7:4	0h RW	**Byte Enable Mask (BEMASK):** A 1 in any bit position indicates that any value in the corresponding byte enable bit in a received cycle will be treated as a match. The corresponding bit in the Byte Enables field, below, is ignored.
3:0	0h RW	**Byte Enables (BE):** Active-high, DWord-aligned byte enables.

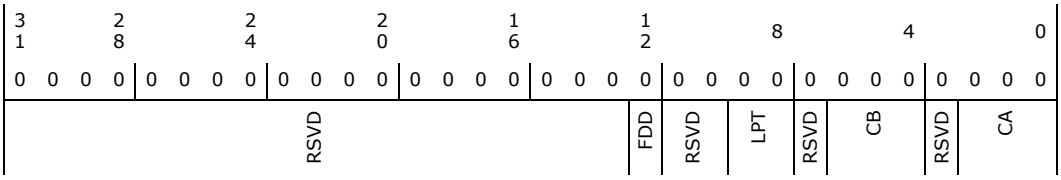
36.2.36 LPC I/O Decode Range (LPCIOD)—Offset 2770h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:13	0h RO	Reserved.
12	0h RW/L	**FDD Range (FDD)**
11:10	0h RO	Reserved.
9:8	0h RW/L	**LPT Range (LPT):** The following table describes which range to decode for the LPT Port. Bits Decode Range 00 378h - 37Fh and 778h – 77Fh 01 278h – 27Fh (port 279h is read only) and 678h – 67Fh 10 3BCh – 3BEh and 7BCh – 7BEh 11 Reserved This register is Read-Only if the DMIC.SRL field is set.
7	0h RO	Reserved.
6:4	0h RW/L	**ComB Range (CB):** The following table describes which range to decode for the COMB Port. Bits Decode Range 000 3F8h – 3FFh (COM1) 001 2F8h – 2FFh (COM2) 010 220h – 227h 011 228h – 22Fh 100 238h – 23Fh 101 2E8h – 2EFh (COM 4) 110 338h – 33Fh 111 3E8h – 3EFh (COM 3) This register is Read-Only if the DMIC.SRL field is set.
3	0h RO	Reserved.
2:0	0h RW/L	**ComA Range (CA):** The following table describes which range to decode for the COMA Port. Bits Decode Range 000 3F8h – 3FFh (COM1) 001 2F8h – 2FFh (COM2) 010 220h – 227h 011 228h – 22Fh 100 238h – 23Fh 101 2E8h – 2EFh (COM 4) 110 338h – 33Fh 111 3E8h – 3EFh (COM 3) This register is Read-Only if the DMIC.SRL field is set.

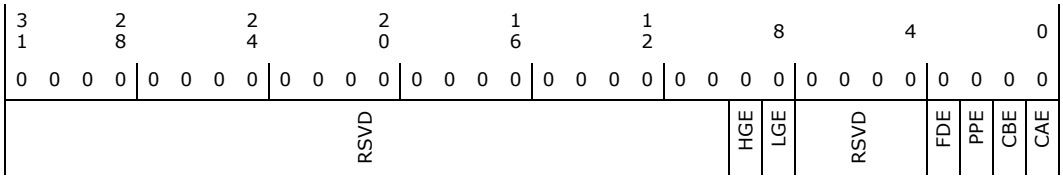
36.2.37 LPC I/O Enable (LPCIOE)—Offset 2774h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:10	0h RO	Reserved.
9	0h RW/L	**High Gameport Enable (HGE):** Enables decoding of the I/O locations 208h to 20Fh to LPC. This register is Read-Only if the DMIC.SRL field is set.
8	0h RW/L	**Low Gameport Enable (LGE):** Enables decoding of the I/O locations 200h to 207h to LPC. This register is Read-Only if the DMIC.SRL field is set.
7:4	0h RO	Reserved.
3	0h RW/L	**Floppy Drive Enable (FDE):** Enables decoding of the FDD range to LPC. Range is selected by LIOD.FDE. This register is Read-Only if the DMIC.SRL field is set.
2	0h RW/L	**Parallel Port Enable (PPE):** Enables decoding of the LPT range to LPC. Range is selected by LIOD.LPT. This register is Read-Only if the DMIC.SRL field is set.
1	0h RW/L	**Com Port B Enable (CBE):** Enables decoding of the COMB range to LPC. Range is selected by LIOD.CB. This register is Read-Only if the DMIC.SRL field is set.
0	0h RW/L	**Com Port A Enable (CAE):** Enables decoding of the COMA range to LPC. Range is selected by LIOD.CA. This register is Read-Only if the DMIC.SRL field is set.

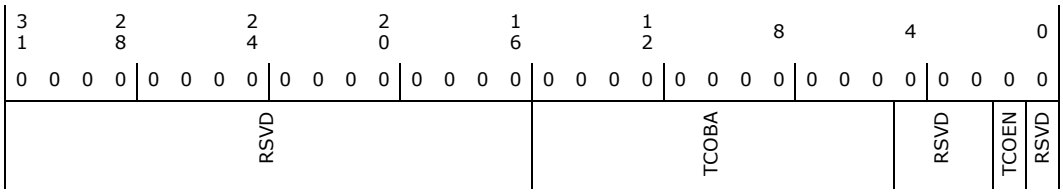
36.2.38 TCO Base Address (TCOBASE)—Offset 2778h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.
15:5	0h RW/L	**TCO Base Address (TCOBA):** Provides the 32 bytes of I/O space for TCO logic, that can be map anywhere in the 64k I/O space on 32-byte boundaries. This register is Read-Only if the DMIC.SRL field is set.
4:2	0h RO	Reserved.
1	0h RW/L	**TCO Enable (TCOEN):** When set, decode of the I/O range specified by the TCO base address. This register is Read-Only if the DMIC.SRL field is set.
0	0h RO	Reserved.

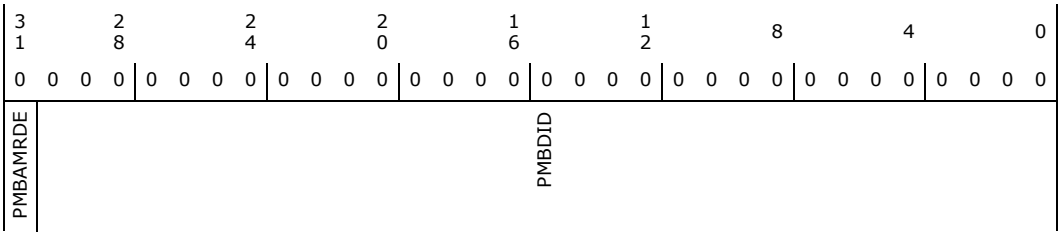
36.2.39 PM Base Control (PMBASEC)—Offset 27B0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31	0h RW/L	**PM Base Address Memory Range Decode Enable (PMBAMRDE):** When enabled, memory cycles that falls within the PMBASEADDR.PMBAMRB and PMBASEADDR.PMBAMRL range inclusive will be forwarded using source decode to the destination ID specified in PMBASEC.PMBDID field. This register is Read-Only if the DMIC.SRL field is set.
30:0	0h RW/L	**PM Base Destination ID (PMBDID):** The destination ID to be used to forward the cycle decoded to hit the PM Base Address range. BIOS must program the bits which are not used to zeros. Hardware will ignore the unused bits. This register is Read-Only if the DMIC.SRL field is set.

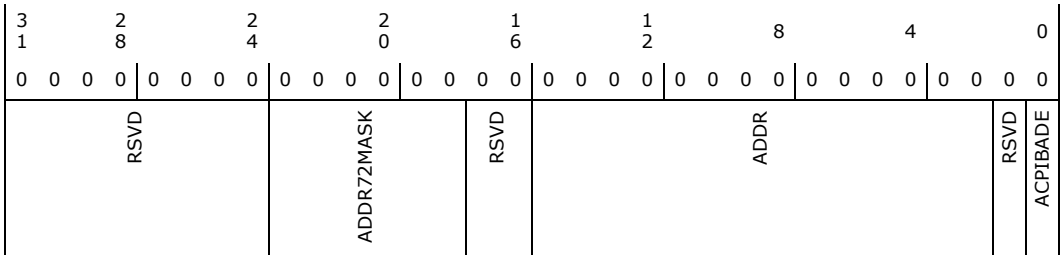
36.2.40 ACPI Base Address (ACPIBA)—Offset 27B4h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23:18	0h RW/L	**Address[7:2] Mask (ADDR72MASK):** A 1 in any bit position indicates that any value in the corresponding address bit in a received cycle will be treated as a match. The corresponding bit in the Address field, below, is ignored. The mask is only provided for the lower 6 bits of the DWord address, allowing for decoding blocks up to 256 bytes in size. This register is Read-Only if the DMIC.SRL field is set.
17:16	0h RO	Reserved.
15:2	0h RW/L	**Address[15:2] (ADDR):** DWord-aligned address. Note that PCH does not provide decode down to the word or byte level. This register is Read-Only if the DMIC.SRL field is set.
1	0h RO	Reserved.
0	0h RW/L	**ACPI Base Address Decode Enable (ACPIBADE):** When this bit is set to 1, then the range specified in this register is enabled for decoding to the destination ID specified in ACPIBDID register. This register is Read-Only if the DMIC.SRL field is set.

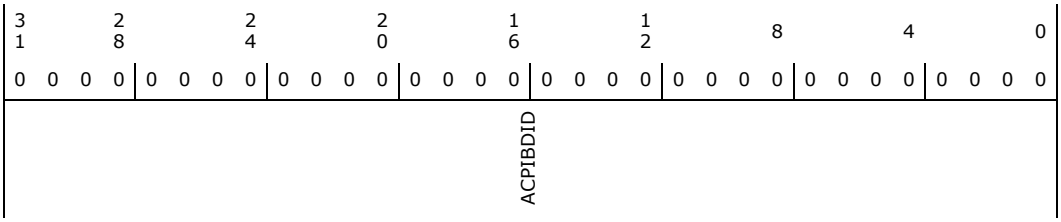
36.2.41 ACPI Base Destination ID (ACPIBDID)—Offset 27B8h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/L	**ACPI Base Destination ID (ACPIBDID):** The destination ID to be used to forward the cycle decoded to hit the ACPI Base Address range. BIOS must program the bits which are not used to zeros. Hardware will ignore the unused bits. This register is Read-Only if the DMIC.SRL field is set.

36.3 DCI Configuration Registers Summary

These registers are within the PCH Private Configuration Space which is accessible through the PCH Sideband Interface. They can be accessed via (SBREG_BAR + PortID + Register Offset).

Table 36-3. Summary of DCI Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
4h	7h	DCI Control Register (ECTRL) - Offset 4h	0h

36.3.1 DCI Control Register (ECTRL)—Offset 4h

These registers are within the PCH Private Configuration Space which is accessible through the PCH Sideband Interface. They can be accessed via (SBREG_BAR + PortID + Register Offset).

Table 36-4. Summary of DCI PCR Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
4h	7h	DCI Control Register (ECTRL)—Offset 4h	0h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																							DCISTS	RSVD			HDCIEN	RSVD			

Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	**Reserved**.
8	0b RO	**DCI Available Status (DCISTS)** 0 = DCI is not available 1 = DCI is available
7:5	000b RO	**Reserved**.
4	0h RW/L	**Host DCI Enable (HDCIEN)** 0 = Disable DCI 1 = Enable DCI This bit resides in the RTC Well and is only reset by RTCRST#. This bit is cleared by writing a '0' to it. Writing a '1' has no effect.
3:0	0h RO	**Reserved**.

36.4 PSF1 PCR Registers Summary

These registers are within the PCH Private Configuration Space which is accessible through the PCH Sideband Interface. They can be accessed via (SBREG_BAR + PortID + Register Offset).

Table 36-5. Summary of PSF1 PCR Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
31Ch	31Fh	D22:F4 Function Disable (PSF_1_AGNT_T0_SHDW_PCIEN_CSE_RS0_D22_F4)—Offset 31Ch	0h
338h	33Bh	D22:F4 PCI Configuration Disable (PSF_1_AGNT_T0_SHDW_CFG_DIS_CSE_RS0_D22_F3)—Offset 338h	0h
41Ch	41Fh	D22:F1 Function Disable (PSF_1_AGNT_T0_SHDW_PCIEN_CSE_RS0_D22_F1)—Offset 41Ch	0h
438h	43B	D22:F1 PCI Configuration Disable (PSF_1_AGNT_T0_SHDW_CFG_DIS_CSE_RS0_D22_F1)—Offset 438h	0h
61Ch	61Fh	D22:F0 Function Disable (PSF_1_AGNT_T0_SHDW_PCIEN_CSE_RS0_D22_F0)—Offset 61Ch	0h
638h	63Bh	D22:F0 PCI Configuration Disable (PSF_1_AGNT_T0_SHDW_CFG_DIS_CSE_RS0_D22_F0)—Offset 438h	0h
81Ch	81Fh	D22:F2 Function Disable (PSF_1_AGNT_T0_SHDW_PCIEN_PTIO_RS0_D22_F2)—Offset 81Ch	0h
838h	83Bh	D22:F2 PCI Configuration Disable (PSF_1_AGNT_T0_SHDW_CFG_DIS_CSE_RS0_D22_F2)—Offset 838h	0h
B00	B03h	D23:F0 Sata PCI Base Address (PSF_1_AGNT_T0_SHDW_BAR0_VR_RS0_D23_F0	0h
B1Ch	B1Fh	D23:F0 Function Disable (PSF_1_AGNT_T0_SHDW_PCIEN_PTIO_RS0_D23_F0)	0h
C00h	C03h	D17:F5 sSata PCI Base Address (PSF_1_AGNT_T0_SHDW_BAR0_VR_RS0_D17_F5	0h
C1Ch	C1Fh	D17:F5 Function Disable (PSF_1_AGNT_T0_SHDW_PCIEN_PTIO_RS0_D17_F5	0h
4014h	4017h	Rootspace Configuration (PSF_1_ROOTSPACE_CONFIG_RS0)—Offset 4014h	0h
4054h	4057h	Multicast Control Register (PSF_1_PSF_MC_CONTROL_MCAST0_RS0_EOI)—Offset 4054h	0h
406Ch	406Fh	Destination ID (PSF_1_PSF_MC_AGENT_MCAST0_RS0_TGT0_EOI)—Offset 406Ch	0h

36.4.1 D22:F4 Function Disable (PSF_1_AGNT_T0_SHDW_PCIEN_CSE_RS0_D22_F4)—Offset 31Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																							FunDis	RSVD							

Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	Reserved.
8	0h	**Intel(R) MEI #3 Function Disable (FunDis):** When set, transactions targeting this PCI function will not be positively decoded, and cfgWr transactions to this function will not be carried out.
7:0	0h RO	Reserved.

36.4.2 D22:F4 PCI Configuration Disable (PSF_1_AGNT_T0_SHDW_CFG_DIS_CSE_RS0_D22_F3)—Offset 338h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																															CfgDis

Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h	**CfgDis:** If set, the PCI configuration space of this device is switched off, and hidden from the OS.

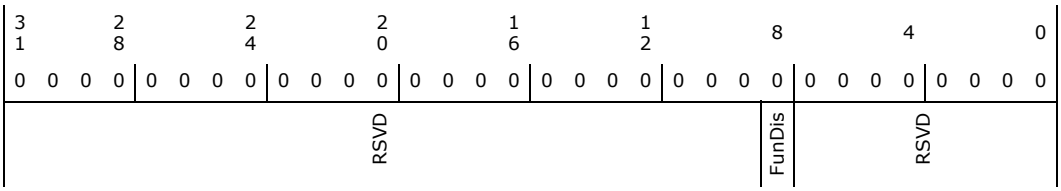
36.4.3 D22:F1 Function Disable (PSF_1_AGNT_T0_SHDW_PCIEN_CSE_RS0_D22_F1)—Offset 41Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	Reserved.
8	0h	**Intel(R) MEI #2 Function Disable (FunDis):** When set, transactions targeting this PCI function will not be positively decoded, and cfgWr transactions to this function will not be carried out.
7:0	0h RO	Reserved.

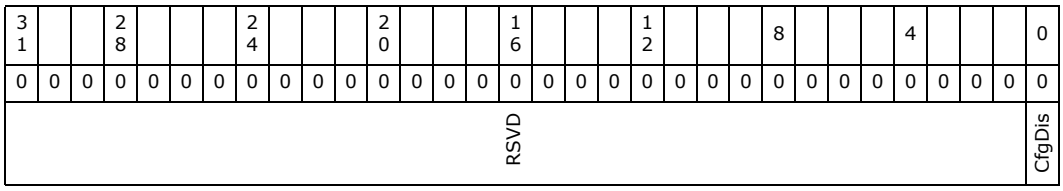
36.4.4 D22:F1 PCI Configuration Disable (PSF_1_AGNT_T0_SHDW_CFG_DIS_CSE_RS0_D22_F1)—Offset 438h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h	**CfgDis:** If set, the PCI configuration space of this device is switched off, and hidden from the OS.

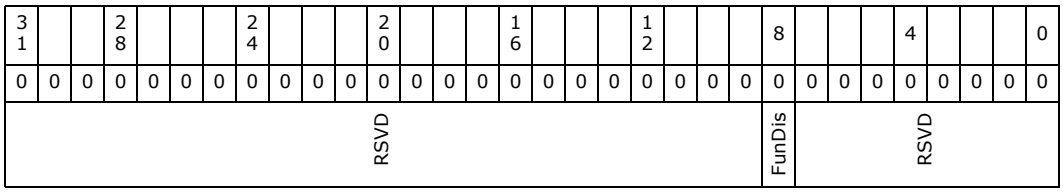
36.4.5 D22:F0 Function Disable (PSF_1_AGNT_T0_SHDW_PCIEN_CSE_RS0_D22_F0)—Offset 61Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	Reserved.
8	0h	**Intel(R) MEI #1 Function Disable (FunDis):** When set, transactions targeting this PCI function will not be positively decoded, and cfgWr transactions to this function will not be carried out.
7:0	0h RO	Reserved.

36.4.6 D22:F0 PCI Configuration Disable (PSF_1_AGNT_T0_SHDW_CFG_DIS_CSE_RS0_D22_F0)—Offset 438h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																															CfgDis

Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h	**CfgDis:** If set, the PCI configuration space of this device is switched off, and hidden from the OS.

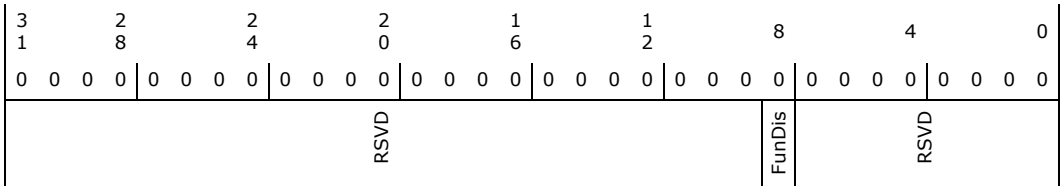
36.4.7 D22:F2 Function Disable (PSF_1_AGNT_T0_SHDW_PCIEN_PTIO_RS0_D22_F2)—Offset 81Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	Reserved.
8	0h	**IDE-R FunDis:** When set, transactions targeting this PCI function will not be positively decoded, and cfgWr transactions to this function will not be carried out.
7:0	0h RO	Reserved.

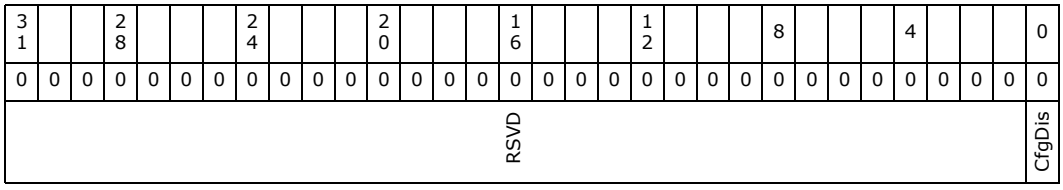
36.4.8 D22:F2 PCI Configuration Disable (PSF_1_AGNT_T0_SHDW_CFG_DIS_CSE_RS0_D22_F2)—Offset 838h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h	**CfgDis:** If set, the PCI configuration space of this device is switched off, and hidden from the OS.

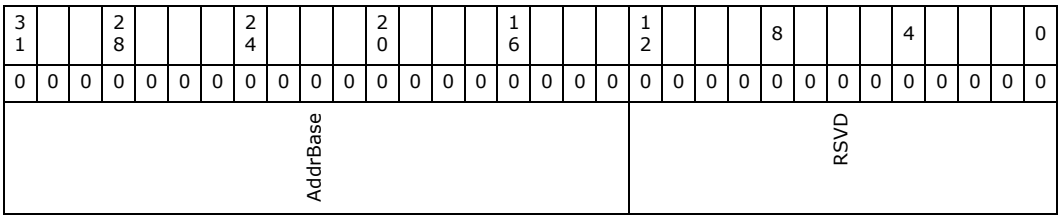
36.4.9 D23:F0 Sata PCI Base Address (PSF_1_AGNT_T0_SHDW_BAR0_VR_RS0_D23_F0)—Offset 0B00h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:13	0h RW	**SATA PCI BAR (AddrBase)**
12:0	0h RO	Reserved.

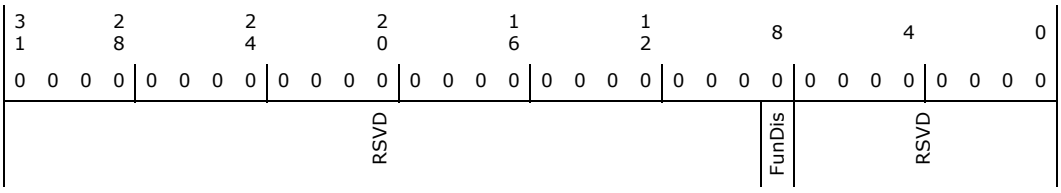
36.4.10 D23:F0 Function Disable (PSF_1_AGNT_T0_SHDW_PCIEN_PTIO_RS0_D23_F0)—Offset B1Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	Reserved.
8	0h	**SATA FunDis:** When set, transactions targeting this PCI function will not be positively decoded, and cfgWr transactions to this function will not be carried out.
7:0	0h RO	Reserved.

36.4.11 D17:F5 sSata PCI Base Address (PSF_1_AGNT_T0_SHDW_BAR0_VR_RS0_D17_F5)—Offset 0C00h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
AddrBase																			RSVD													

Bit Range	Default and Access	Field Name (ID): Description
31:13	0h RW	**SATA PCI BAR (AddrBase)**
12:0	0h RO	Reserved.

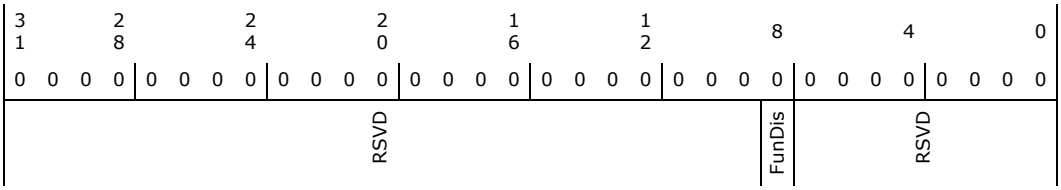
36.4.12 D17:F5 Function Disable (PSF_1_AGNT_T0_SHDW_PCIEN_PTIO_RS0_D17_F5)—Offset C1Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	Reserved.
8	0h	**sSATA FunDis:** When set, transactions targeting this PCI function will not be positively decoded, and cfgWr transactions to this function will not be carried out.
7:0	0h RO	Reserved.

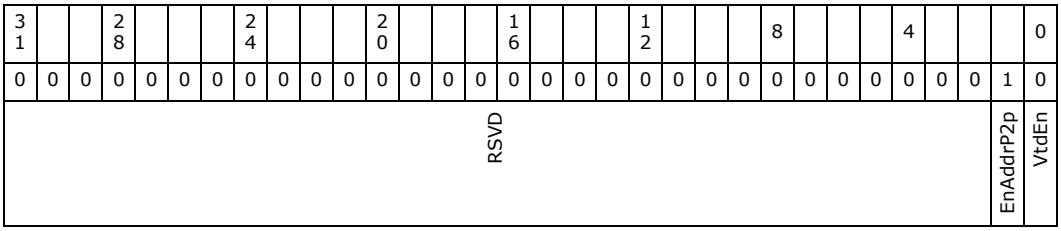
36.4.13 Rootspace Configuration (PSF_1_ROOTSPACE_CONFIG_RS0)—Offset 4014h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 2h



Bit Range	Default and Access	Field Name (ID): Description
31:2	0h RO	Reserved.
1	1h	**EnAddrP2p:** If set, address-based p2p transactions are allowed for this root space.
0	0h RW	**Vtd Enable (VtdEn):** If set Intel Virtualization Technology for Directed I/O is enabled for this root space. This bit should reflect the system-wide VT-D setting, and is typically maintained by BIOS.

36.4.14 Multicast Control Register (PSF_1_PSF_MC_CONTROL_MCAST0_RS0_EOI)—Offset 4054h

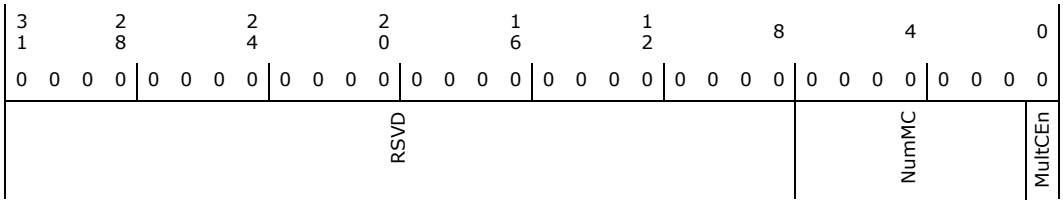
This register contains information for the PSF to be able to do a multicast operation on each root-space for incoming IOSF messages that requires Broadcast from Root Complex. The only multicast messages supported on PSF are EOI and MCTP messages. There will be a copy of this register for EOI and MCTP, and for each root-space.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:8	0h RO	Reserved.
7:1	0h	**Multicast (NumMC):** Number of multicast Agents on this root-space on the PSF segment.
0	0h	**Multicast Enable (MultCEn):** Multicast Enable. If set Multicast transactions are supported on this root-space.

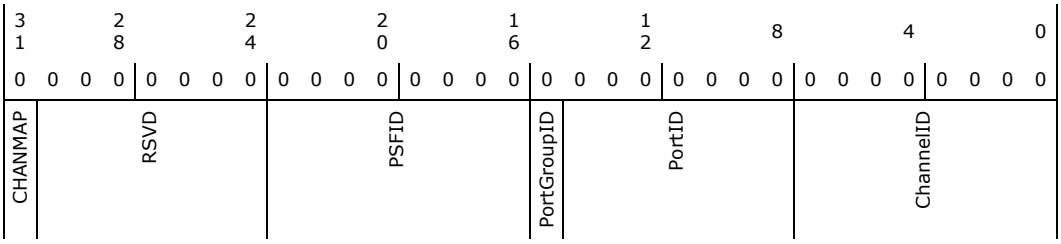
36.4.15 Destination ID (PSF_1_PSF_MC_AGENT_MCAST0_RS0_TGT0_EOI)—Offset 406Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31	0h	**Channel ID Map (CHANMAP):** If this bit is set, the ChannelID field is ignored and is looked up in the CHANMAP register set that belongs to PortGroupID:PortID
30:24	0h RO	Reserved.
23:16	0h	**PSF ID (PSFID):** Default value=0x0,
15	0h	**Port Group ID (PortGroupID):** Default value=0
14:8	0h	**Port ID (PortID):** Default value=0x0, Since 1'b0 indicates source-decode, the port ID needs to be larger than 0
7:0	0h	**Channel ID (ChannelID):** Default value=0x0,

36.5 PSF2 PCR Registers Summary

These registers are within the PCH Private Configuration Space which is accessible through the PCH Sideband Interface. They can be accessed via (SBREG_BAR + PortID + Register Offset).

Table 36-6. Summary of PSF2 PCR registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
11Ch	11Fh	D20:F2 Function Disable (PSF_2_AGNT_T0_SHDW_PCIEN_TRH_RS0_D20_F2_OFFSET1)	00000000h
138h	13Bh	D20:F2 PCI Configuration Disable (PSF_2_AGNT_T0_SHDW_CFG_DIS_CSE_RS0_D20_F2)	00000000h
31Ch	31Fh	D20:F0 Function Disable (PSF_2_AGNT_T0_SHDW_PCIEN_SCC_RS0_D20_F0)	00000000h
338h	33Bh	D20:F0 PCI Configuration Disable (PSF_2_AGNT_T0_SHDW_CFG_DIS_CSE_RS0_D20_F0)	00000000h
4014h	4017h	Rootspace Configuration (PSF_2_ROOTSPACE_CONFIG_RS0)	00000002h

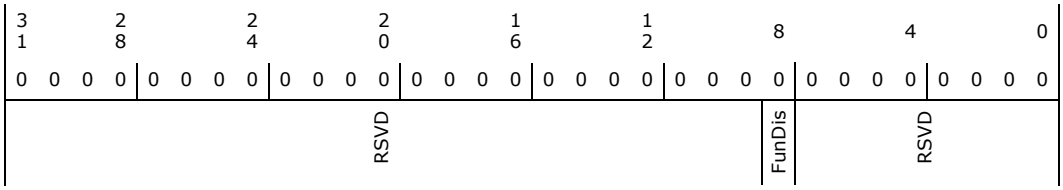
36.5.1 D20:F2 Function Disable (PSF_2_AGNT_T0_SHDW_PCIEN_TRH_RS0_D20_F2_OFFSET1)—Offset 11Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	Reserved.
8	0h	**Thermal Reporting Function Disable (FunDis):** When set, transactions targeting this PCI function will not be positively decoded, and cfgWr transactions to this function will not be carried out.
7:0	0h RO	Reserved.

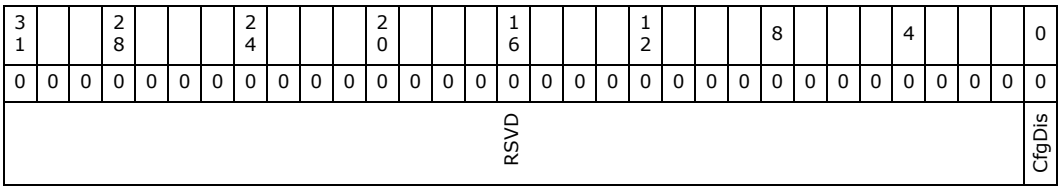
36.5.2 D20:F2 PCI Configuration Disable (PSF_2_AGNT_T0_SHDW_CFG_DIS_CSE_RS0_D20_F2)—Offset 138h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h	**CfgDis:** If set, the PCI configuration space of this device is switched off, and hidden from the OS.

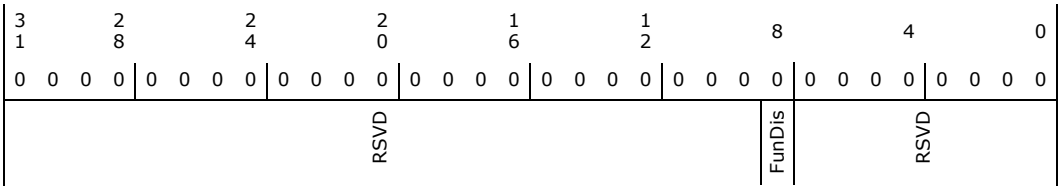
36.5.3 D20:F0 Function Disable (PSF_2_AGNT_T0_SHDW_PCIEN_SCC_RS0_D20_F0)—Offset 31Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	Reserved.
8	0h	**xHCI Function Disable (FunDis):** When set, transactions targeting this PCI function will not be positively decoded, and cfgWr transactions to this function will not be carried out.
7:0	0h RO	Reserved.

36.5.4 D20:F0 PCI Configuration Disable (PSF_2_AGNT_T0_SHDW_CFG_DIS_CSE_RS0_D20_F0)—Offset 338h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																															CfgDis

Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h	**CfgDis:** If set, the PCI configuration space of this device is switched off, and hidden from the OS.

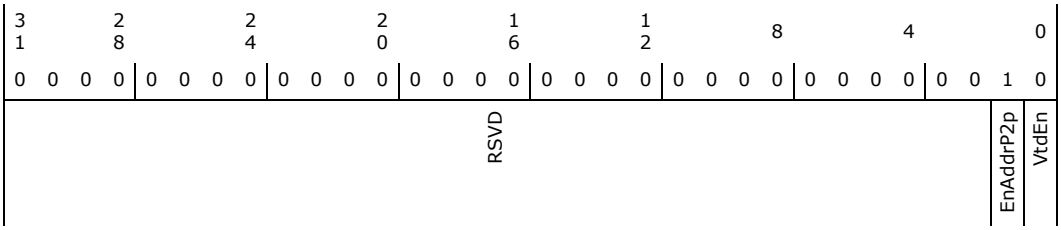
36.5.5 Rootspace Configuration (PSF_2_ROOTSPACE_CONFIG_RS0)—Offset 4014h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 2h



Bit Range	Default and Access	Field Name (ID): Description
31:2	0h RO	Reserved.
1	1h	**Address Enable (EnAddrP2p):** Default value=0x1, If set, address-based p2p transactions are allowed for this root space.
0	0h	**VTd Enable (VtdEn):** Default value=0x0, If set Intel Virtualization Technology for Directed I/O is enabled for this root space. This bit should reflect the system-wide VT-D setting, and is typically maintained by BIOS

36.6 PSF3 PCR Registers Summary

These registers are within the PCH Private Configuration Space which is accessible through the PCH Sideband Interface. They can be accessed via (SBREG_BAR + PortID + Register Offset).

Table 36-7. Summary of PSF3 PCR Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
11Ch	11Fh	D16:F4 Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_IECSE_RS0_D16_F4)	0h
138h	13Bh	D16:F4 PCI Configuration Disable (PSF_3_AGNT_T0_SHDW_CFG_DIS_IECSE_RS0_D16_F4)	0h
21Ch	21Fh	D16:F1 Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_IECSE_RS0_D16_F1)	0h
238h	23Bh	D16:F1 PCI Configuration Disable (PSF_3_AGNT_T0_SHDW_CFG_DIS_IECSE_RS0_D16_F1)	0h
41C	41Fh	D16:F0 Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_IECSE_RS0_D16_F0)	0h
438	43Bh	D16:F0 PCI Configuration Disable (PSF_3_AGNT_T0_SHDW_CFG_DIS_IECSE_RS0_D16_F0)	0h
51Ch	51Fh	D16:F3 Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_IECSE_RS0_D16_F3)	0h

Table 36-7. Summary of PSF3 PCR Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
538h	53Bh	D16:F3 PCI Configuration Disable (PSF_3_AGNT_T0_SHDW_CFG_DIS_IECSE_RS0_D16_F3)	0h
61C	61Fh	D16:F2 Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_IEPTIO_RS0_D16_F2)	0h
638h	63Bh	D16:F2 PCI Configuration Disable (PSF_3_AGNT_T0_SHDW_CFG_DIS_IEPTIO_RS0_D16_F2)	0h
800h	803h	D20:F4 Intel Trace Hub PCI Base Address (PSF_3_AGNT_T0_SHDW_BAR0_NPK_RS0_D20_F4)	0h
81Ch	81Fh	D20:F4 Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_NPK_RS0_D20_F4)	0h
838h	83Bh	D20:F4 Intel Trace Hub PCI Configuration Disable (PSF_3_AGNT_T0_SHDW_CFG_DIS_NPK_RS0_D20_F4)	0h
A1Ch	A1Fh	D31:F0 Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_LPC_RS0_D31_F0)	0h
A38h	A3Bh	D31:F0 IPCI Configuration Disable (PSF_3_AGNT_T0_SHDW_CFG_DIS_LPC_RS0_D31_F0)	0h
B1C	B1Fh	D31:F4 Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_SMB_RS0_D31_F4)	0h
B38h	B3Bh	D31:F4 PCI Configuration Disable (PSF_3_AGNT_T0_SHDW_CFG_DIS_SMB_RS0_D31_F4)	0h
131Ch	131Fh	D31:F5 Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_SPI_RS0_D31_F5)	0h
1338h	133Bh	D31:F5 PCI Configuration Disable (PSF_3_AGNT_T0_SHDW_CFG_DIS_SPI_RS0_D31_F5)	0h
141C	141Fh	D31:F0 Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_SPI_RS0_D31_F0)	0h
1438h	143Bh	D31:F0 PCI Configuration Disable (PSF_3_AGNT_T0_SHDW_CFG_DIS_SPI_RS0_D31_F0)	0h
161Ch	161Fh	D31:F6 PCI Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_GBE_RS0_D31_F6)	0h
1638h	163Bh	D31:F6 PCI Configuration Disable (PSF_3_AGNT_T0_SHDW_CFG_DIS_SPI_RS0_D31_F6)	0h
181Ch	181Fh	D31:F3 PCI Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_AUD_RS0_D31_F3)	0h
1838h	183Bh	D31:F3 PCI Configuration Disable (PSF_3_AGNT_T0_SHDW_CFG_DIS_AUD_RS0_D31_F3)	0h
191Ch	191Fh	D17:F0 PCI Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_MSU_RS0_D17_F0)	0h
1A1Ch	1A1Fh	D17:F1 PCI Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_MSU_RS0_D17_F1)	0h
4014h	4017h	Rootspace Configuration (PSF_1_ROOTSPACE_CONFIG_RS0)h	2h
4058h	4057h	Multicast Control Register (PSF_1_PSF_MC_CONTROL_MCAST0_RS0_EOI)	0h
4064h	406Fh	Destination ID (PSF_1_PSF_MC_AGENT_MCAST0_RS0_TGT0_EOI)	0h

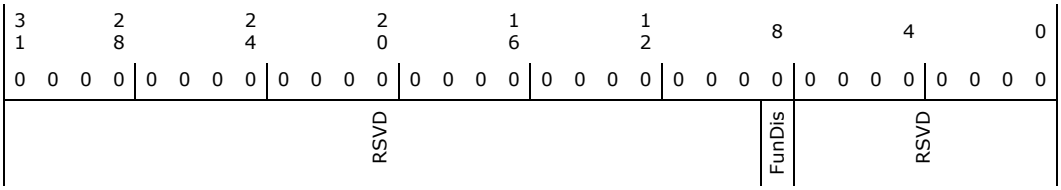
36.6.1 D16:F4 Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_IECSE_RS0_D16_F4)—Offset 11Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	Reserved.
8	0h	**Innovation Engine HECI #3 FunDis:** When set, transactions targeting this PCI function will not be positively decoded, and cfgWr transactions to this function will not be carried out.
7:0	0h RO	Reserved.

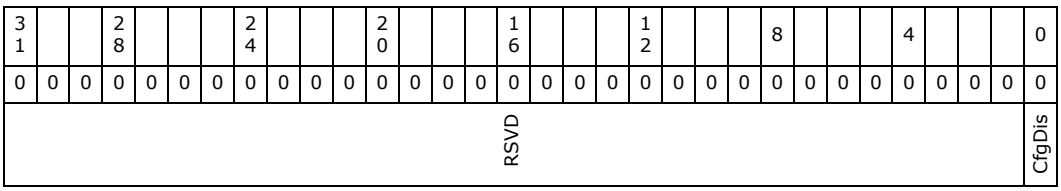
36.6.2 D16:F4 PCI Configuration Disable (PSF_3_AGNT_T0_SHDW_CFG_DIS_IECSE_RS0_D16_F4)—Offset 138h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h	**CfgDis:** If set, the PCI configuration space of this device is switched off, and hidden from the OS.

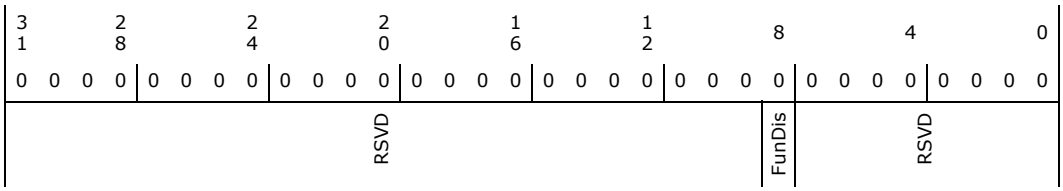
36.6.3 D16:F1 Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_IECSE_RS0_D16_F1)—Offset 21Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	Reserved.
8	0h	**Innovation Engine HECI #2 FunDis:** When set, transactions targeting this PCI function will not be positively decoded, and cfgWr transactions to this function will not be carried out.
7:0	0h RO	Reserved.

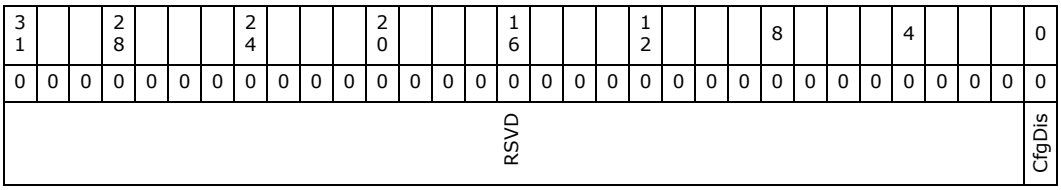
36.6.4 D16:F1 PCI Configuration Disable (PSF_3_AGNT_T0_SHDW_CFG_DIS_IECSE_RS0_D16_F1)—Offset 238h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h	**CfgDis:** If set, the PCI configuration space of this device is switched off, and hidden from the OS.

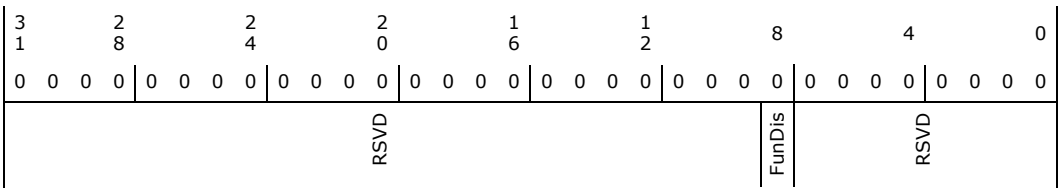
36.6.5 D16:F0 Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_IECSE_RS0_D16_F0)—Offset 41Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	Reserved.
8	0h	**Innovation Engine HECI #1 FunDis:** When set, transactions targeting this PCI function will not be positively decoded, and cfgWr transactions to this function will not be carried out.
7:0	0h RO	Reserved.

36.6.6 D16:F0 PCI Configuration Disable (PSF_3_AGNT_T0_SHDW_CFG_DIS_IECSE_RS0_D16_F0)—Offset 438h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																															CfgDis

Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h	**CfgDis:** If set, the PCI configuration space of this device is switched off, and hidden from the OS.

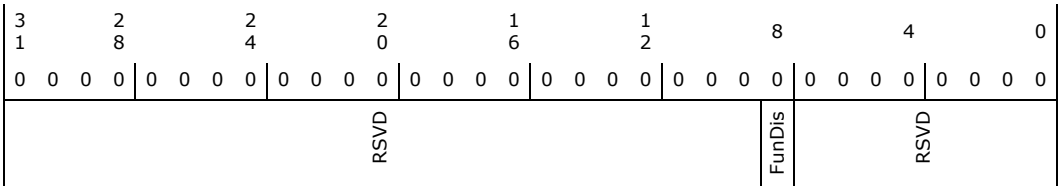
36.6.7 D16:F3 Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_IECSE_RS0_D16_F3)—Offset 51Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	Reserved.
8	0h	**Innovation Engine KT -R FunDis:** When set, transactions targeting this PCI function will not be positively decoded, and cfgWr transactions to this function will not be carried out.
7:0	0h RO	Reserved.

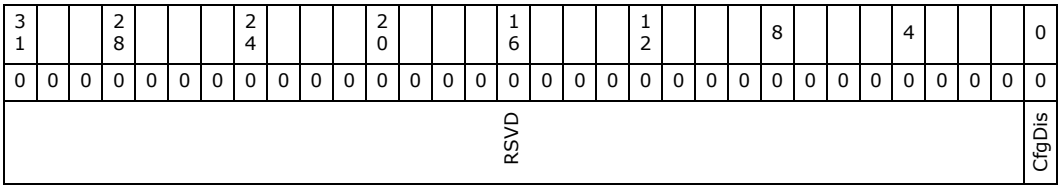
36.6.8 D16:F3 PCI Configuration Disable (PSF_3_AGNT_T0_SHDW_CFG_DIS_IECSE_RS0_D16_F3)—Offset 538h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h	**CfgDis:** If set, the PCI configuration space of this device is switched off, and hidden from the OS.

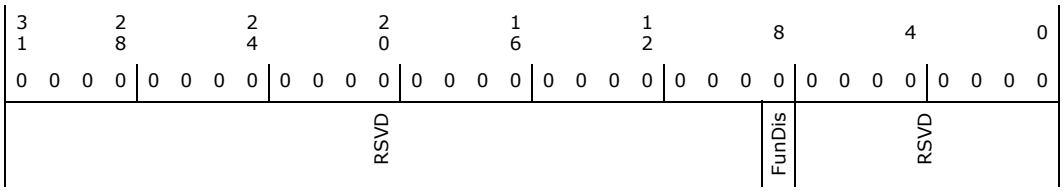
36.6.9 D16:F2 Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_IEPTIO_RS0_D16_F2)—Offset 61Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	Reserved.
8	0h	**Innovation Engine IDE-R FunDis:** When set, transactions targeting this PCI function will not be positively decoded, and cfgWr transactions to this function will not be carried out.
7:0	0h RO	Reserved.

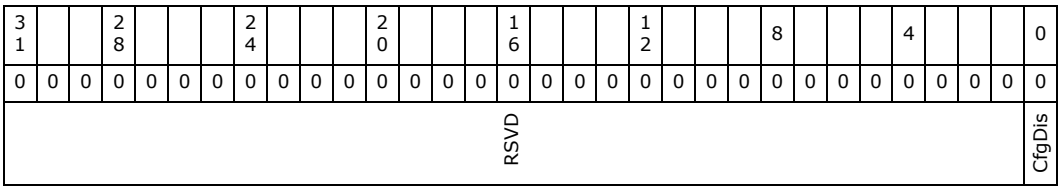
36.6.10 D16:F2 PCI Configuration Disable (PSF_3_AGNT_T0_SHDW_CFG_DIS_IEPTIO_RS0_D16_F2)—Offset 638h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h	**CfgDis:** If set, the PCI configuration space of this device is switched off, and hidden from the OS.

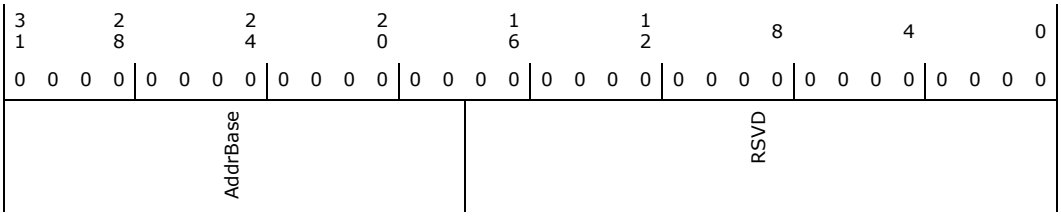
36.6.11 D20:F4 Intel Trace Hub PCI Base Address (PSF_3_AGNT_T0_SHDW_BAR0_NPK_RS0_D20_F4)—Offset 800h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:18	0h RW	**Trace Hub PCI BAR (AddrBase)**
17:0	0h RO	Reserved.

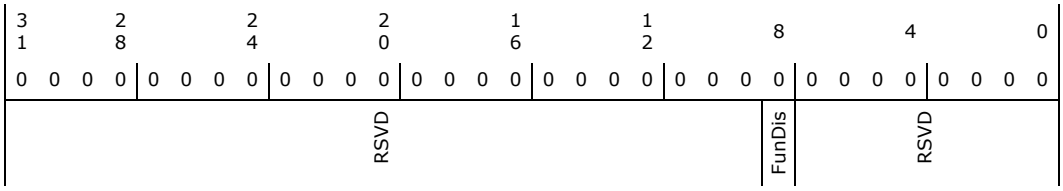
36.6.12 D20:F4 Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_NPK_RS0_D20_F4)—Offset 81Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	Reserved.
8	0h	**Intel Trace Hub FunDis:** When set, transactions targeting this PCI function will not be positively decoded, and cfgWr transactions to this function will not be carried out.
7:0	0h RO	Reserved.

36.6.13 D20:F4 Intel Trace Hub PCI Configuration Disable (PSF_3_AGNT_T0_SHDW_CFG_DIS_NPK_RS0_D20_F4)—Offset 838h

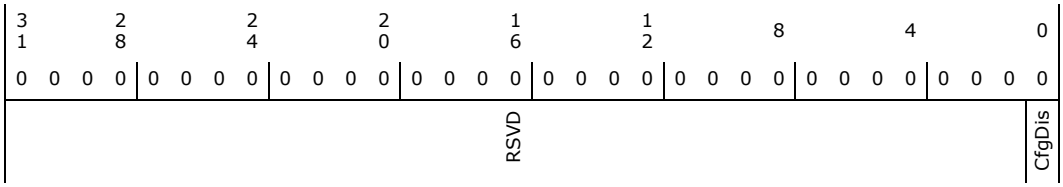
If set, this bit disables the shadowed PCI configuration header of the device. When a PCI configuration header is disabled, cfgWr transactions are no longer shadowed, and whatever the current contents of the address resources (BAR[0..1]) and requester ID (B:D:F) becomes the fixed location of the device in the address space. With its PCI configuration header disabled, the device effectively becomes an ACPI device, with fixed address ranges and requester ID.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h	**CfgDis:** Default value=0x0, If set, the PCI configuration space of this device is switched off, and hidden from the OS. This bit is writable through IOSF SB interface only.

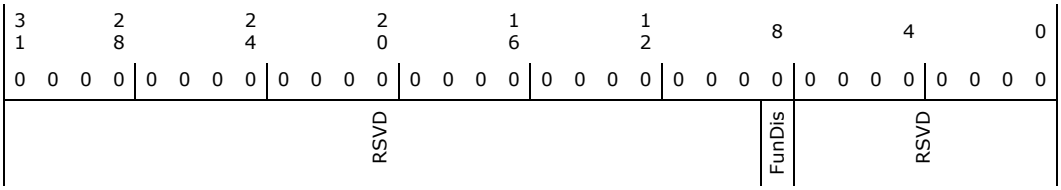
36.6.14 D31:F0 Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_LPC_RS0_D31_F0)—Offset A1Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	Reserved.
8	0h	**LPC FunDis:** When set, transactions targeting this PCI function will not be positively decoded, and cfgWr transactions to this function will not be carried out.
7:0	0h RO	Reserved.

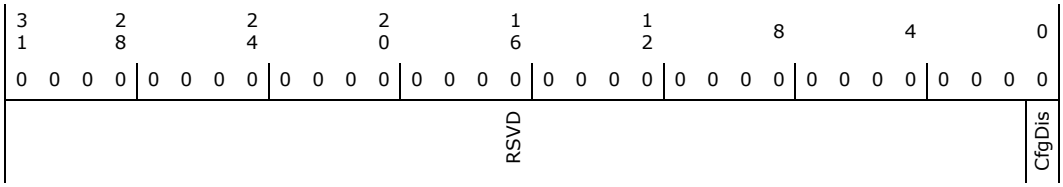
36.6.15 D31:F0 PCI Configuration Disable (PSF_3_AGNT_T0_SHDW_CFG_DIS_LPC_RS0_D31_F0)—Offset A38h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h	**CfgDis:** Default value=0x0, If set, the PCI configuration space of this device is switched off, and hidden from the OS. This bit is writable through IOSF SB interface only.

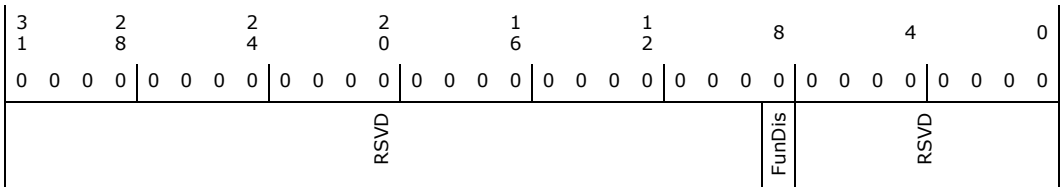
36.6.16 D31:F4 Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_SMB_RS0_D31_F4)—Offset B1Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	Reserved.
8	0h	**SMBus FunDis:** When set, transactions targeting this PCI function will not be positively decoded, and cfgWr transactions to this function will not be carried out.
7:0	0h RO	Reserved.

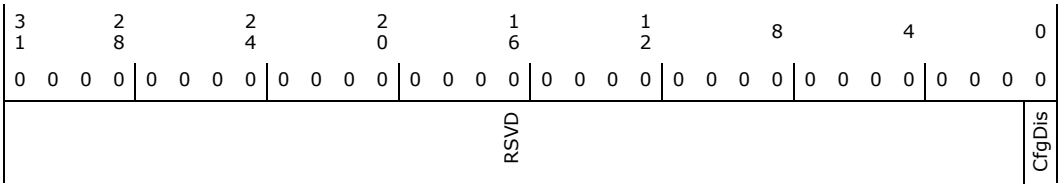
36.6.17 D31:F4 PCI Configuration Disable (PSF_3_AGNT_T0_SHDW_CFG_DIS_SMB_RS0_D31_F4)—Offset B38h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h	**CfgDis:** Default value=0x0, If set, the PCI configuration space of this device is switched off, and hidden from the OS. This bit is writable through IOSF SB interface only.

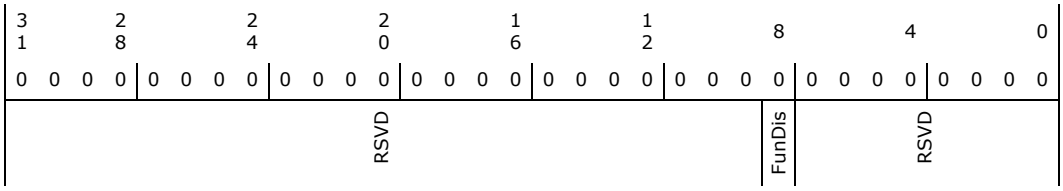
36.6.18 D31:F5 Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_SPI_RS0_D31_F5)—Offset 131Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	Reserved.
8	0h	**SPI FunDis:** When set, transactions targeting this PCI function will not be positively decoded, and cfgWr transactions to this function will not be carried out.
7:0	0h RO	Reserved.

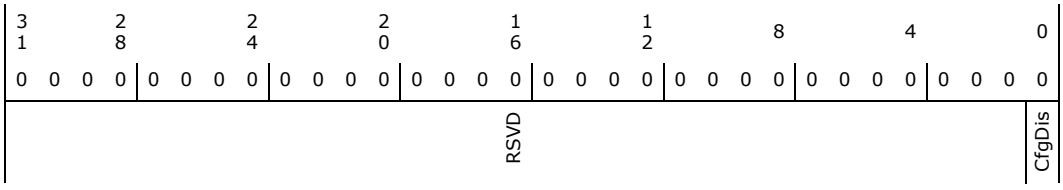
36.6.19 D31:F5 PCI Configuration Disable (PSF_3_AGNT_T0_SHDW_CFG_DIS_SPI_RS0_D31_F5)—Offset 1338h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h	**CfgDis:** Default value=0x0, If set, the PCI configuration space of this device is switched off, and hidden from the OS. This bit is writable through IOSF SB interface only.

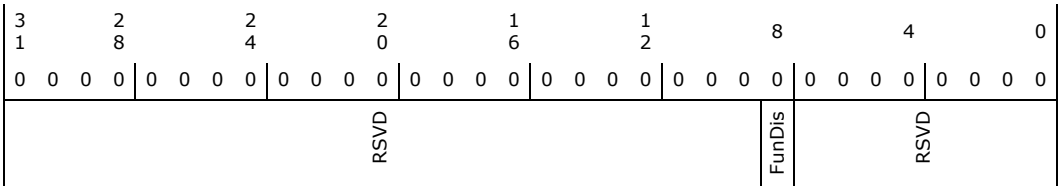
36.6.20 D31:F0 Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_SPI_RS0_D31_F0)—Offset 141Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	Reserved.
8	0h	**eSPI FunDis:** When set, transactions targeting this PCI function will not be positively decoded, and cfgWr transactions to this function will not be carried out.
7:0	0h RO	Reserved.

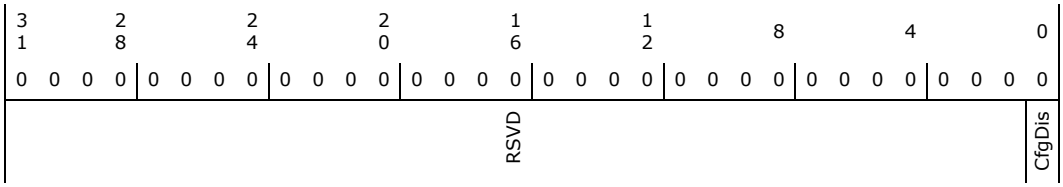
36.6.21 D31:F0 PCI Configuration Disable (PSF_3_AGNT_T0_SHDW_CFG_DIS_SPI_RS0_D31_F0)—Offset 1438h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h	**CfgDis:** Default value=0x0, If set, the PCI configuration space of this device is switched off, and hidden from the OS. This bit is writable through IOSF SB interface only.

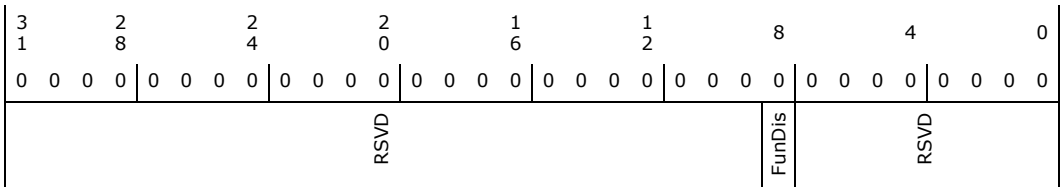
36.6.22 D31:F6 PCI Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_GBE_RS0_D31_F6)—Offset 161Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	Reserved.
8	0h RO	**GbE Function Disable (FunDis):** When set, transactions targeting this PCI function will not be positively decoded, and cfgWr transactions to this function will not be carried out. SW can not write to this bit and disable the 1GbE device. It must be disabled via Soft Strap.
7:0	0h RO	Reserved.

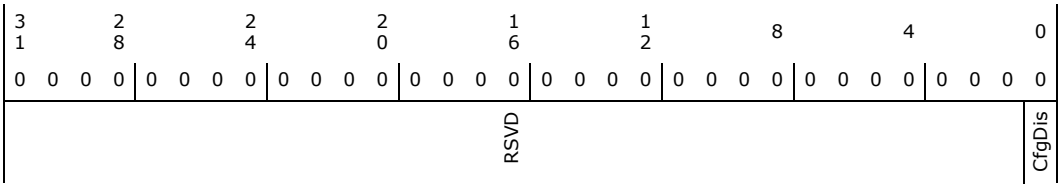
36.6.23 D31:F6 PCI Configuration Disable (PSF_3_AGNT_T0_SHDW_CFG_DIS_SPI_RS0_D31_F6)—Offset 1638h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h	**CfgDis:** Default value=0x0, If set, the PCI configuration space of this device is switched off, and hidden from the OS. This bit is writable through IOSF SB interface only.

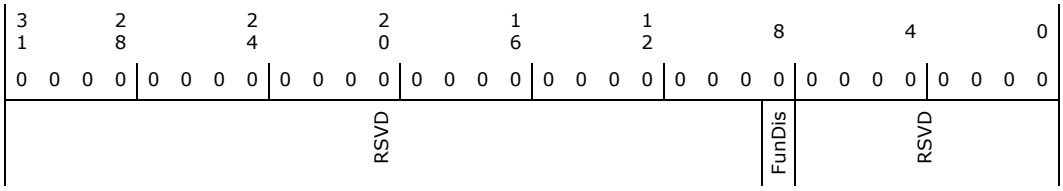
36.6.24 D31:F3 PCI Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_AUD_RS0_D31_F3)—Offset 181Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	Reserved.
8	0h	**Audio Disable (FunDis):** When set, transactions targeting this PCI function will not be positively decoded, and cfgWr transactions to this function will not be carried out.
7:0	0h RO	Reserved.

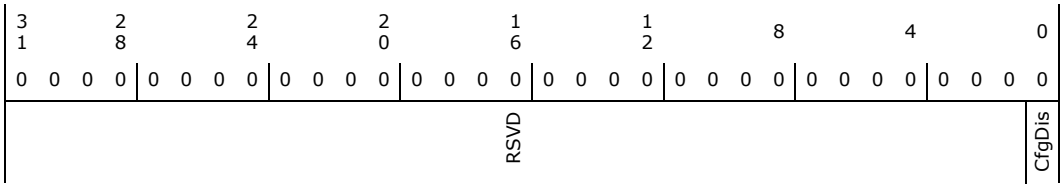
36.6.25 D31:F3 PCI Configuration Disable (PSF_3_AGNT_T0_SHDW_CFG_DIS_AUD_RS0_D31_F3)—Offset 1838h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:1	0h RO	Reserved.
0	0h	**CfgDis:** Default value=0x0, If set, the PCI configuration space of this device is switched off, and hidden from the OS. This bit is writable through IOSF SB interface only.

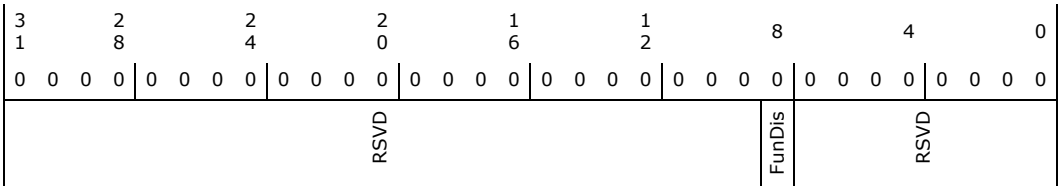
36.6.26 D17:F0 PCI Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_MSU_RS0_D17_F0)—Offset 191Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	Reserved.
8	0h	**MROM0 Disable (FunDis):** When set, transactions targeting this PCI function will not be positively decoded, and cfgWr transactions to this function will not be carried out.
7:0	0h RO	Reserved.

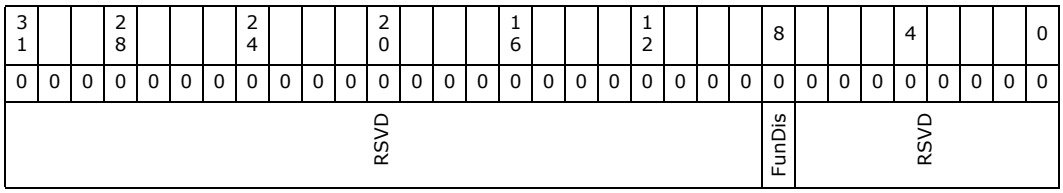
36.6.27 D17:F1 PCI Function Disable (PSF_3_AGNT_T0_SHDW_PCIEN_MSU_RS0_D17_F1)—Offset 1A1Ch

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	Reserved.
8	0h	**MROM1 Disable (FunDis):** When set, transactions targeting this PCI function will not be positively decoded, and cfgWr transactions to this function will not be carried out.
7:0	0h RO	Reserved.

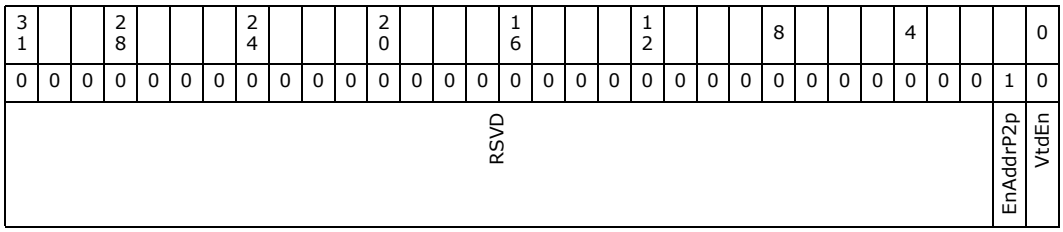
36.6.28 Root Space Config (PSF_3_ROOTSPACE_CONFIG_RS0)—Offset 4014h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 2h



Bit Range	Default and Access	Field Name (ID): Description
31:2	0h RO	Reserved.
1	1h	**Address Enable (EnAddrP2p):** Default value=0x1, If set, address-based p2p transactions are allowed for this root space.
0	0h	**VTd Enable (VtdEn):** Default value=0x0, If set Intel Virtualization Technology for Directed I/O is enabled for this root space. This bit should reflect the system-wide VT-D setting, and is typically maintained by BIOS

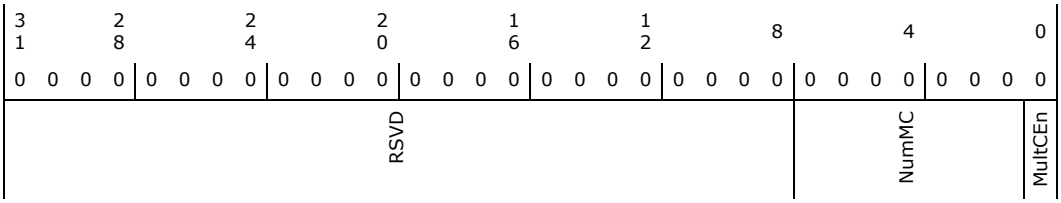
36.6.29 Multicast Control (PSF_3_PSF_MC_CONTROL_MCAST0_RS0_EOI)—Offset 4058h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:8	0h RO	Reserved.
7:1	0h	**Multicast Agent (NumMC):** Number of multicast Agents on this root-space on the PSF segment.
0	0h	**Multicast Enable (MultCEn):** Multicast Enable. If set Multicast transactions are supported on this root-space, and can be received on any PSF port.

36.6.30 Destination ID (PSF_3_PSF_MC_AGENT_MCAST0_RS0_TGT0_EOI)—Offset 4064h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
CHANMAP	RSVD							PSFID								PortGroupID	PortID							ChannelID							

Bit Range	Default and Access	Field Name (ID): Description
31	0h	**Channel ID Map (CHANMAP):** If this bit is set, the ChannelID field is ignored and looked up in the CHANMAP register set that belongs to PortGroupID:PortID
30:24	0h RO	Reserved.
23:16	0h	**PSF ID (PSFID):** Default value=0x0,
15	0h	**Port Group ID (PortGroupID):** Default value=0x0
14:8	0h	**Port ID (PortID):** Since 1'b0 indicates source-decode, the port ID needs to be larger than 0
7:0	0h	**Channel ID (ChannelID):** Default value=0x0,

36.7 PSF4 PCR Registers Summary

These registers are within the PCH Private Configuration Space which is accessible through the PCH Sideband Interface. They can be accessed via (SBREG_BAR + PortID + Register Offset).

Table 36-8. Summary of PSF4 PCR registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
4014h	4017h	PSF_4_ROOTSPACE_CONFIG_RS0)—Offset 4014h	00000002h

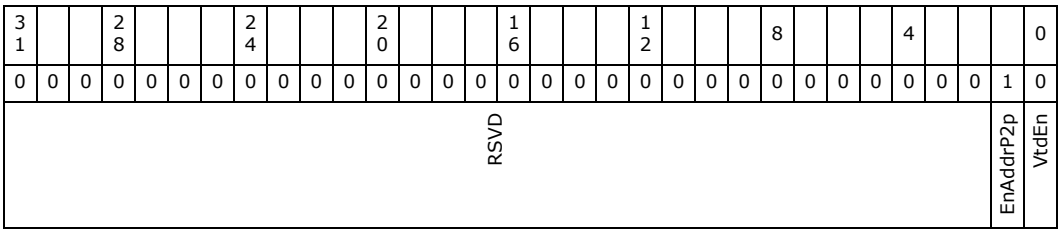
36.7.1 Rootspace Configuration (PSF_4_ROOTSPACE_CONFIG_RS0)—Offset 4014h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 2h



Bit Range	Default and Access	Field Name (ID): Description
31:2	0h RO	Reserved.
1	1h	**Address Enable (EnAddrP2p):** Default value=0x1, If set, address-based p2p transactions are allowed for this root space.
0	0h	**VTd Enable (VtdEn):** Default value=0x0, If set Intel Virtualization Technology for Directed I/O is enabled for this root space. This bit should reflect the system-wide VT-D setting, and is typically maintained by BIOS

36.8 I/O Trap Registers Summary

These registers are within the PCH Private Configuration Space which is accessible through the PCH Sideband Interface. They can be accessed via (SBREG_BAR + PortID + Register Offset).

Table 36-9. Summary of I/O Trap Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
1E00h	1E03h	Trap Status Register (TRPSTS)—Offset 1E00h	0h
1E10h	1E13h	Trapped Cycle Register (TRPCYC1)—Offset 1E10h	0h
1E18h	1E1Bh	Trapped Write Data Register (TRPWRDATA1)—Offset 1E18h	0h
1E80h	1E83h	I/O Trap Registers 1 (IOTRP1_1)—Offset 1E80h	0h
1E84h	1E87h	I/O Trap Registers 1 (IOTRP1_2)—Offset 1E84h	0h
1E88h	1E8Bh	I/O Trap Registers 2 (IOTRP2_1)—Offset 1E88h	0h
1E8Ch	1E8Fh	I/O Trap Registers 2 (IOTRP2_2)—Offset 1E8Ch	0h
1E90h	1E93h	I/O Trap Registers 3 (IOTRP3_1)—Offset 1E90h	0h
1E94h	1E97h	I/O Trap Registers 3 (IOTRP3_2)—Offset 1E94h	0h
1E98h	1E9Bh	I/O Trap Registers 4 (IOTRP4_1)—Offset 1E98h	0h
1E9Ch	1E9Fh	I/O Trap Registers 4 (IOTRP4_2)—Offset 1E9Ch	0h

36.8.1 Trap Status Register (TRPSTS)—Offset 1E00h

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD																												SMISTAT			

Bit Range	Default and Access	Field Name (ID): Description
31:4	0h RO	Reserved.
3:0	0h RWC	**Cycle Trap SMI# Status (SMISTAT):** These bits are set by hardware when the corresponding Cycle Trap register is enabled and a matching cycle is received (and trapped). These bits are OR'ed together to create a single status bit in the Power Management register space.Note that the SMI# and trapping must be enabled in order to set these bits. This is because, in order to do the cycle comparison, packets must be delayed by several clocks from the DMI pins to the internal receiver. This delay is only enabled when at least one of the trap ranges is enabled. These bits are set before the completion is generated for the trapped cycle, thereby guaranteeing that the processor can enter the SMI# handler when the instruction completes. Each status bit is cleared by writing a '1' to the corresponding bit location in this register.

36.8.2 Trapped Cycle Register (TRPCYC1)—Offset 1E10h

This register saves information about the I/O Cycle that was trapped and generated the SMI# for software to read.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD							TRPRWR	RSVD				TRPBE				TRPADDR													RSVD			

Bit Range	Default and Access	Field Name (ID): Description
31:25	0h RO	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
24	0h RO/V	**Read-Write (TRPRWR):** 1 = Read, 0 = Write
23:20	0h RO	Reserved.
19:16	0h RO/V	**Active-High Byte Enables (TRPBE):** This is the DWord-aligned byte enables associated with the trapped cycle. A 1 in any bit location indicates that the corresponding byte is enabled in the cycle.
15:2	0h RO/V	**IO Address (TRPADDR):** This is the DWord-aligned address of the trapped cycle.
1:0	0h RO	Reserved.

36.8.3 Trapped Write Data Register (TRPWRDATA1)—Offset 1E18h

This register saves the data from I/O write cycles that are trapped for software to read.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
TRPDATA																															

Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RO/V	**Data (TRPDATA):** DWord of I/O write data. This field is undefined after trapping a read cycle.

36.8.4 I/O Trap Registers 1 (IOTRP1_1)—Offset 1E80h

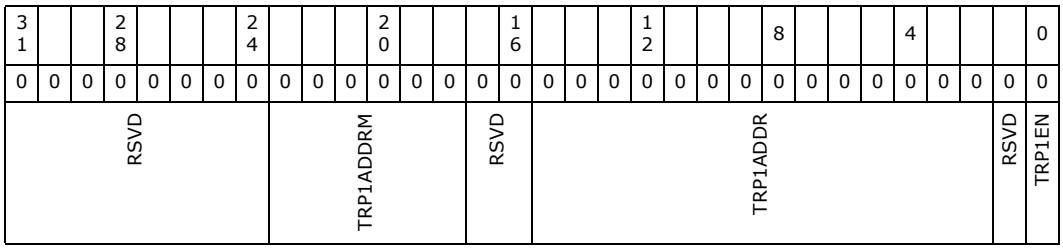
These registers are used to specify the set of I/O cycles to be trapped and to enable this functionality.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23:18	0h RW	**Address Mask (TRP1ADDRM):** A '1' in any bit position indicates that any value in the corresponding address bit in a received cycle will be treated as a match. The corresponding bit in the Address field, below, is ignored. The mask is only provided for the lower 6 bits of the DWord address, allowing for traps on address ranges up to 256 bytes in size.
17:16	0h RO	Reserved.
15:2	0h RW	**Address (TRP1ADDR):** DWord-aligned address
1	0h RO	Reserved.
0	0h RW	**Trap and SMI Enable (TRP1EN):** When this bit is set to 1, then the trapping logic specified in this register is enabled.

36.8.5 I/O Trap Registers 1 (IOTRP1_2)—Offset 1E84h

These registers are used to specify the set of I/O cycles to be trapped and to enable this functionality.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD															TRP1RWM	TRP1RW	RSVD								TRP1BEM				TRP1BE			

Bit Range	Default and Access	Field Name (ID): Description
31:18	0h RO	Reserved.
17	0h RW	**Read-Write Mask (TRP1RWM):** When this bit is '1', the trapping logic will operate on both read and write cycles. When this bit is '0', the cycle must match the type specified in bit 48.
16	0h RW	**Read/Write (TRP1RW):** 1 = Read, 0 = Write, the value in this field does not matter if bit 49 is set.
15:8	0h RO	Reserved.
7:4	0h RW	**Byte Enable Mask (TRP1BEM):** A '1' in any bit position indicates that any value in the corresponding byte enable bit in a received cycle will be treated as a match. The corresponding bit in the Byte Enables field, below, is ignored.
3:0	0h RW	**Byte Enables (TRP1BE):** Active-high, DWord-aligned byte enables

36.8.6 I/O Trap Registers 2 (IOTRP2_1)—Offset 1E88h

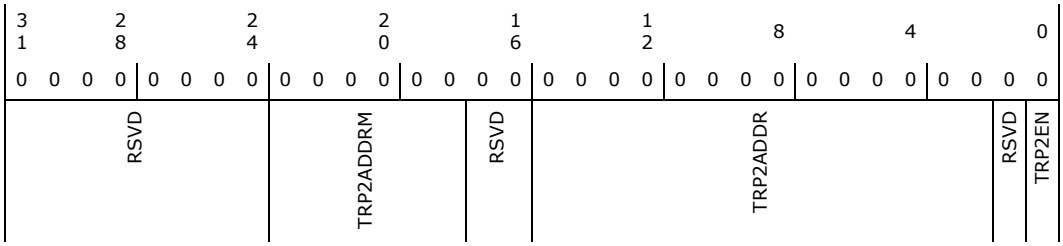
These registers are used to specify the set of I/O cycles to be trapped and to enable this functionality.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23:18	0h RW	**Address Mask (TRP2ADDRM):** A '1' in any bit position indicates that any value in the corresponding address bit in a received cycle will be treated as a match. The corresponding bit in the Address field, below, is ignored. The mask is only provided for the lower 6 bits of the DWord address, allowing for traps on address ranges up to 256 bytes in size.
17:16	0h RO	Reserved.
15:2	0h RW	**Address (TRP2ADDR):** DWord-aligned address
1	0h RO	Reserved.
0	0h RW	**Trap and SMI Enable (TRP2EN):** When this bit is set to 1, then the trapping logic specified in this register is enabled.

36.8.7 I/O Trap Registers 2 (IOTRP2_2)—Offset 1E8Ch

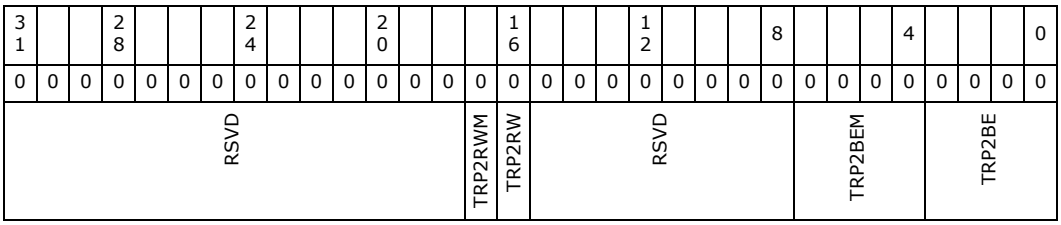
These registers are used to specify the set of I/O cycles to be trapped and to enable this functionality.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:18	0h RO	Reserved.
17	0h RW	**Read-Write Mask (TRP2RWM):** When this bit is '1', the trapping logic will operate on both read and write cycles. When this bit is '0', the cycle must match the type specified in bit 48.
16	0h RW	**Read/Write (TRP2RW):** 1 = Read, 0 = Write, the value in this field does not matter if bit 49 is set.
15:8	0h RO	Reserved.
7:4	0h RW	**Byte Enable Mask (TRP2BEM):** A '1' in any bit position indicates that any value in the corresponding byte enable bit in a received cycle will be treated as a match. The corresponding bit in the Byte Enables field, below, is ignored.
3:0	0h RW	**Byte Enables (TRP2BE):** Active-high, DWord-aligned byte enables

36.8.8 I/O Trap Registers 3 (IOTRP3_1)—Offset 1E90h

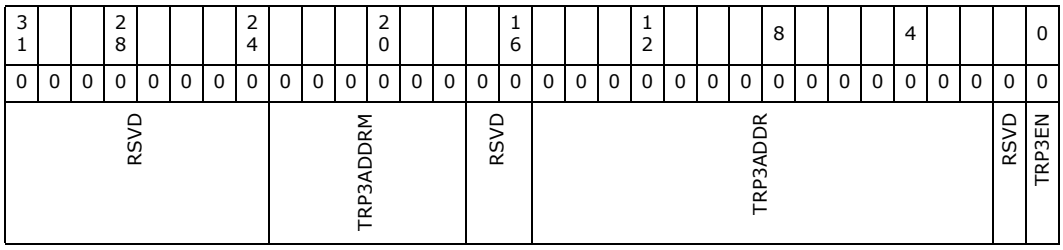
These registers are used to specify the set of I/O cycles to be trapped and to enable this functionality.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23:18	0h RW	**Address Mask (TRP3ADDRM):** A '1' in any bit position indicates that any value in the corresponding address bit in a received cycle will be treated as a match. The corresponding bit in the Address field, below, is ignored. The mask is only provided for the lower 6 bits of the DWord address, allowing for traps on address ranges up to 256 bytes in size.
17:16	0h RO	Reserved.
15:2	0h RW	**Address (TRP3ADDR):** DWord-aligned address
1	0h RO	Reserved.
0	0h RW	**Trap and SMI Enable (TRP3EN):** When this bit is set to 1, then the trapping logic specified in this register is enabled.

36.8.9 I/O Trap Registers 3 (IOTRP3_2)—Offset 1E94h

These registers are used to specify the set of I/O cycles to be trapped and to enable this functionality.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD															TRP3RWM	TRP3RW	RSVD								TRP3BEM				TRP3BE			

Bit Range	Default and Access	Field Name (ID): Description
31:18	0h RO	Reserved.
17	0h RW	**Read-Write Mask (TRP3RWM):** When this bit is '1', the trapping logic will operate on both read and write cycles. When this bit is '0', the cycle must match the type specified in bit 48.
16	0h RW	**Read/Write (TRP3RW):** 1 = Read, 0 = Write, the value in this field does not matter if bit 49 is set.
15:8	0h RO	Reserved.
7:4	0h RW	**Byte Enable Mask (TRP3BEM):** A '1' in any bit position indicates that any value in the corresponding byte enable bit in a received cycle will be treated as a match. The corresponding bit in the Byte Enables field, below, is ignored.
3:0	0h RW	**Byte Enables (TRP3BE):** Active-high, DWord-aligned byte enables

36.8.10 I/O Trap Registers 4 (IOTRP4_1)—Offset 1E98h

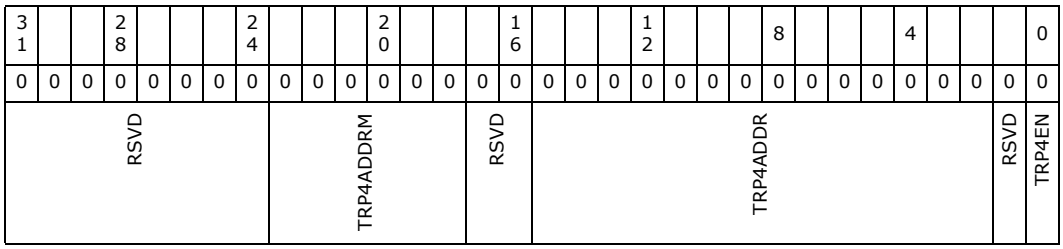
These registers are used to specify the set of I/O cycles to be trapped and to enable this functionality.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23:18	0h RW	**Address Mask (TRP4ADDRM):** A '1' in any bit position indicates that any value in the corresponding address bit in a received cycle will be treated as a match. The corresponding bit in the Address field, below, is ignored. The mask is only provided for the lower 6 bits of the DWord address, allowing for traps on address ranges up to 256 bytes in size.
17:16	0h RO	Reserved.
15:2	0h RW	**Reserved (TRP4ADDR):** DWord-aligned address
1	0h RO	Reserved.
0	0h RW	**Trap and SMI Enable (TRP4EN):** When this bit is set to 1, then the trapping logic specified in this register is enabled.

36.8.11 I/O Trap Registers 4 (IOTRP4_2)—Offset 1E9Ch

These registers are used to specify the set of I/O cycles to be trapped and to enable this functionality.

Access Method

Type: MSG Register
(Size: 32 bits)

Device:
Function:

Default: 0h

31			28				24				20				16				12				8				4				0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RSVD															TRP4RWM	TRP4RW	RSVD								TRP4BEM				TRP4BE			

Bit Range	Default and Access	Field Name (ID): Description
31:18	0h RO	Reserved.
17	0h RW	**Read-Write Mask (TRP4RWM):** When this bit is '1', the trapping logic will operate on both read and write cycles. When this bit is '0', the cycle must match the type specified in bit 48.
16	0h RW	**Read/Write (TRP4RW):** 1 = Read, 0 = Write, the value in this field does not matter if bit 49 is set.
15:8	0h RO	Reserved.
7:4	0h RW	**Byte Enable Mask (TRP4BEM):** A '1' in any bit position indicates that any value in the corresponding byte enable bit in a received cycle will be treated as a match. The corresponding bit in the Byte Enables field, below, is ignored.
3:0	0h RW	**Byte Enables (TRP4BE):** Active-high, DWord-aligned byte enables

36.9 P2SB Bridge

36.9.1 Overview

The PCH incorporates a wide variety of devices and functions. The registers within these devices are mainly accessed through the primary interface, such as PCI configuration space and IO/MMIO space. Some devices also have registers that are distributed within the PCH Private Configuration Space at individual endpoints (Target Port IDs) which are only accessible through the PCH Sideband Interface.

These PCH Private Configuration Space Registers can be addressed via SBREG_BAR or through SBI Index Data pair programming.

Table 36-10. Private Configuration Space Register Target Port IDs (Sheet 1 of 2)

PCH Device/Function Type	Target Port ID
PCIe Uplink controller (mux x8)	0x06

Table 36-10. Private Configuration Space Register Target Port IDs (Sheet 2 of 2)

PCH Device/Function Type	Target Port ID
PCIe Uplink controller	0x07
sSATA	0x0F
General Purpose I/O (GPIO) Community 5	0x11
HSIO Strap Configuration	0x89
General Purpose I/O (GPIO) Community 4	0xAB
General Purpose I/O (GPIO) Community 3	0xAC
General Purpose I/O (GPIO) Community 2	0xAD
General Purpose I/O (GPIO) Community 1	0xAE
General Purpose I/O (GPIO) Community 0	0xAF
DCI	0xB8
PSF1- Function Disable	0xBA
PSF2- Function Disable	0xBB
PSF3- Function Disable	0xBC
PSF4- Function Disable	0xBD
ISH Controller	0xBF
Real Time Clock (RTC)	0xC3
Processor Interface, 8254 Timer, HPET, APIC	0xC4
SMBus	0xC6
LPC	0xC7
USB2.0	0xCA
FIA Configuration	0x13
HDA	0xD7
SATA	0xD9
Integrated Clock Controller (ICC)	0xDC
PCIe Controller #1 (SPA)	0xE0
PCIe Controller #2 (SPB)	0xE1
PCIe Controller #3 (SPC)	0xE2
PCIe Controller #4 (SPD)	0xE3
PCIe Controller #5 (SPE)	0xE4
xHCI	0xE6
MODPHY1 (HSIO Lanes #0 - 9)	0xE9
MODPHY2 (HSIO Lanes)	0xA9
MODPHY4 (Uplink for PCIe Gen2)	0xEB
MODPHY5 (Uplink for PCIe Gen3)	0x10
eSPI / SPI	0xEE
DMI Configuration	0xEF

36.9.2 P2SB Configuration Registers Summary

Table 36-11. Summary of Internal Configuration Registers

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
0h	3h	PCI Identifier (PCIID)—Offset 0h	X_8086h
4h	5h	PCI Command (PCICMD)—Offset 4h	4h
8h	8h	Revision ID (PCIRID)—Offset 8h	0h
9h	Ch	Class Code (PCICC)—Offset 9h	58000h
Eh	Eh	PCI Header Type (PCIHTYPE)—Offset Eh	0h
10h	13h	Sideband Register Access BAR (SBREG_BAR)—Offset 10h	4h
14h	17h	Sideband Register BAR High DWORD (SBREG_BARH)—Offset 14h	0h
2Ch	2Fh	PCI Subsystem Identifiers (PCIHSS)—Offset 2Ch	0h
50h	51h	VLW Bus:Device:Function (VBDF)—Offset 50h	F8h
52h	53h	ERROR Bus:Device:Function (EBDF)—Offset 52h	F8h
54h	57h	Routing Configuration (RCFG)—Offset 54h	C700h
60h	60h	High Performance Event Timer Configuration (HPTC)—Offset 60h	0h
64h	65h	IOxAPIC Configuration (IOAC)—Offset 64h	0h
6Ch	6Dh	IOxAPIC Bus:Device:Function (IBDF)—Offset 6Ch	F8h
70h	71h	HPET Bus:Device:Function (HBDF)—Offset 70h	F8h
80h	83h	Sideband Register posted 0 (SBREGPOSTED0)—Offset 80h	0h
84h	87h	Sideband Register posted 1 (SBREGPOSTED1)—Offset 84h	0h
88h	8Bh	Sideband Register posted 2 (SBREGPOSTED2)—Offset 88h	0h
8Ch	8Fh	Sideband Register posted 3 (SBREGPOSTED3)—Offset 8Ch	0h
90h	93h	Sideband Register posted 4 (SBREGPOSTED4)—Offset 90h	0h
94h	97h	Sideband Register posted 5 (SBREGPOSTED5)—Offset 94h	0h
98h	9Bh	Sideband Register posted 6 (SBREGPOSTED6)—Offset 98h	0h
9Ch	9Fh	Sideband Register posted 7 (SBREGPOSTED7)—Offset 9Ch	0h
A0h	A3h	Display Bus:Device:Function (DISPBDF)—Offset A0h	60010h
A4h	A5h	ICC Register Offsets (ICCOS)—Offset A4h	0h
B0h	B3h	Endpoint Mask 0 (EPMASK0)—Offset B0h	0h
B4h	B7h	Endpoint Mask 1 (EPMASK1)—Offset B4h	0h
B8h	BBh	Endpoint Mask 2 (EPMASK2)—Offset B8h	0h
BCh	BFh	Endpoint Mask 3 (EPMASK3)—Offset BCh	0h
C0h	C3h	Endpoint Mask 4 (EPMASK4)—Offset C0h	0h
C4h	C7h	Endpoint Mask 5 (EPMASK5)—Offset C4h	0h
C8h	CBh	Endpoint Mask 6 (EPMASK6)—Offset C8h	0h
CCh	CFh	Endpoint Mask 7 (EPMASK7)—Offset CCh	0h
D0h	D3h	SBI Address (SBIADDR)—Offset D0h	0h
D4h	D7h	SBI Data (SBIDATA)—Offset D4h	0h
D8h	D9h	SBI Status (SBISTAT)—Offset D8h	0h
DAh	DBh	SBI Routing Identification (SBIRID)—Offset DAh	0h
DCh	DFh	SBI Extended Address (SBIEXTADDR)—Offset DCh	0h
E0h	E4h	P2SB Control (P2SBC) - Offset E0h	0h

Table 36-11. Summary of Internal Configuration Registers (Continued)

Offset Start	Offset End	Register Name (ID)—Offset	Default Value
E4h	E4h	Power Control Enable (PCE)—Offset E4h	1h

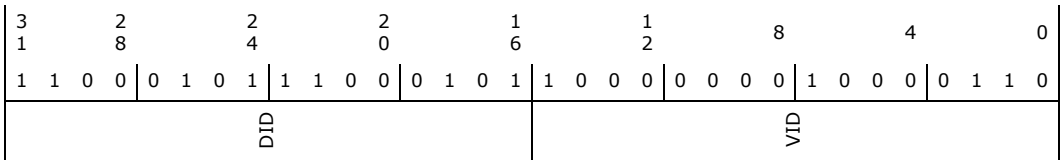
36.9.3 PCI Identifier (PCIID)—Offset 0h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: C5C58086h



Bit Range	Default and Access	Field Name (ID): Description
31:16	-- RO	**Device Identification (DID):** Indicates the Device ID of the controller. See the Device and Version ID Table in Volume 1 of the datasheet for the default value.
15:0	8086h RO	**Vendor Identification (VID):** Indicates Intel.

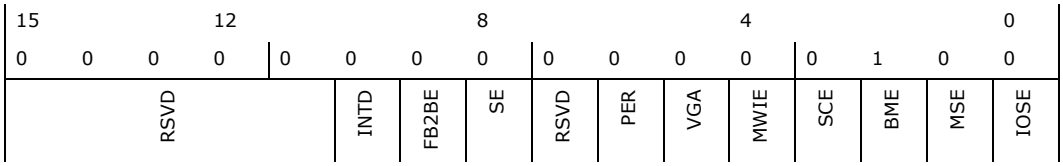
36.9.4 PCI Command (PCICMD)—Offset 4h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 1

Default: 4h



Bit Range	Default and Access	Field Name (ID): Description
15:11	0h RO	Reserved.
10	0h RO	**Interrupt Disable (INTD):** P2SB does not issue any interrupts on its own behalf
9	0h RO	**Fast Back to Back Enable (FB2BE):** Not applicable

Bit Range	Default and Access	Field Name (ID): Description
8	0h RO	**SERR# Enable (SE):** P2SB does not issue SERR# on its own behalf.
7	0h RO	Reserved.
6	0h RO	**Parity Error Response (PER):** No support for parity error detection.
5	0h RO	**VGA Palette Snoop (VGA):** Not applicable.
4	0h RO	**Memory Write and Invalidate Enable (MWIE):** Not applicable.
3	0h RO	**Special Cycle Enable (SCE):** Not applicable.
2	1h RO	**Bus Master Enable (BME):** Bus mastering cannot be disabled as this device acts as a proxy for non-PCI devices.
1	0h RW	**Memory Space Enable (MSE):** Will control the P2SB acceptance of PCI MMIO BARs only. Other legacy regions are unaffected by this bit.
0	0h RW	**I/O Space Enable (IOSE):** Legacy regions are unaffected by this bit.

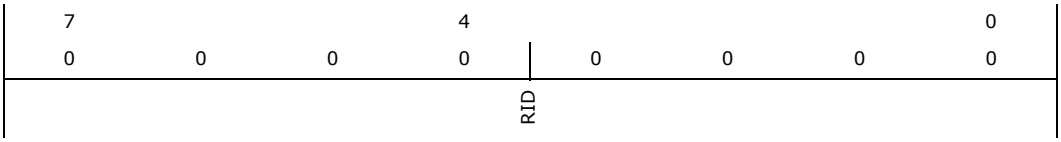
36.9.5 Revision ID (PCIRID)—Offset 8h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7:0	0h RO	**Revision ID (RID):** Indicates stepping of the host controller. Refer to Device and Revision ID table in Vol1 of the datasheet for specific value.

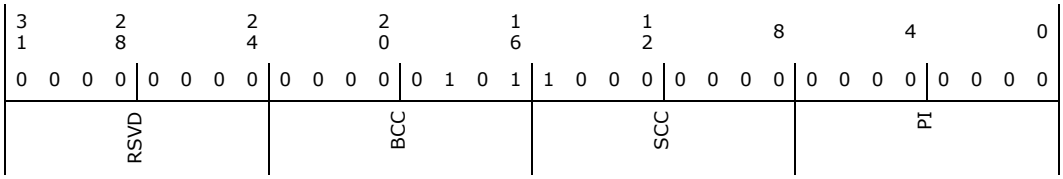
36.9.6 Class Code (PCICC)—Offset 9h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: 58000h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RO	Reserved.
23:16	5h RO	**Base Class Code (BCC):** Indicates a memory controller device class.
15:8	80h RO	**Sub-Class Code (SCC):** Indicates an unspecified other memory controller.
7:0	0h RO	**Programming Interface (PI):** No programming interface.

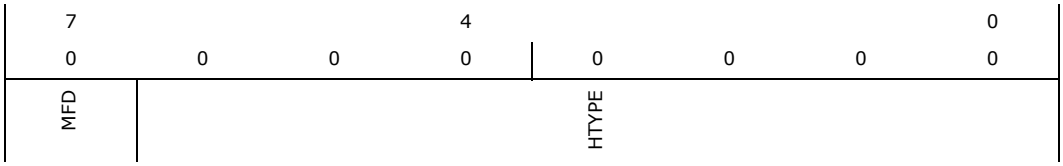
36.9.7 PCI Header Type (PCIHTYPE)—Offset Eh

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7	0h RO	**Multi-Function Device (MFD):** Indicates that this is part of a multi-function device.
6:0	0h RO	**Header Type (HTYPE):** Indicates a generic device header.

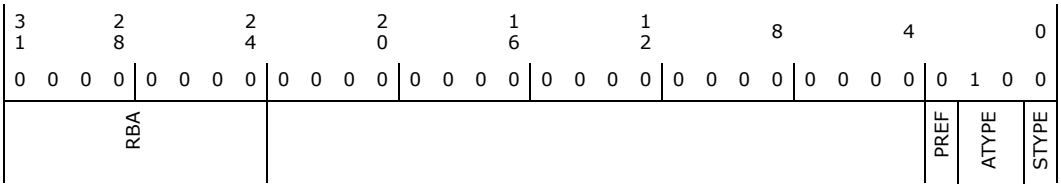
36.9.8 Sideband Register Access BAR (SBREG_BAR)—Offset 10h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: 4h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RW	**Register Base Address (RBA):** Lower DWORD of the base address for the sideband register access BAR.
23:4	0h RO	**:** Hardwired to 0 to request a BAR of 16MB
3	0h RO	**Prefetchable (PREF):** Indicates this is not prefetchable.
2:1	2h RO	**Address Type (ATYPE):** Indicates that this can be placed anywhere in 64b space.
0	0h RO	**Space Type (STYPE):** Indicates memory space

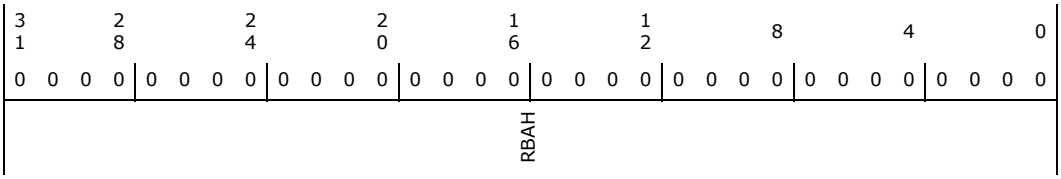
36.9.9 Sideband Register BAR High DWORD (SBREG_BARH)—Offset 14h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Register Base Address (RBAH):** Upper DWORD of the base address for the sideband register access BAR.

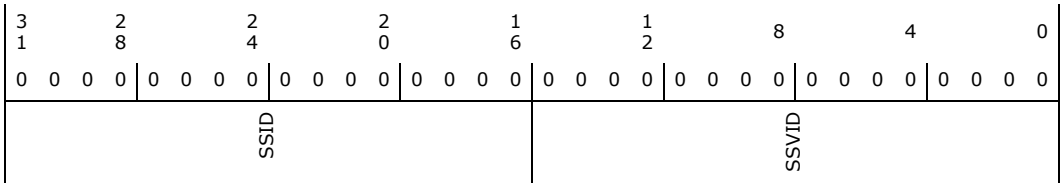
36.9.10 PCI Subsystem Identifiers (PCIHSS)—Offset 2Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RW/O	**Subsystem ID (SSID):** Written by BIOS. Not used by hardware.
15:0	0h RW/O	**Subsystem Vendor ID (SSVID):** Written by BIOS. Not used by hardware.

36.9.11 VLW Bus:Device:Function (VBDF)—Offset 50h

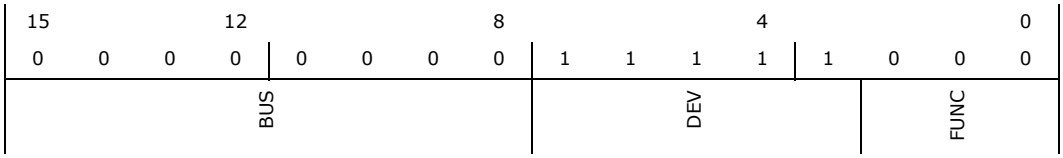
This register specifies the bus:device:function ID that will be used for its Requester ID. This field is default to Bus 0: Device 31: Function 0 after reset. BIOS shall program this field accordingly if unique bus:device:function number is required.

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 1

Default: F8h



Bit Range	Default and Access	Field Name (ID): Description
15:8	0h RW	**Bus Number (BUS)**
7:3	1Fh RW	**Device Number (DEV)**
2:0	0h RW	**Function Number (FUNC)**

36.9.12 ERROR Bus:Device:Function (EBDF)—Offset 52h

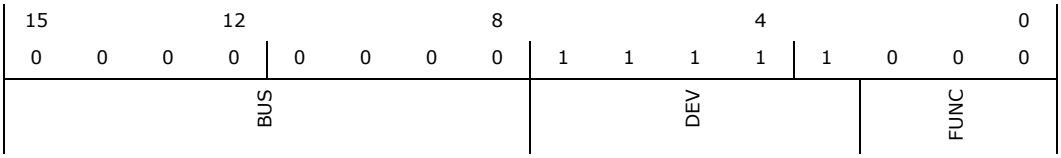
This register specifies the bus:device:function ID that the Error Signaling messages will use for its Requester ID. This field is default to Bus 0: Device 31: Function 0 after reset. BIOS shall program this field accordingly if unique bus:device:function number is required.

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 1

Default: F8h



Bit Range	Default and Access	Field Name (ID): Description
15:8	0h RW	**Bus Number (BUS)**
7:3	1Fh RW	**Device Number (DEV)**
2:0	0h RW	**Function Number (FUNC)**

36.9.13 Routing Configuration (RCFG)—Offset 54h

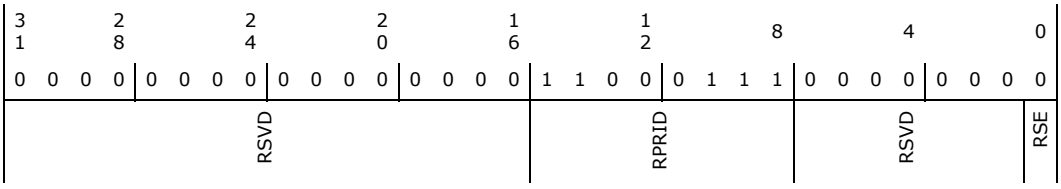
This register contains information used for routing transactions between primary and sideband interfaces.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: C700h



Bit Range	Default and Access	Field Name (ID): Description
31:16	0h RO	Reserved.

Bit Range	Default and Access	Field Name (ID): Description
15:8	C7h RW	**Reserved Page Register Destination ID (RPRID):** Specifies the IOSF-SB destination ID for sending Reserved Page Register cycles (e.g., Port 80h). By default this will load to the ID of the LPC or eSPI device depending on which has been strapped active in the system.
7:1	0h RO	Reserved.
0	0h RW	**RTC Shadow Enable (RSE):** When set, all I/O writes to the RTC will be also sent to the PMC. This allows cases where the battery backed storage is in an external PMIC.

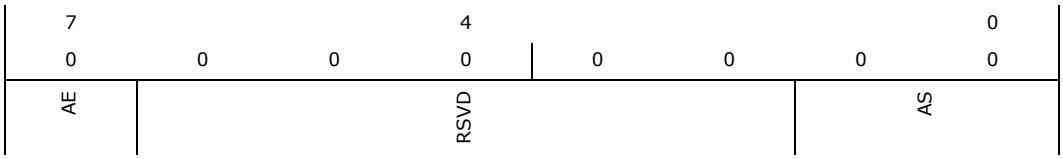
36.9.14 High Performance Event Timer Configuration (HPTC)—Offset 60h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
7	0h RW	**Address Enable (AE):** When set, the P2SB will decode the High Performance Timer memory address range selected by bits [1:0] below.
6:2	0h RO	Reserved.
1:0	0h RW	**Address Select (AS):** This 2-bit field selects 1 of 4 possible memory address ranges for the High Performance Timer functionality. The encodings are: 00 : FED0_0000h - FED0_03FFFh 01 : FED0_1000h - FED0_13FFFh 10 : FED0_2000h - FED0_23FFFh 11 : FED0_3000h - FED0_33FFFh

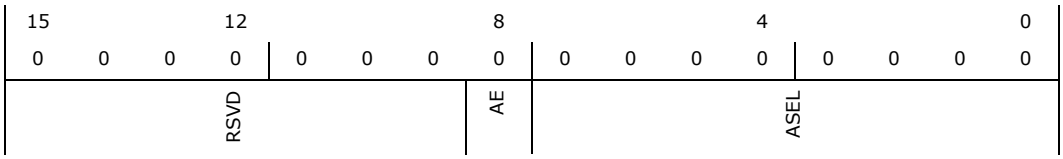
36.9.15 IOxAPIC Configuration (IOAC)—Offset 64h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:9	0h RO	Reserved.
8	0h RW	**Address Enable (AE):** When set, the P2SB will decode the IOxAPIC memory address range selected by bits 7:0 below.
7:0	0h RW	**APIC Range Select (ASEL):** These bits define address bits 19:12 for the IOxAPIC range. The default value of 00h enables compatibility with prior products as an initial value. This value must not be changed unless the IOxAPIC Enable bit is cleared.

36.9.16 IOxAPIC Bus:Device:Function (IBDF)—Offset 6Ch

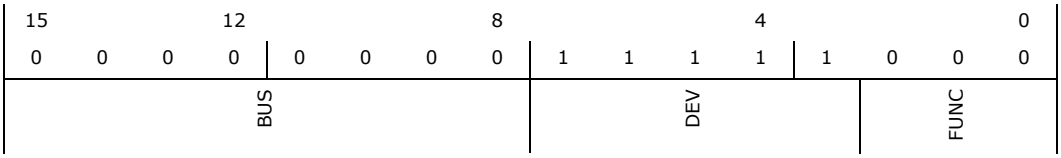
This register specifies the bus:device:function ID that the IOxAPIC will use in the following: As the Requester ID when initiating Interrupt Messages to the CPU As the Completer ID when responding to the reads targeting the IOxAPICs Memory-Mapped I/O registers This field is default to Bus 0: Device 31: Function 0 after reset. BIOS shall program this field accordingly if unique bus:device:function number is required for the internal IOxAPIC.

Access Method

Type: CFG Register (Size: 16 bits)

Device: 31
Function: 1

Default: F8h



Bit Range	Default and Access	Field Name (ID): Description
15:8	0h RW	**Bus Number (BUS)**
7:3	1Fh RW	**Device Number (DEV)**
2:0	0h RW	**Function Number (FUNC)**

36.9.17 HPET Bus:Device:Function (HBDF)—Offset 70h

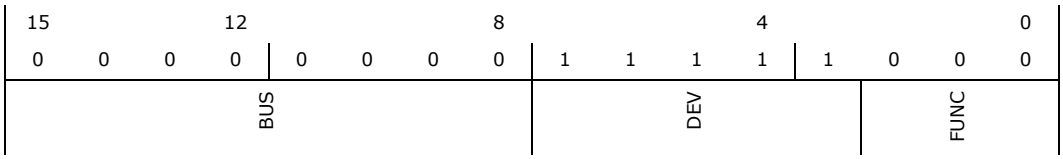
This register specifies the bus:device:function ID that the HPET device will use in the following: As the Requester ID when initiating Interrupt Messages to the CPU As the Completer ID when responding to the reads targeting the corresponding HPETs Memory-Mapped I/O registers This field is default to Bus 0: Device 31: Function 0 after reset. BIOS shall program this field accordingly if unique bus:device:function number is required for the corresponding HPET.

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 1

Default: F8h



Bit Range	Default and Access	Field Name (ID): Description
15:8	0h RW	**Bus Number (BUS)**
7:3	1Fh RW	**Device Number (DEV)**
2:0	0h RW	**Function Number (FUNC)**

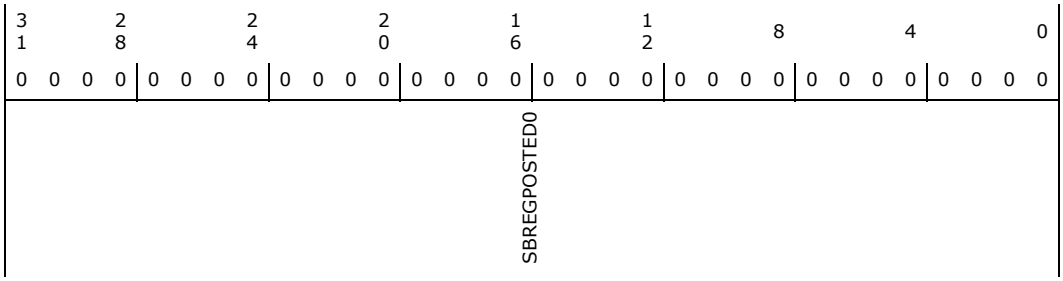
36.9.18 Sideband Register Posted 0 (SBREGPOSTED0)—Offset 80h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Sideband Register posted 0 (SBREGPOSTED0):** One hot masks for setting SBREG to posted for IOSF-SB endpoint IDs 31-0.

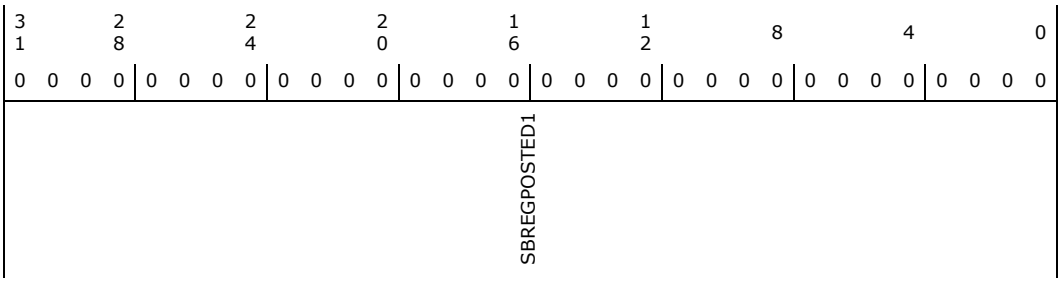
36.9.19 Sideband Register Posted 1 (SBREGPOSTED1)—Offset 84h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Sideband Register posted 1 (SBREGPOSTED1):** One hot masks for setting SBREG to posted for IOSF-SB endpoint IDs 63-32.

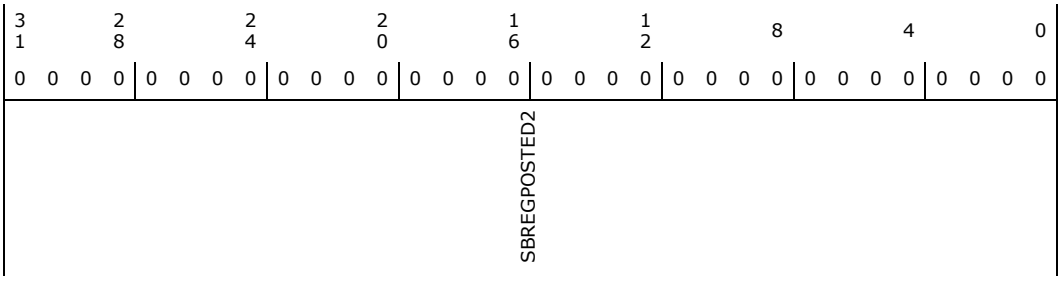
36.9.20 Sideband Register Posted 2 (SBREGPOSTED2)—Offset 88h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Sideband Register posted 0 (SBREGPOSTED2):** One hot masks for setting SBREG to posted for IOSF-SB endpoint IDs 95-64.

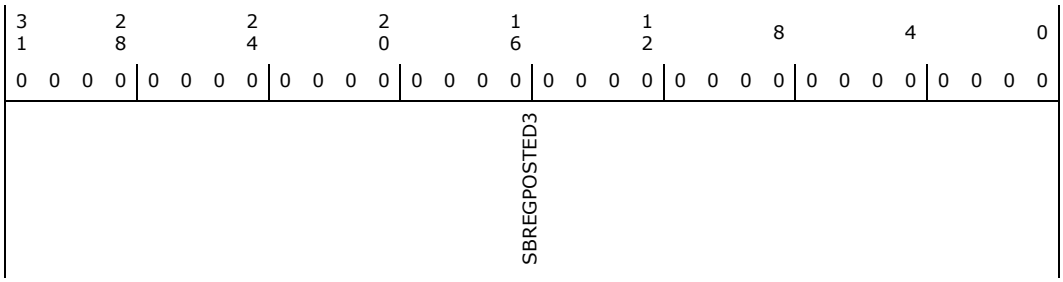
36.9.21 Sideband Register Posted 3 (SBREGPOSTED3)—Offset 8Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Sideband Register posted 3 (SBREGPOSTED3):** One hot masks for setting SBREG to posted for IOSF-SB endpoint IDs 127-96.

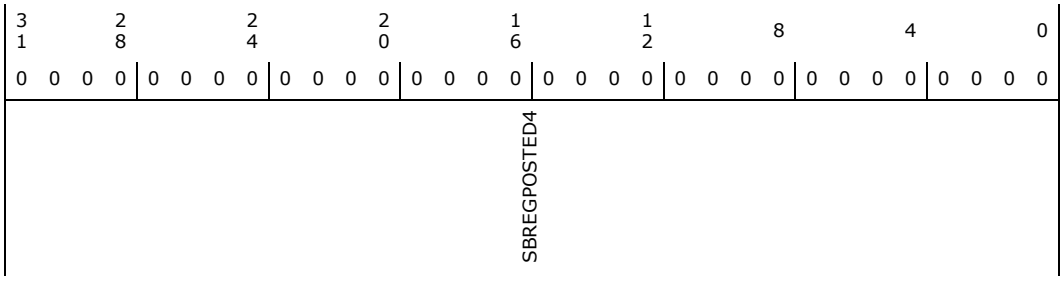
36.9.22 Sideband Register Posted 4 (SBREGPOSTED4)—Offset 90h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Sideband Register posted 4 (SBREGPOSTED4):** One hot masks for setting SBREG to posted for IOSF-SB endpoint IDs 159-128.

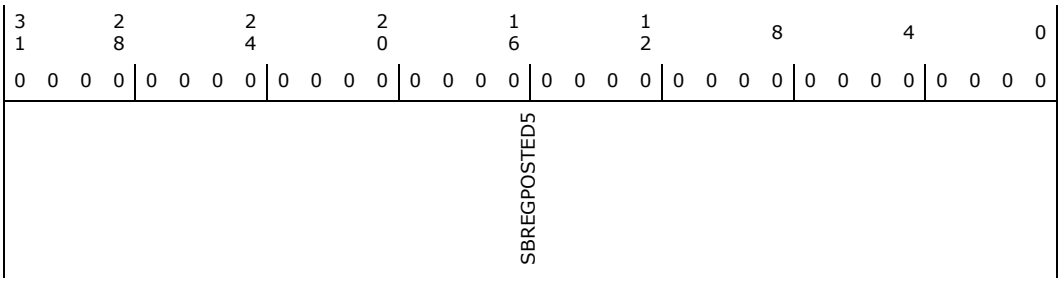
36.9.23 Sideband Register Posted 5 (SBREGPOSTED5)—Offset 94h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Sideband Register posted 5 (SBREGPOSTED5):** One hot masks for setting SBREG to posted for IOSF-SB endpoint IDs 191-160.

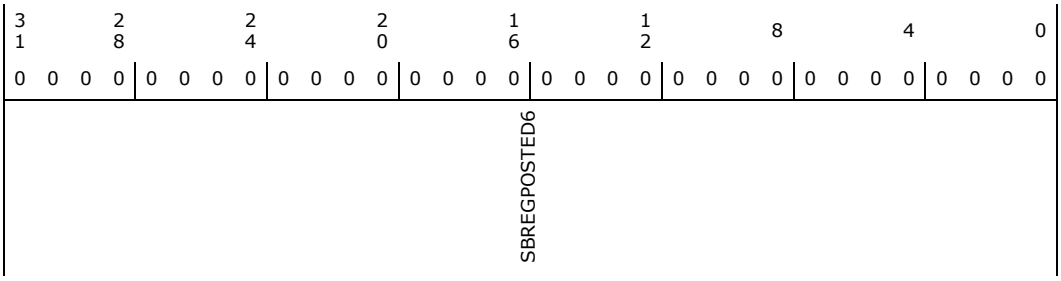
36.9.24 Sideband Register Posted 6 (SBREGPOSTED6)—Offset 98h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Sideband Register posted 6 (SBREGPOSTED6):** One hot masks for setting SBREG to posted for IOSF-SB endpoint IDs 223-192.

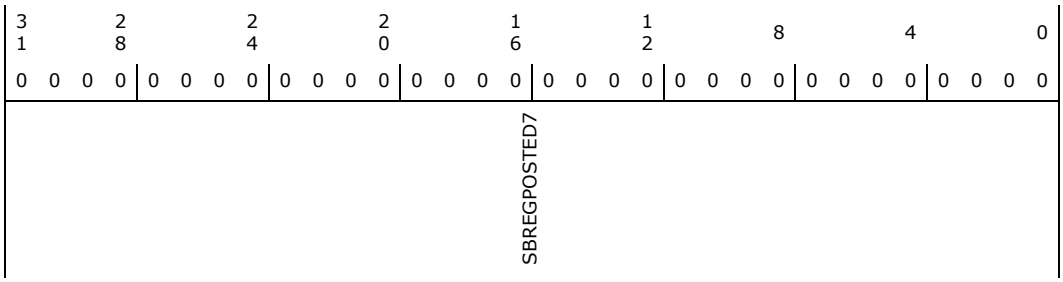
36.9.25 Sideband Register Posted 7 (SBREGPOSTED7)—Offset 9Ch

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW	**Sideband Register posted 7 (SBREGPOSTED7):** One hot masks for setting SBREG to posted for IOSF-SB endpoint IDs 255-224.

36.9.26 Display Bus:Device:Function (DISPBDF)—Offset A0h

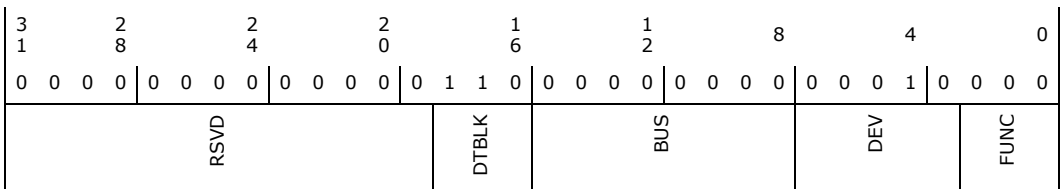
This register specifies the bus:device:function ID that the Display initiated upstream RAVDMs will use for its Requester ID. This will also be used for the claiming these Route-by-ID RAVDMs downstream.

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: 60010h



Bit Range	Default and Access	Field Name (ID): Description
31:19	0h RO	Reserved.
18:16	6h RW	**Display Target Block (DTBLK):** This register contains the Target BLK field that will be used when sending RAVDM messages to the CPU Complex North Display.
15:8	0h RW	**Bus Number (BUS):** The bus number of the Display in the CPU Complex.
7:3	2h RW	**Device Number (DEV):** The bus number of the Display in the CPU Complex.
2:0	0h RW	**Function Number (FUNC):** The function number of the Display in the CPU Complex

36.9.27 ICC Register Offsets (ICCOS)—Offset A4h

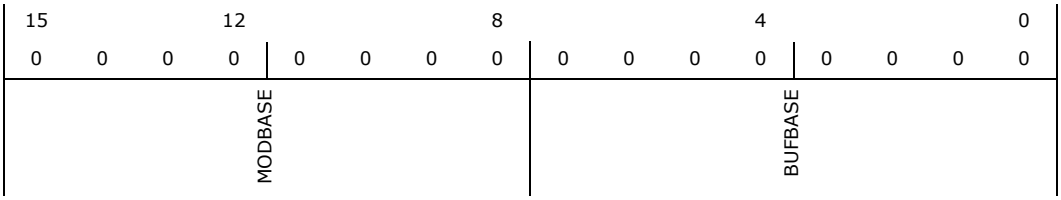
This register contains the offsets to be used when sending RAVDMs to the Integrated Clock Controller. Each of the two spaces decoded for the ICC have a separate base address that will be used when sending those transactions on IOSF-SB to the ICC.

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:8	0h RW	**Modulator Control Address Offset (MODBASE):** This specifies the upper 8b for the 16b address that will be used for sending RAVDM access that target the Modulator Control range of the ICC (FFF00h - FFFFFh).
7:0	0h RW	**Buffer Address Offset (BUFBASE):** This specifies the upper 8b for the 16b address that will be used for sending RAVDM access that target the Buffer range of the ICC (FFE00h - FFEFFh).

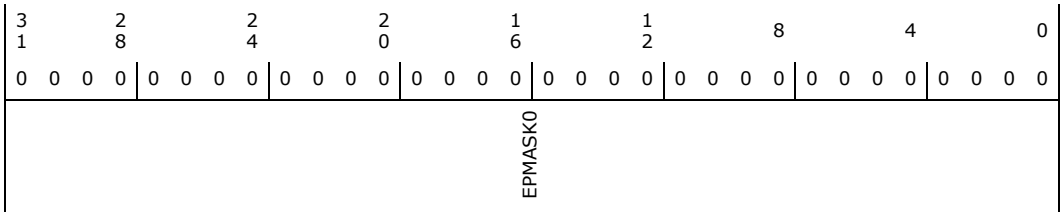
36.9.28 Endpoint Mask 0 (EPMASK0)—Offset B0h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/L	**Endpoint Mask 0 (EPMASK0):** One hot masks for disabling IOSF-SB endpoint IDs 31-0.

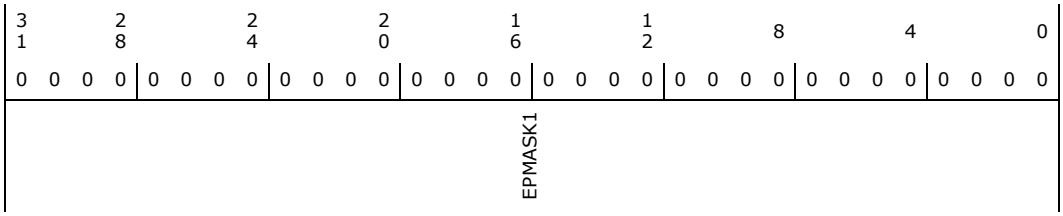
36.9.29 Endpoint Mask 1 (EPMASK1)—Offset B4h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/L	**Endpoint Mask 1 (EPMASK1):** One hot masks for disabling IOSF-SB endpoint IDs 63-32.

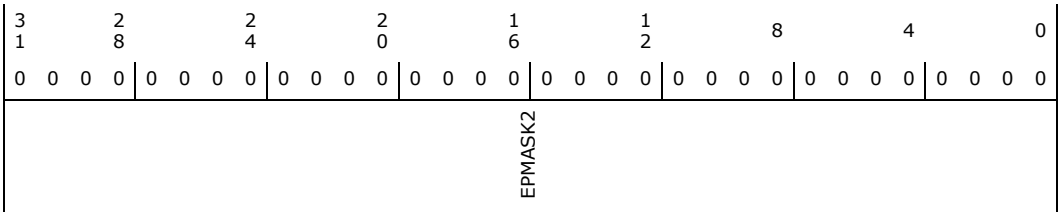
36.9.30 Endpoint Mask 2 (EPMASK2)—Offset B8h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/L	**Endpoint Mask 2 (EPMASK2):** One hot masks for disabling IOSF-SB endpoint IDs 95-64

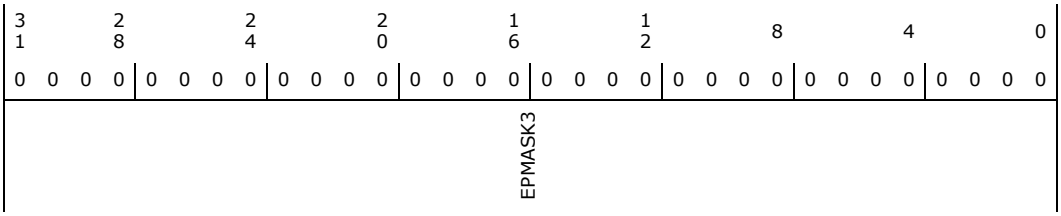
36.9.31 Endpoint Mask 3 (EPMASK3)—Offset BCh

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/L	**Endpoint Mask 3 (EPMASK3):** One hot masks for disabling IOSF-SB endpoint IDs 127-96

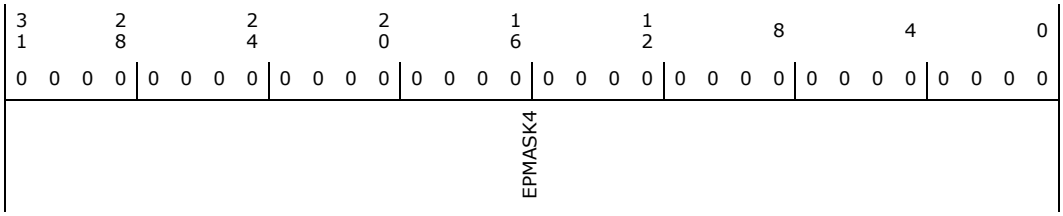
36.9.32 Endpoint Mask 4 (EPMASK4)—Offset C0h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/L	**Endpoint Mask 4 (EPMASK4):** One hot masks for disabling IOSF-SB endpoint IDs 128-159

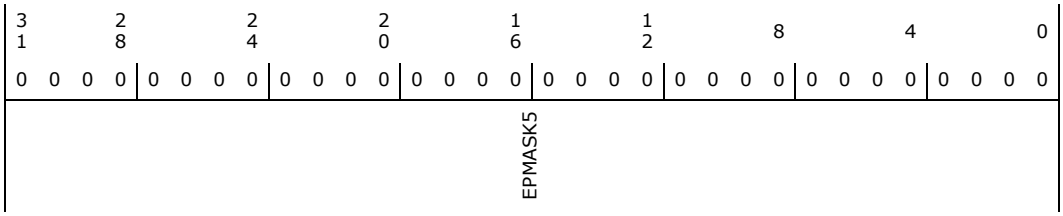
36.9.33 Endpoint Mask 5 (EPMASK5)—Offset C4h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/L	**Endpoint Mask 5 (EPMASK5):** One hot masks for disabling IOSF-SB endpoint IDs 191-160

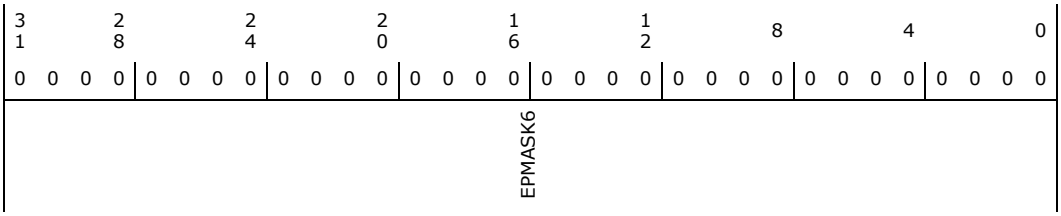
36.9.34 Endpoint Mask 6 (EPMASK6)—Offset C8h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/L	**Endpoint Mask 6 (EPMASK6):** One hot masks for disabling IOSF-SB endpoint IDs 223-192

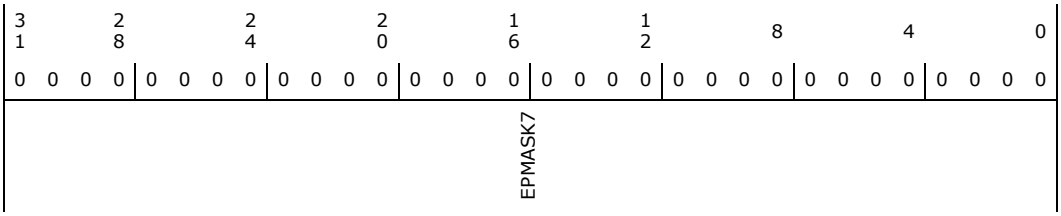
36.9.35 Endpoint Mask 7 (EPMASK7)—Offset CCh

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/L	**Endpoint Mask 7 (EPMASK7):** One hot masks for disabling IOSF-SB endpoint IDs 255-224

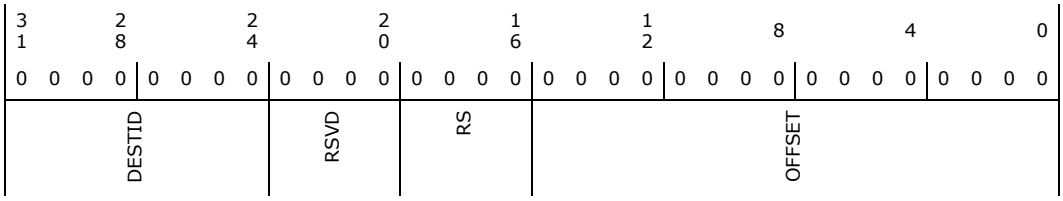
36.9.36 SBI Address (SBIADDR)—Offset D0h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:24	0h RW	**Destination Port ID (DESTID):** The content of this register field is sent in the IOSF Sideband Message Register Access dest field.
23:20	0h RO	Reserved.
19:16	0h RW	**Root Space (RS):** Destination IOSF-SB Root Space. ***Note:*** This register may only be written during manufacturing test. P2SB will only accept writes to this register from transactions with a SAI equal to the SBI_RS_ACCESS_SAI parameter. This should be assigned to the SAI used by the functional test module (typically TAM) that will perform this register write on IOSF-P.
15:0	0h RW	**Address Offset (OFFSET):** Register address offset. The content of this register field is sent in the IOSF Sideband Message Register Access address(15:0) field.

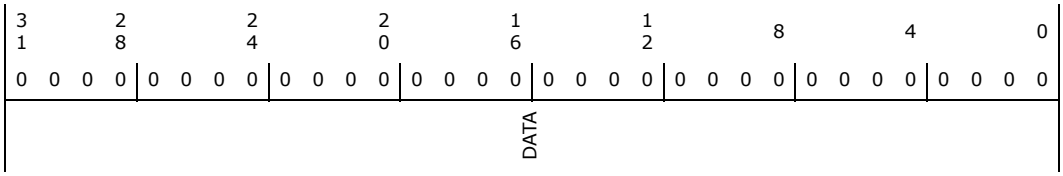
36.9.37 SBI Data (SBIDATA)—Offset D4h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/V	**Data (DATA):** The content of this register field is sent on the IOSF sideband Message Register Access data(31:0) field.

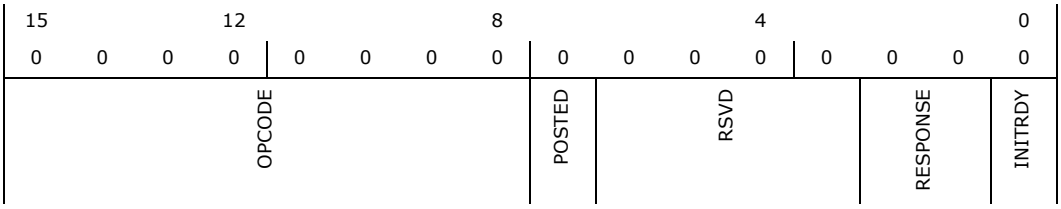
36.9.38 SBI Status (SBISTAT)—Offset D8h

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:8	0h RW	**Opcode (OPCODE):** This is the Opcode sent in the IOSF sideband message.
7	0h RW	**Posted (POSTED):** When set to 1, the message will be send as a posted message instead of non-posted. This should only be used if the receiver is known to support posted operations for the specified operation.
6:3	0h RO	Reserved.
2:1	0h RO/V	**Response Status (RESPONSE):** 00 - Successful 01 - Unsuccessful / Not Supported 10 - Powered Down 11 - Multi-cast Mixed This register reflects the response status for the previously completed transaction. The value of this register is only meaningful if SBISTAT.INITRDY is zero.
0	0h RW/V/L	**Initiate/ Ready# (INITRDY):** 0: The IOSF sideband interface is ready for a new transaction 1: The IOSF sideband interface is busy with the previous transaction. A write to set this register bit to 1 will trigger an IOSF sideband message on the private IOSF sideband interface. The message will be formed based on the values programmed in the Sideband Message Interface Register Access registers. Software needs to ensure that the interface is not busy (SBISTAT.INITRDY is clear) before writing to this register.

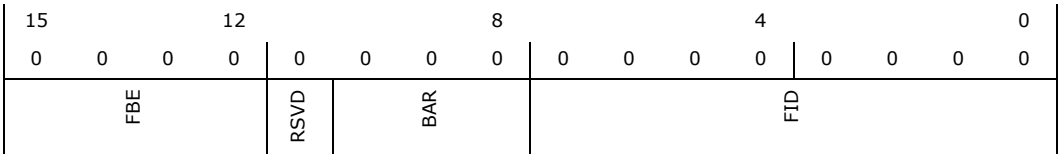
36.9.39 SBI Routing Identification (SBIRID)—Offset DAh

Access Method

Type: CFG Register
(Size: 16 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
15:12	0h RW	**First Byte Enable (FBE):** The content of this field is sent in the IOSF Sideband Register Access FBE field.
11	0h RO	Reserved.
10:8	0h RW	**Base Address Register (BAR):** The contents of this field are sent in the IOSF Sideband Register Access BAR field. This should be zero performing a Memory Mapped operation to a PCI compliant device.
7:0	0h RW	**Function ID (FID):** The contents of this field are sent in the IOSF Sideband Register access FID field. This field should generally remain at zero unless specifically required by a particular application.

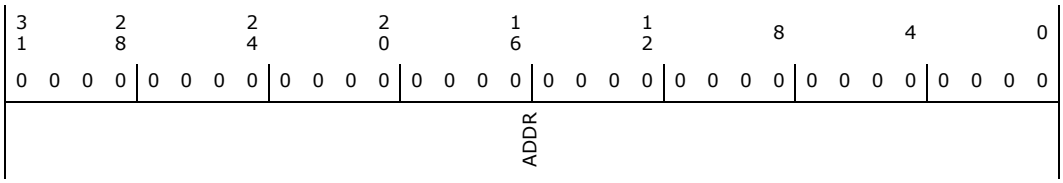
36.9.40 SBI Extended Address (SBIEXTADDR)—Offset DCh

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:0	0h RW/V	**Extended Address (ADDR):** The content of this register field is sent on the IOSF sideband Message Register Access address(48:32) field. This must be set to all 0 if 16b addressing is desired.

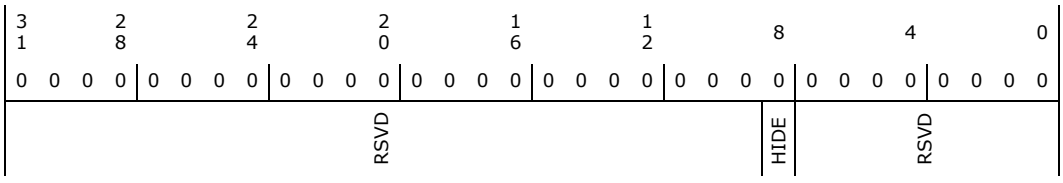
36.9.41 P2SB Control (P2SBC) —Offset E0h

Access Method

Type: CFG Register
(Size: 32 bits)

Device: 31
Function: 1

Default: 0h



Bit Range	Default and Access	Field Name (ID): Description
31:9	0h RO	**Reserved (RSVD)**
8	0b RW	**Device Hide(HIDE):** When this bit is set, the P2SB will return 1s on any PCI Configuration Read on IOSF-P. All other transactions, including PCI Configuration writes, are unaffected by this. This does not affect reads performed on the IOSF-SB interface.
7:0	0h RO	**Reserved (RSVD)**

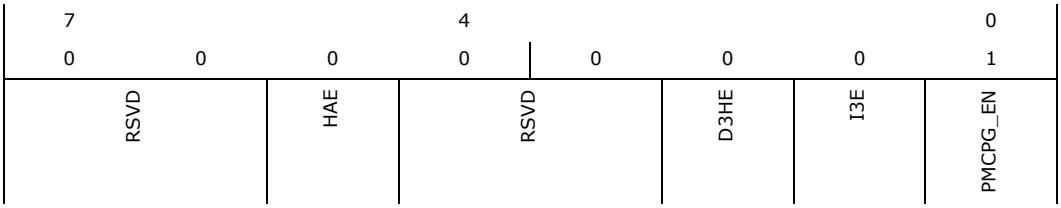
36.9.42 Power Control Enable (PCE)—Offset E4h

Access Method

Type: CFG Register
(Size: 8 bits)

Device: 31
Function: 1

Default: 1h



Bit Range	Default and Access	Field Name (ID): Description
7:6	0h RO	Reserved.
5	0h RW	**Hardware Autonomous Enable (HAE):** When set, the P2SB will automatically engage power gating when it has reached its idle condition.
4:3	0h RO	Reserved.
2	0h RO	**D3-Hot Enable (D3HE):** No support for $D3_{HOT}$ power gating.
1	0h RO	**I3 Enable (I3E):** No support for S0i3 power gating.
0	1h RW	**PMC Power Gating Enable (PMCPG_EN):** When set to 1, the P2SB will engage power gating if it is idle and the pmc_p2sb_sw_pg_req_b signal is asserted.

36.10 PHY Tuning Registers

This section provides a reference for the tuning registers for the USB2 PHYs and the HSIO PHYs. The HSIO PHYs encompass PCIe* 1/2/3, SATA 1.5/3.0/6.0 Gbps, and USB3. The actual values to be programmed into these registers are not in this document. **A separate document, the *Intel® C620 Series Chipset HSIO Tuning Guide*, will provide the details on what the registers do and values to be programmed into them**. The PhyTune Compliance tool will access these registers, too.

36.10.1 USB 2.0 Additional Electrical Control Registers

The following "USB 2.0 Additional Registers" are distributed within the PCH Private Configuration Space which is accessible through the PCH Sideband Interface using the following Target Port (Destination Port) Identification:

— Target Port (Destination Port) Identification = 0xCA

Note: For complete details on how to use the PCH Sideband Interface to access these PCH Private Configuration Registers reference the latest *Intel® C620 Series Chipset Platform Controller Hub BIOS Specification*.

36.10.2 USB2 PER PORT 1 Electrical Control Register(USB2PP1)

Offset Address:

Port 1: CA4100h
Port 2: CA4200h
Port x: CA4x00h

BIOS Reference Code Platform Configuration PCH Policy Details:

- File = PchPolicyCommon.h Release or Greater
- USB2 HSIO Structures
  - — Bit[14] = PERPORTTXPEHALF
  - — Bits[13:11] = PERPORTPETXISET
  - — Bits[10:8] = PERPORTTXISET

Note: These registers are located in the suspend well.

Bit	Description
31:15	Reserved
14	**Per Port Half Bit Pre-emphasis - R/W** Configuration bit (per port) to select between half-bit or full-bit implementation — 1 - select half-bit pre-emphasis — 0 - select full-bit pre-emphasis ***Note:*** Refer to Section 36.2.3 to configure these bits.
13:11	**Per Port HS Pre-emphasis bias - R/W** ***Note:*** Refer to Section 36.2.3 to configure these bits.
10:8	**Per Port HS TX Bias - R/W** ***Note:*** Refer to Section 36.2.3 to configure these bits.
7:0	Reserved

36.10.3 USB2 PER PORT 2 Electrical Control Register (USB2PP2)

Offset Address:

Port 1: CA4126h
Port 2: CA4226h
Port x: CA4x26h

BIOS Reference Code Platform Configuration PCH Policy Details:

- File = PchPolicyCommon.h Release or Greater
- USB2 HSIO Structure
  - — Bits[24:23] = IUSBTXEMPHASISEN

Note: These registers are located in the suspend well.

Bit	Description
31:25	Reserved
24:23	**Per Port HS TX Emphasis - R/W** Enables the HS Tx Emphasis for the respective port ***Note:*** Refer to Section 36.2.3 to configure these bits.
22:0	Reserved

36.10.4 High Speed I/O (HSIO) Modular Physical Layer (ModPHY)

The Intel® C620 Series Chipset Platform Controller Hub ModPHY architecture handles the HSIO Physical Layer Data Receiver (RX) and Data Transmitter (TX) protocol and electrical signaling requirements where its modular capabilities allow the HSIO lanes to be statically mapped to different controllers using the Flexible I/O technology. It also contains the Internal Clock Phase Lock Loop (PLL) Registers; which apply to the USB3Gen2PCIe PLL.

The Flexible I/O technology allows the HSIO lanes to be configured for connection to the PCIe* Gen 1/2/3, SATA 1.5/3.0/6.0 Gb/s, or USB 3.0 controllers. The ModPHY architecture in conjunction with the Flexible I/O technology enables customers to optimize the allocation of HSIO interfaces to better meet the I/O needs of their system.

Table 36-12. Intel® C620 Series Chipset PCH SBREG_BAR and SBI Address Offset Details (Sheet 1 of 2)

Flex HSIO Lane #	Interface		Target Port ID	High Speed I/O (HSIO) Control Register Offset Address - DWORD									
EDS	Mode Type	Lane #		TX DWORD 5	TX DWORD 6	TX DWORD 8	TX DWORD 19	RX DWORD 7[1]	RX DWORD 20	RX DWORD 21	RX DWORD 25	RX DWORD 40	RS DWORD 41[2]
0	USB	1	0xE9	0x0094	0x0098	0x00A0	0x00CC	n/a	0x0150	0x0154	0x0164	0x01A0	0x01A4
1	USB	2	0xE9	0x0294	0x0298	0x02A0	0x02CC	n/a	0x0350	0x0354	0x0364	0x03A0	0x3A4
2	USB	3	0xE9	0x0494	0x0498	0x04A0	0x04CC	N/A	0x0550	0x0554	0x0564	0x05A0	0X05A4
3	USB	4	0xE9	0x0694	0x0698	0x06A0	0x06CC	N/A	0x0750	0x0754	0x0764	0x07A0	0x07A4
4	USB	5	0xE9	0x0894	0x0898	0x08A0	0x08CC	N/A	0x0950	0x0954	0x0964	0x09A0	0x09A4
5	USB	6	0xE9	0x0A94	0x0A98	0x0AA0	0x0ACC	N/A	0x0B50	0x0B54	0x0B64	0x0BA0	0x0BA4
6	USB/PCIe*	7/1	0xE9	0x0C94	0x0C98	0x0CA0	0x0CCC	0x0D1C	0x0D50	0x0D54	0x0D64	0x0DA0	0x0DA4
7	USB/PCIe*	8/2	0xE9	0x0E94	0x0E98	0x0EA0	0x0ECC	0x0F1C	0x0F50	0x0F54	0x0F64	0x0FA0	0x0FA4

Table 36-12. Intel® C620 Series Chipset PCH SBREG_BAR and SBI Address Offset Details (Sheet 2 of 2)

Flex HSIO Lane #	Interface		Target Port ID	High Speed I/O (HSIO) Control Register Offset Address - DWORD									
EDS	Mode Type	Lane #		TX DWORD 5	TX DWORD 6	TX DWORD 8	TX DWORD 19	RX DWORD 7[1]	RX DWORD 20	RX DWORD 21	RX DWORD 25	RX DWORD 40	RS DWORD 41[2]
8	USB/PCIe*	9/3	0xE9	0x1094	0x1098	0x10A0	0x10CC	0x111C	0x1150	0x1154	0x1164	0x11A0	0x11A4
9	USB/PCIe*	10/4	0xE9	0x1294	0x1298	0x12A0	0x12CC	0x131C	0x1350	0x1354	0x1364	0x13A0	0x13A4
10	PCIe*	5	0xA9	0x0094	0x0098	0x00A0	0x00CC	0x011C	0x0150	0x0154	0x0164	0x01A0	0x01A4
11	PCIe*	6	0xA9	0x0294	0x00298	0x02A0	0x02CC	0x031C	0x0350	0x0354	0x0364	0x03A0	0x3A4
12	PCIe/ sSATA*	7/0	0xA9	0x0494	0x0498	0x04A0	0x04CC	0x051C	0x0550	0x0554	0x0564	0x05A0	0X05A4
13	PCIe/ sSATA*	8/1	0xA9	0x0694	0x0698	0x06A0	0x06CC	0x071C	0x0750	0x0754	0x0764	0x07A0	0x07A4
14	PCIe*/ sSATA	9/2	0xA9	0x0894	0x0898	0x08A0	0x08CC	0x091C	0x0950	0x0954	0x0964	0x09A0	0x09A4
15	PCIe*/ sSATA	10/3	0xA9	0x0A94	0x0A98	0x0AA0	0x0ACC	0x0B1C	0x0B50	0x0B54	0x0B64	0x0BA0	0x0BA4
16	PCIe*/ sSATA	11/4	0xA9	0x0C94	0x0C98	0x0CA0	0x0CCC	0x0D1C	0x0D50	0x0D54	0x0D64	0x0DA0	0x0DA4
17	PCIe*/ sSATA	12/5	0xA9	0x0E94	0x0E98	0x0EA0	0x0ECC	0x0F1C	0x0F50	0x0F54	0x0F64	0x0FA0	0x0FA4
18	PCIe*/ SATA/ Uplink	13/0	0xA9	0x1094	0x1098	0x10A0	0x10CC	0x111C	0x1150	0x1154	0x1164	0x11A0	0x11A4
19	PCIe*/ SATA/ Uplink	14/1	0xA9	0x1294	0x1298	0x12A0	0x12CC	0x131C	0x1350	0x1354	0x1364	0x13A0	0x13A4
20	PCIe*/ SATA/ Uplink	15/2	0xA9	0x1494	0x1498	0x14A0	0x14CC	0x151C	0x1550	0x1554	0x1564	0x15A0	0x15A4
21	PCIe*/ SATA/ Uplink	16/3	0xA9	0x1694	0x1698	0x16A0	0x16CC	0x171C	0x1750	0x1754	0x1764	0x17A0	0x17A4
22	PCIe*/ SATA/ Uplink	17/4	0xA9	0x1894	0x1898	0x18A0	0x18CC	0x191C	0x1950	0x1954	0x1964	0x19A0	0x19A4
23	PCIe*/ SATA/ Uplink	18/5	0xA9	0x1A94	0x1A98	0x1AA0	0x1ACC	0x1B1C	0x1B50	0x1B54	0x1B64	0x1BA0	0x1BA4
24	PCIe*/ SATA/ Uplink	19/6	0xA9	0x1C94	0x1C98	0x1CA0	0x1CCC	0x1D1C	0x1D50	0x1D54	0x1D64	0x1DA0	0x1DA4
25	PCIe*/ SATA/ Uplink	20/7	0xA9	0x1E94	0x1E98	0x1EA0	0x1ECC	0x1F1C	0x1F50	0x1F54	0x1F64	0x1FA0	0x1FA4

Notes: This chart has PCIe root ports only, no PCIe uplink is included.

§

37 Debug Registers

37.1 Introduction

The following registers are accessed with the Debug tool provided through SPS FW and C scripts. They may or may not be described in other sections of this document or other documents, but are combined here in one chapter for decode reference. As they are accessed through the SPS interface, the location of these registers is not visible so there's no field for location.

37.2 Register Summary

Table 37-1. Summary of Debug Registers

Register Name (ID)—	Default Value
IP_STS	00h
Link_STS	00h
PREP_STS	00h
IE_STS	00h
HOST_PR_CAUSE_0	00h
HOST_PR_CAUSE_1	00h
HOST_PR_CAUSE_2	00h
HOST_RSTOUT	00h
SUS_SIG_MON_0	F8h
SUS_SIG_MON_1	03h
SUS_SIG_MON_2	40h
SUS_SIG_MON_3	00h
INIT_PWR_OK	00h
PMC Soft Strap 1	06h
PMC Soft Strap 2	10h
PMC Soft Strap 3	80h
PMC Soft Strap 11	0Ah
PMC Soft Strap 12	40h
GLBLRST_CAUSE_3	00h
FW_GLBLRST_CAUSE_0	00h
FW_GLBLRST_CAUSE_1	00h
GLBLRST_CAUSE_0	00h
GLBLRST_CAUSE_1	00h
GLBLRST_CAUSE_2	00h
CORE_SIG_MON_0	9Eh
CORE_SIG_MON_1	00h
MR_GENCTLA	00h
MR_GENCTLC	00h
STATE_TRANS_DATA	00h

Table 37-1. Summary of Debug Registers (Continued)

Register Name (ID)—	Default Value
SLPOUT	00h
SLPOUT2	00h
DSW_WAKE_STS	00h
DSW_PM_FUSE_MON_0	18h

37.2.1 IP_STS

7			4				0
0	0	0	0	0	0	0	0
RSVD						SBI_BP_RP_ALL1_STS	IP_READY_ALL1_STS

Bit Range	Default Value	Field Name (ID): Description
7:2	00h	**Reserved**
1	0b	**SBI BootPrep/ResetPrep All 1 Status (SBI_BP_RP_ALL1_STS)** This bit is set to '1' when all the enabled bits in the SBI_BP_RP_ACK_STS_* registers are '1'. The corresponding enables are in the IP_STS_MASK_* set of registers.
0	0b	**IP_READY All 1 Status (IP_READY_ALL1_STS)** This bit is set to '1' when all the enabled bits in the IP_READY_STS_* registers are '1'. The corresponding enables are in the IP_STS_MASK_* set of registers.

37.2.2 Link_STS

7			4	3			0
0	0	0	0	0	0	0	0
RSVD			DO_RESETB_IN_STS	RESET_DON_IN_STS	DMI_L23_STS	EVA_23_STS	SP_23_STS

Bit Range	Default Value	Field Name (ID): Description
7-5	000b	**Reserved**
4	0b	**DO_RESET_IN# status (DO_RESETB_IN_STS)** This bit is set to ‘1’ by hardware with the DO_RESET_IN# pin transitions from ‘1’ to ‘0’.

Bit Range	Default Value	Field Name (ID): Description
3	0b	**Reset Done In Status (RESET_DONE_IN_STS)** This bit is set to ‘1’ by hardware when the RESET_DON_IN pin transitions from ‘0’ to ‘1’.
2	0b	**DMI L2/L3 Status (DMI_L23_STS)** This bit is set to '1' by hardware when DMI is indicating that the link is in the L2/L3 power state (level event).
1	0b	**EVA Port L2/L3 Status (EVA_L23_STS)** This bit is set to '1' by hardware when all EVA PCI Express Ports are indicating that their links are in the L2/L3 power state (level event).
0	0b	**South Port L2/L3 Status (SP_L23_STS)** This bit is set to '1' by hardware when all in the L2/L3 power state (level event).

37.2.3 PREP_STS- Offset 0x000Eh

7			4	3			0
0	0	0	0	0	0	0	0
PERST_STS	RSVD				FRC_PGPOK	RP_PREP_STS	BP_PREP_STS

Bits	Default Value Access	Description
7	0b	PE_RST# Status (PERST_STS) This bit is set to ‘1’ by hardware after PMC has detected a change on PE_RST#.
6:3	0000b	Reserved
2	0b	**ForcePwrGatePOK Status (FRC_PGPOK_STS)** This bit is set to '1' by hardware after PMC has completed sending the ForcePwrGatePOK message to the enabled groups.
1	0b	**Reset Prep Status (RP_PREP_STS)** This bit is set to 1 by hardware when all expected ACK has been received from IP in response for the message sent and message type is Reset Prep.
0	0b	**Boot Prep Status (BP_PREP_STS)** This bit is set to 1 by hardware when all expected ACK has been received from IP in response for the message sent and message type is Boot Prep.

37.2.4 IE_STS

7			4	3			0
0	0	0	0	0	0	0	0
IE_PR_STS	IEWAK_STS	IE_RTSP_MSG_STS	IE_REG_STS	STALL_PMC_IE_HS_STS	RSVD		

Bits	Default Value	Description
7	0b	**IE Partition Reset Status (IE_PR_STS)** This bit is set to '1' when IE FW writes a '1' to the GENCTL."IE-Partition Reset" bit or when PMC sees a rising edge on one or more of the IE error signals which are currently mapped to IE partition reset, not global reset (all are pulse events) or when a IE HW error occurs and is routed to IE partition reset (level event).
6	0b	**IE Wake Status (IEWAK_STS)** This is a shared status bit to flag that an enabled IE wake event has occurred. There are no Event Status registers for this leaf node - the IE wake status and enable registers (typically IEWS and IEWE) serve this purpose - see IEWAK_EE_* for details. The corresponding event enable bits are located in IEWAK_EE_*. This bit is set to '1' by hardware when any IE status / IE enable / IEWAK_EE_* triplet is active (level event).
5	0b	**IE RTSP Message Received Status (IE_RTSP_MSG_STS)** This is a shared status bit to flag messages that have been received on the IE fabric and root space. The event status bits are located in the MBB_MSG_ES_* registers and the corresponding event enable bits are located in MBB_MSG_EE_*. This bit is set to '1' by hardware when any event status / event enable pair is active (level event).
4	0b	**IE Register Write Status (IE_REG_STS)** This is a shared status bit to flag an IE FW write to an enabled register. The event status bits are located in the IE_REG_ES_* registers and the corresponding event enable bits are located in IE_REG_EE_*. This bit is set to '1' by hardware when any event status / event enable pair is active (level event).
3	0b	**Stall PMC/IE Handshaking Status (STALL_PMC_IE_HS_STS)** This bit will be set to '1' by hardware whenever GENCTL.STALL_PMC_IE_HS is '0' (level event).
2:0	000b	Reserved

37.2.5 HOST_PR_CAUSE_0

7			4	3			0
0	0	0	0	0	0	0	0
IE_HRPD	TCO_WDT	IE_PRPC	IH_HR	RSVD	SYSRST	CF9	DO_RESET_IN_ES

Bits		Description
7	0b RW	**IE-Initiated Host Reset With Power Down (IE_HRPD)** This bit is set when IE FW writes a '1' to GENCTL."IE-Initiated Host Reset With Power Down".
6	0b RW	**Host TCO Watchdog Timer Second Expiration (TCO_WDT)** This bit is set when the host TCO watchdog timer reaches zero for the second time.
5	0b RW	**IE-Initiated Host Reset With Power Cycle (IE_PRPC)** This bit is set when IE FW writes a '1' to GENCTL."IE-Initiated Host Reset With Power cycle".
4	0b RW	**IE-Initiated Host Reset Without Power Cycle (IE_HR)** This bit is set when IE FW writes a '1' to GENCTL."IE-Initiated Host Reset Without Power cycle".
3	0b RW	**Reserved**
2:0	0b RW	**SYS_RST# Assertion (SYSRST)** This bit is set based on an asserting edge of the SYS_RESET# pin (post debounce circuitry)
1	0b RW	**Write to I/O Port CF9 (CF9)** This bit is set when host software writes a value of 06h or 0Eh to the CF9 register.
0	0b RW	**Do Reset In Event Status (DO_RESET_IN_ES)** This bit is set when PMC FW writes a “1” to SUS_MISC_CTL.SET_DO_RESET_IN_HER_ES

37.2.6 HOST_PR_CAUSE_1

7			4	3			0
0	0	0	0	0	0	0	0
GLBL_TO_HOST	RSVD	HSMB_HRPC	HSMB_HR	RSVD	MI_HRPD	MI_PRPC	MI_HR

Bits	Default Value	Description
7	0b	**Global Reset to Host Reset Event Status (GLBL_TO_HOST)** This bit is set when the PMC converts a global reset to host partition reset due to xxxx.GBL2HOST being set when the global reset occurred.
6	0b	**RSVD**
5	0b	**Host SMBus Host Reset With Power Cycle (HSMB_HRPC)** This bit is set when the PMC writes a '1' to SUS_MISC_CTL.SET_HSMB_HRPC_ES.
4	0b	**Host SMBus Host Reset Without Power Cycle (HSMB_HR)** This bit is set when the PMC writes a '1' to SUS_MISC_CTL.SET_HSMB_HR_ES.
3	0b	**RSVD**
2:0	0b	**ME-Initiated Host Reset With Power Down (MI_HRPD)** This bit is set when Intel ME FW writes a '1' to GENTCTL."ME-Initiated Host reset with Power Down".
1	0b	**ME-Initiated Host Reset With Power Cycle (MI_PRPC)** This bit is set when Intel ME FW writes a '1' to GENTCTL."ME-Initiated Host reset with Power Cycle".
0	0b	**ME-Initiated Host Reset Without Power Cycle (MI_HR)** This bit is set when Intel ME FW writes a '1' to GENTCTL."ME-Initiated Host reset without Power Cycle".

37.2.7 HOST_PR_CAUSE_2

7			4	3			0
0	0	0	0	0	0	0	0
RSVD							AC_RU_HR

Bits	Default Value	Description
7:1	0000000b	**RSVD**
0	0b	**AC RU Initiated Host Reset (AC_RU_HR)** This bit is set when the AC RU requests a Host Reset.

37.2.8 HOST_RSTOUT

7			4	3			0
0	0	0	0	0	0	0	0
RSVD						HOST_SIDE_RST_VAL	HOST_PRIM_RST_VAL

Bits	Default Value	Description
7:2	000000b	**RSVD**
1	0b	**Host Side Reset (HOST_SIDE_RST_VAL)** Value to drive on the IOSF Host Side RST# signal.
0	0b	**HOST_PRIM_RST# Value (HOST_PRIM_RST_VAL)** Value to drive on the Host_Prim_Rst# pin

37.2.9 SUS_SIG_MON_0

7			4	3			0
1	1	1	1	1	0	0	0
RSVD		PWRBTN_PIN	SUSFLR_STS	GP27_PIN	RSVD	SYSPWER_PIN	PCHPWR_PIN

Bit	Default Value	Description
7:6	11b	**RSVD**
5	1b	**PWRBTN# Pin Value (PWRBTN_PIN)** Live value of the PWRBTN# pin, after synchronization to PMC_CLK and any debounce
4	1b	**SUS Well Power Failure Status (SUSFLR_STS)** This bit is set to '1' by hardware when a global reset is triggered by loss of Primary well power. This includes DeepSx and G3.
3	1b	**GP27 Pin Value (GP27_PIN)** Live value of the GP27 pin, after synchronization to the pmc_clk.
2	0b	**RSVD**
1	0b	**SYS_PWROK Pin Value (SYSPWR_PIN)** Live value of the SYS_PWROK pin, after synchronization to pmc_clk.
0	0b	**PCH_PWROK Pin Value (PCHPWR_PIN)** Live value of the PCH_PWROK pin, after isolation to the Primary well and synchronization to pmc_clk.

37.2.10 SUS_SIG_MON_1

7			4	3			0
0	0	0	0	0	0	1	1
FAST_RST	RSVD					PME_pIN	WAKE_pin

Bit	Default Value	Description
7	0b	**Fast Reset Test Mode (FAST_RST)** FW can check this bit to see if the Fast Reset Test Mode is enabled (‘1’ = enabled)
6:2	00000b	**RSVD**
1	1b	**PME# Pin Value (PME_PIN)** Live value of the PME# pine
0	1b	**WAKE# Pin Value (WAKE_PIN)** Live value of the WAKE# pin

37.2.11 SUS_SIG_MON_2

7			4	3			0
0	1	0	0	0	0	0	0
RSVD		XTAL_OSC_VLD	RSVD				PMC_UG_STS

Bit	Default Value	Description
7:6	01b	**RSVD**
5	0b	**Crystal Oscillator Valid (Xtal_Osc_VLD)** If the 48 MHz Crystal Oscillator is operational this bit is 1
4:1	0000	**RSVD**
0	1b	**PMC Ungate Status (PMC_UG_STS)** This bit is to ‘1’ when the PMC exits a power gated state. This bit is the only way to detect a power gate exit and differentiated it’s behavior from GLOBAL_RST# assertion cases. This bit is reset with a Global reset

37.2.12 SUS_SIG_MON_3

7			4	3			0
0	0	0	0	0	0	0	0
RSVD			PERSTB_IN_VAL	DO_RESET_IN_VAL	RESET_DON_IN_VAL	RSVD	

Bits	Default Value	Description
7:5	000b	**RSVD**
4	0b	**PE_RST# Pin Value (PERSTB_IN_VAL)** Live value of the PE_RST# pin after synchronization to the pmc_clk
3	0b	**DO_RESET_IN# Pin Value (DO_RESET_IN_VAL)** Live value of the DO_RESET_IN# pin, after synchronization to pmc_clk.
2	0b	**RESET_DONE_IN Pin Value (RESET_DONE_IN_VAL)** Live value of the RESET_DONE_IN pin, after synchronization to pmc_clk.
1:0	00b	**RSVD**

37.2.13 INIT_PWR_OK

7			4	3			0
0	0	0	0	0	0	0	0
RSVD		INT_CORE_POK	INT_ASW_POK	RSVD	INT_AUX_POK	INT_SUS_POK	RSVD

Bit	Default Value	Description
7:6	00b	**RSVD**
5	0b	**Internal CORE Well Power OK (INT_CORE_POK)** Live value of the internal CORE well power OK indication, after isolation to the Primary well and synchronization to pmc_clk.
4	0b	**Internal AS Well Power OK (INT_ASW_POK)** Live value of the internal "AS well" power OK indication, after isolation to the Primary well and synchronization to pmc_clk.
3	0b	**RSVD**
2	0b	**Internal AUX Well Power OK (INT_AUX_POK)** Live value of the internal AUX well power OK indication, after isolation to the Primary well and synchronization to pmc_clk.
1	0b	**Internal SUS Well Power OK (INT_SUS_POK)** Live value of the internal Primary well power OK indication, after isolation to the Primary well and synchronization to pmc_clk. The primary well requires that the DSW is up. Note that this bit should never be '0' when the PMC is functional

Bit	Default Value	Description
0	0b	**RSVD**

37.2.14 PMC Soft Strap 1

This register reflects what are read in from the PCH Descriptor Record 119, bits [7:0].

7			4	3			0
0	0	0	0	0	1	1	0
RSVD	AUX_PWR_POLICY	THRMTRIP_POINT	ME_HOST_BOOT_PREP_EN	RSVD	SYS_PWROK_POINT	CPWRG_RST_WTDIS	CPWRG_STCH_WTDIS

Bit	Default Value	Description
7	0b	**RSVD**
6	0b	**Aux Well Power Policy (AUX_PWR_POLICY)** 0: Power down the AUX well prior to MBB_RST# deassertion 1: Leave the AUX Well powered
5	0b	**THRMTRIP# Valid Point (THRMTRIP_POINT)** 0: Start monitoring THRMTRIP# at PLTRST# 1: Start monitoring THRMTRIP# AT CPUPWRGD
4	0b	**ME FW Host Boot Preparation Enable (ME_HOST_BOOT_PREP_EN)** 0: Proceed with Host boot without waiting for Intel ME FW 1: Wait for a positive indication from Intel ME FW during the host boot sequence
3	0b	**RSVD**
2	1b	**SYS_PWROK Sync Point (SYS_PWROK_POINT)** 0: Wait for SYS_PWROK before driving CPUWPRGD high 1: Wait for SYS_PWROK before driving SUS_STAT# high. Note that there is no SUS_STAT# pin on the Intel® C620 Series Chipset.
1	1b	**CPUPWRGD-to-RST# WD Timer Disable (CPWRG_RST_WTDIS)** 0: The PCH enables the CPUPWRGD-to-CPURST# rising edge watchdog timer 1: The PCH disabled the CPUPWRGD-to-CPURST# rising edge watchdog timer
0	0b	**CPUPWRGD Stretching WD timer Disable (CPWRG_STCH_WTDIS)** 0: The PCH enables the CPUPWRGD stretching watchdog timer. 1: The PCH disables the CPUPWRGD stretching watchdog timer

37.2.15 PMC Soft Strap 2

This register reflects what are read in from the PCH Descriptor Record 119, bits [15:8].

7			4	3			0
0	0	0	1	0	0	0	0
RSVD	RSVD	SYSPWR_FLR_DIS	EVA_SIG_EN	CPUPWRGD to SUS_STAT		START_ISCLK_ACK to CPUWRGD	

Bit	Default Values	Description
7	0b	**RSVD**
6	0b	**RSVD**
5	0b	**SYS_PWROK Failure Detection Disable (SYSPWR_FLR_DIS)** 0: Enable detection of SYS_PWROK failures 1: Disable detection of SYS_PWROK failures.
4	1b	**RSVD**
3:2	00b	**CPUPWRGD to SUS_STAT** Minimum time from CPUPWRGD assertion to SUS_STAT# deassertion 00: 1 ms 01: 10 ms 10: 5 ms 11: 2 ms ***Note:*** The Intel® C620 Series Chipset PCH does not have a SUS_STAT# pin.
1:0	00b	**START_ISCLK_ACK to CPUWRGD** Min time from PLL lock to CPUPWRGD high. 00: 100 ms 01: 50 ms 10: 5 ms 11: 1 ms

37.2.16 PMC Soft Strap 3

This register reflects what are read in from the PCH Descriptor Record 119, bits [23:16].

7			4	3			0
1	0	0	0	0	0	0	0
RSVD			DEEPSX_PLT_CFG	GbE_PHY_PU_TIME	RSVD		

Bit	Default Value	Description
7:5	100b	**RSVD**
4	0b	**DeepSx Platform Configuration (DEEPSX_PLT_CFG)** This bit reflects whether DeepSX is enabled at the board level 0: The platform does not support DeepSx 1: The platform supports DeepSx
3:	0b	**GbE PHY Power up Time (**GbE_PHY_PU_TIME) This bit determines how long the PMC will wait for the GbE PHY (legacy 1GbE, not the 10/1 LAN) to power up after de-assertion of SLP_GbE# 0: 100 ms 1: 50 ms
2:0	000b	**RSVD**

37.2.17 PMC Soft Strap 11

This register reflects what are read in from the PCH Descriptor Record 121, bits [23:16].

7			4	3			0
0	0	0	0	1	0	1	0
RSVD	PECI Ownership	USB Port 1 Ownership	USB Port 0 Ownership	RSVD	EPO Mode Enable	IE Disable	IE FW Host Boot Prep

Bit	Default Value	Description
7	0b	**RSVD**
6	0b	**PECI Ownership** Selects between Intel ME and IE for ownership of PECI. 0: Intel ME 1: IE

Bit	Default Value	Description
5	0b	**USB Port1 Ownership** Selects between Intel ME and IE for ownership of USB Port1. 0: Intel ME 1: IE
4	0b	**USB Port0 Ownership** Selects between Intel ME and IE for ownership of USB Port0. 0: Intel ME 1: IE
3	1b	**RSVD**
2	0b	**EPO Mode enable** This soft strap allows the device to operate in either EP Only or DMI Mode. 0: DMI is enabled. 1: DMI is disabled and the PCH is in EPO mode.
1:	1b	IE **Disable** Set this bit to "0" to enable IE, otherwise set to "1" to disable/powergate IE. 0: IE Enabled 1: IE Disabled
0	0b	**IE FW Host Boot Prep** 0: Proceed to Host boot without waiting for IE FW. 1: Wait for a positive indication from IE FW during the Host Boot Sequence.

37.2.18 PMC Soft Strap 12

This register reflects what are read in from the PCH Descriptor Record 121, bits [31:24].

7			4	3			0
0	1	0	0	0	0	0	0
RSVD		SYS_RESET# Debounce	WD_TO_Multiplier			RSVD	

Bit	Default Value	Description
7:6	01b	**RSVD**
5	0b	**Sys_Reset# Debounce disable** 0: 16 ms debounce logic on SYS_RESET# is enabled 1: 16 ms debounce logic on SYS_RESET# is disabled

Bit	Default Value	Description
4:2	000b	**WD_TO_Multiplier** These bits are used as a multiplier on the WD timeout, which is nominally 8 seconds. 000: Multiply x1 001: Multiply x2 010: Multiply x4 011: Multiply x 8 100: Multiply x 16 101: Multiply by 32 110: Multiply x 64 111: Disabled
1:0	00b	**RSVD**

37.2.19 GLBLRST_CAUSE_3

7			4	3			0
0	0	0	0	0	0	0	0
IE_GBL	IE_PB0	IE_MAX_UXS_ERR	IE_MIA_UX_ERR	IE_UNCOR_ERR	IE_WDT	AC_RU_ERR	RSVD

Bits		Description
7	0b	**IE-Initiated Global Reset (IE_GBL)** This bit is set to '1' by hardware when a global reset is triggered by an IE FW write of 1's to both GENCTL. "IE-Partition Reset" and GENCTL. "IE-Initiated Host Reset With Power Cycle" in the same write cycle (this is IE FW's method of requesting a global reset).
6	0b	**IE-Initiated Power Button Override (IE_PBO)** This bit is set to '1' by hardware when a global reset is triggered by an IE FW write of '1' to GENCTL."IE-Initiated Power Button Override".
5	0b	**IE Unexpected Error (IE_MIA_UXS_ERR)** This bit is set to '1' by hardware when a global reset is triggered by an unexpected error in the IE.
4	0b	**IE Unexpected Shutdown Error (IE_MIA_UX_ERR)** This bit is set to '1' by hardware when a global reset is triggered by an unexpected shutdown error in the IE.
3	0b	**IE HW Uncorrectable Error (IE_UNCOR_ERR)** This bit is set to '1' by hardware when a global reset is triggered by IE hardware due to the detection of an uncorrectable ECC or parity error on a data read from one of its SRAMs.
2	0b	**IE Firmware Watchdog Timer (IE_WDT)** This bit is set to '1' by hardware when a global reset is triggered by the second expiration of the IE Firmware watchdog timer.
1:	0b	**AC RU Error (AC_RU_ERR)** This bit is set to "1" by hardware when a global reset is triggered by the reset unit of the Intel QAT and 10/1 GbE LAN
0	0b	**RSVD**

37.2.20 FW_GLBLRST_CAUSE_0

7			4	3			0
0	0	0	0	0	0	0	0
RSVD			PMC_3STRIKE	HSMB_MSG	HOST_RST_PROM	SX_ENTRY_TIMEOUT	HOST_RESET_TIMEOUT

Bits	Default Value	Description
7:5	000b	**RSVD**
4	0b	**PMC 3 strike boot failure counter (PMC_3STRIKE)** If this bit is set, the cause of the previous global reset was 3 consecutive global resets during an attempt to boot in conjunction with PM_CFG.GR_TSC_EN being 1.
3	0b	**HOST SMBus Message (HSMB_MSG)** If this bit is set, the cause of the previous global reset was a global reset request received over the host SMBUS.
2	0b	**Host Partition Reset Promotion (HOST_RST_PROM)** If this bit is set, the cause of the previous global reset was a host partition reset that was promoted to a global reset either due to Intel ME or host policy.
1:	0b	**Sx Entry Timeout (SX_ENTRY_TIMEOUT)** If this bit is set, the cause of the previous global reset was an expiration of the timer that runs during Sx entry.
0	0b	**Host Partition Reset Timeout (HOST_RESET_TIMEOUT)** If this bit is set, the cause of the previous global reset was an expiration of the timer that runs during host partition resets.

37.2.21 FW_GLBLRST_CAUSE_1

7			4	3			0
0	0	0	0	0	0	0	0
IE_SET_PBO_STS	RSVD						

Bits	Default Value	Description
7	0b	**IE Set Power Button Status (IE_SET_PBO_STS)** If this bit is set, the cause of the previous global reset was IE FW setting the power button override status.
6:0	0000000b	**RSVD**”

37.2.22 GLBLRST_CAUSE_0

7			4	3			0
0	0	0	0	0	0	0	0
RSVD	ME_GBL	CPU_TRIP	ME_PBO	PCH_CAT_TMP	PMC_UNC_ERR	PB_OVR	RSVD

Bits	Default Value	Description
7	0b	**RSVD**
6	0b	**ME-Initiated Global Reset (ME_GBL)**
5	0b	**CPU Thermal Trip (CPU_TRIP)** This bit is set to '1' by hardware when a global reset is triggered by a CPU thermal trip event (i.e., an assertion of the THRMTRIP# pin).
4	0b	**Intel ME-Initiated Power Button Override (ME_PBO)** This bit is set to '1' by hardware when a global reset is triggered by an Intel ME FW write of '1' to GENCTL. "ME-Initiated Power Button Override".
3	0b	**PCH Catastrophic Temperature Event (PCH_CAT_TMP)** This bit is set to '1 by hardware when a global reset is triggered by a catastrophic temperature event from the ICH internal thermal sensor.
2	0b	**PMC SUS RAM Uncorrectable Error (PMC_UNC_ERR)** This bit is set to '1' by hardware when a global reset is triggered due to an uncorrectable parity error on a data read from one of the PMC Primary well register files.
1	0b	**Power Button Override (PB_OVR)** This bit is set to '1' by hardware when a global reset is triggered by a power button override (i.e., an assertion of the PWRBTN# pin for 5 seconds).
0	0b	**RSVD**

37.2.23 GLBLRST_CAUSE_1

7			4	3			0
0	0	0	0	0	0	0	0
ME_UX_ERR	ME_UXS_ERR	RSVD	SYSPWR_FLR	PCHPWER_FLR	PMC_GBL	ME_WDT	PMC_WDT

Bits	Default Value	Description
7	0b	**Intel ME Unexpected Error (ME_UX_ERR)** Set to '1' when a global reset is triggered by an unexpected error in Intel ME.
6	0b	**Intel ME Unexpected Shutdown Error (ME_UXS_ERR)** Set to '1' when a global reset is triggered by an unexpected shutdown in Intel ME.
5	0b	**RSVD**

Bits	Default Value	Description
4	0b	**SYS_PWROK Failure (SYSPWR_FLR)** This bit is set to '1' by hardware when a global reset is triggered by an unexpected loss of SYS_PWROK. FW arms this global reset source via GBLRST_CTL.EN_SYSPWR_FLR.
3	0b	**PCH_PWROK Failure (PCHPWR_FLR)** This bit is set to '1' by hardware when a global reset is triggered by an unexpected loss of PCH_PWROK. FW arms this global reset source via GBLRST_CTL.EN_PCHPWR_FLR.
2	0b	**PMC Global Reset (PMC_GBL)** This bit is set to '1' by hardware when a global reset is triggered by a request from PMC (i.e., a write of '1' to the GBLRST_CTL.TRIG_GBL bit).
1:	0b	**Intel ME Firmware Watchdog Timer (ME_WDT)** This bit is set to '1' by hardware when a global reset is triggered by the second expiration of the Intel ME firmware watchdog timer.
0	0b	**PMC Watchdog Timer (PMC_WDT)** This bit is set to '1' by hardware when a global reset is triggered by the second expiration of the PMC watchdog timer.

37.2.24 GLBLRST_CAUSE_2

7			4	3			0
0	0	0	0	0	0	0	0
RSVD					ADR_GPIO_RST	ME_UNCOR_ERR	CPU_THRM_WDT

Bits	Default Value	Description
7:3	00000b	**RSVD**
2	0b	**ADR GPIO Reset (ADR_GPIO_RST)** This bit is set to '1' by hardware when a global reset is triggered by the assertion of the GPIO assigned to ADR.
1:	0b	**ME HW Uncorrectable Error (ME_UNCOR_ERR)** This bit is set to '1' by hardware when a global reset is triggered by Intel ME hardware due to the detection of an uncorrectable ECC or parity error on a data read from one of its SRAMs.
0	0b	**CPU Thermal Runaway Watchdog Timer (CPU_THRM_WDT)** This bit is set to '1' by hardware when a global reset is triggered by the expiration of the CPU Thermal Runaway Watchdog Timer.

37.2.25 CORE_SIG_MON_0

7			4	3			0
1	0	0	1	1	1	1	0
SYSRST_PIN	CF9_BIT3	ICC_PLLOFF_SIG	LTRESET_SIG	RSVD		THRMTRIP_PIN	RSVD

Bits	Default Value	Description
7	1b	**SYS_RESET# Pin Value (SYSRST_PIN)** Live value of the SYS_RESET# pin, after synchronization to pmc_clk_core and 16 ms debounce.
6	0b	**Value of Bit 3 of the CF9 Register (CF9_BIT3)** Provides the live value of bit 3 of the CF9 register in Host fixed I/O space. This value of this bit determines how some Host partition reset events should behave (such as CF9 writes and SYS_RESET#).
5	0b	**ICC PLLOFF Signal Value (ICC_PLLOFF_SIG)** Live value of the PLLOFF signal for the ICC PLL after synchronization to pmc_clk_core.
4	1b	**LTRESET Signal Value (LTRESET_SIG)** Live value of the LTRESETsignal, representing a received LTRESET DMI message, after synchronization to pmc_clk_core.
3:2	11	**RSVD**
1:	1b	**THRMTRIP# Pin Value (THRMTRIP_PIN)** Live value of the THRMTRIP# pin, after deglitch and synchronization to pmc_clk_core.
0	0b	**RSVD**

37.2.26 CORE_SIG_MON_1

7			4	3			0
0	0	0	0	0	0	0	0
RSVD						TT_STATE	

Bits	Default Value	Description
7:2	000000b	**RSVD**
1:0	00b	**Thermal Throttle State (TT_STATE)** This field reports the current Thermal Throttle State driven from the Temperature Sensor cluster.

37.2.27 MR_GENCTLA

7			4	3			0
0	0	0	0	0	0	0	0
RSVD			SHAD_CPUWPRGD_OVR	RSVD	SHAD_DIS_HOST_WAK_ASW	SHAD_MSK_HWAKE	RSVD

Bits	Default Value	Description
7:5	000b	**RSVD**
4	0b	**Shadow CPUPWRGD Override (SHAD_CPUPWRGD_OVR)**
3	0b	**RSVD**
2	0b	**Shadow Disable Host Wakes (ASW) (SHAD_DIS_HOST_WAK_ASW)**
1	0b	**Shadow Maskable Host Wake (SHAD_MSK_HWAKE)**
0	0b	**RSVD**

37.2.28 MR_GENCTLC

7			4	3			0
0	0	0	0	0	0	0	0
RSVD			SHAD_STALL_PMC_ME_HS	SHAD_HBPD_OVR	SHAD_PROM_HPR_GR	SHAD_M2H_PCH_SSKU_VALID	SHAD_ME_NPCR_WITH_HOST

Bits	Default Value	Description
7:5	000b	**RSVD**
4	0b	**Shadow Stall PMC/ME Handshaking for S345/Reset Warn (SHAD_STALL_PMC_ME_HS)**
3	0b	**Shadow Host Boot Prep Done Override (SHAD_HBPD_OVR)**
2	0b	**Shadow Promote Host Partition Reset to Global Reset (SHAD_PROM_HPR_GR)**
1	0b	**Shadow ME To Host Indication - PCH Soft SKU Valid (SHAD_M2H_PCH_SSKU_VALID)**
0	0b	**Shadow ME Non-Power Cycle Reset With Host (SHAD_ME_NPCR_WITH_HOST)**

37.2.29 STATE_TRANS_DATA

7			4	3			0
0	0	0	0	0	0	0	0
RSVD				DRV_STATE_TRANS_DATA			

Bits	Default Value	Description
7:4	0000b	**RSVD**
3:0	0000b	**Drive State Transition Data (DRV_STATE_TRANS_DATA)** Value to drive on the corresponding Intel ME bits.

37.2.30 SLPOUT

7			4	3			0
0	0	0	0	0	0	0	0
RSVD				SLP_A_VAL	SLP_S5_VAL	SLP_S4_VAL	SLP_S3_VAL

Bits	Default Value	Description
7:4	0000b	**RSVD**
3	0b	**SLP_A# Value (SLP_A_VAL)** Value to drive on the SLP_A# pin.
2	0b	**SLP_S5# Value (SLP_S5_VAL)** Value to drive on the SLP_S5# pin.
1	0b	**SLP_S4# Value (SLP_S4_VAL)** Value to drive on the SLP_S4# pin.
0	0b	**SLP_S3# Value (SLP_S3_VAL)** Value to drive on the SLP_S3# pin.

37.2.31 SLPOUT2

7			4	3			0
0	0	0	0	0	0	0	0
RSVD						IE_SLP_A_VAL	ME_SLP_A_VAL

Bit	Default Value	Description
7:2	000000b	**RSVD**
1	0b	**IE SLP_A# Value (IE_SLP_A_VAL)** Contribution from IE to drive on the SLP_A# pin.
0	0b	**ME SLP_A# Value (ME_SLP_A_VAL)** Contribution from Intel ME to drive on the SLP_A# pin.

37.2.32 DSW_WAKE_STS

7			4	3			0
0	0	0	0	0	0	0	0
RSVD		DSW_ME_ALARM_STS	DSW_HOST_RTC_STS	DSW_PWRBTN_STA	DSW_AC_PRESENT_STS	DSW_USB_CON_STS	RSVD

Bit	Default Value	Description
7:6	00b	**RSVD**
5	0b	**DSW ME Alarm Status (DSW_ME_ALARM_STS)** DSW hardware sets this bit to indicate a Intel ME RTC alarm event has occurred while in DeepSx.
4	0b	**DSW Host RTC Status (DSW_HOST_RTC_STS)** DSW hardware sets this bit to indicate a host RTC event has occurred while in DeepSx.
3	0b	**DSW Power Button Status (DWS_PWRBTN_STS)** DSW hardware sets this bit to indicate a PWRBTN# event has occurred while in DeepSX.
2	0b	**DSW AC_PRESENT Status (DSW_AC_PRESENT_STS)** DSW hardware sets this bit to indicate that AC_PRESENT has toggled in either direction while in DeepSX
1	0b	**DSW USB Connection Status (DSW_USB_CON_STS)** DSW hardware sets this bit to indicate a USB connection event has occurred while in DeepSX if DSX_CFG.USB_CON_DSX_MODE = '1'

Bit	Default Value	Description
0	0b	**RSVD**

37.2.33 DSW_PM_FUSE_MON_0

7			4	3			0
0	0	0	1	1	0	0	0
RSVD			CORE_VR_ALLOWED_FUSE	RSVD		CORE_VID_FUSE	

Bit	Default Value	Description
7:5	00b	**RSVD**
4	1b	**CORE VR Allowed Fuse (CORE_VR_ALLOWED_FUSE)**
3:2	10	**RSVD**
1:0	0b	**CORE Voltage ID FUSE (CORE_VID_FUSE)** This field reports the core voltage ID as determined during manufacturing test and blown into fuses. 00 = 0.85V 01 = 0.90V 10 = 0.95V 11 = 1.00V These bits are only valid when CORE_VR_ALLOWED is 1b.

§

38 10 GbE Controller

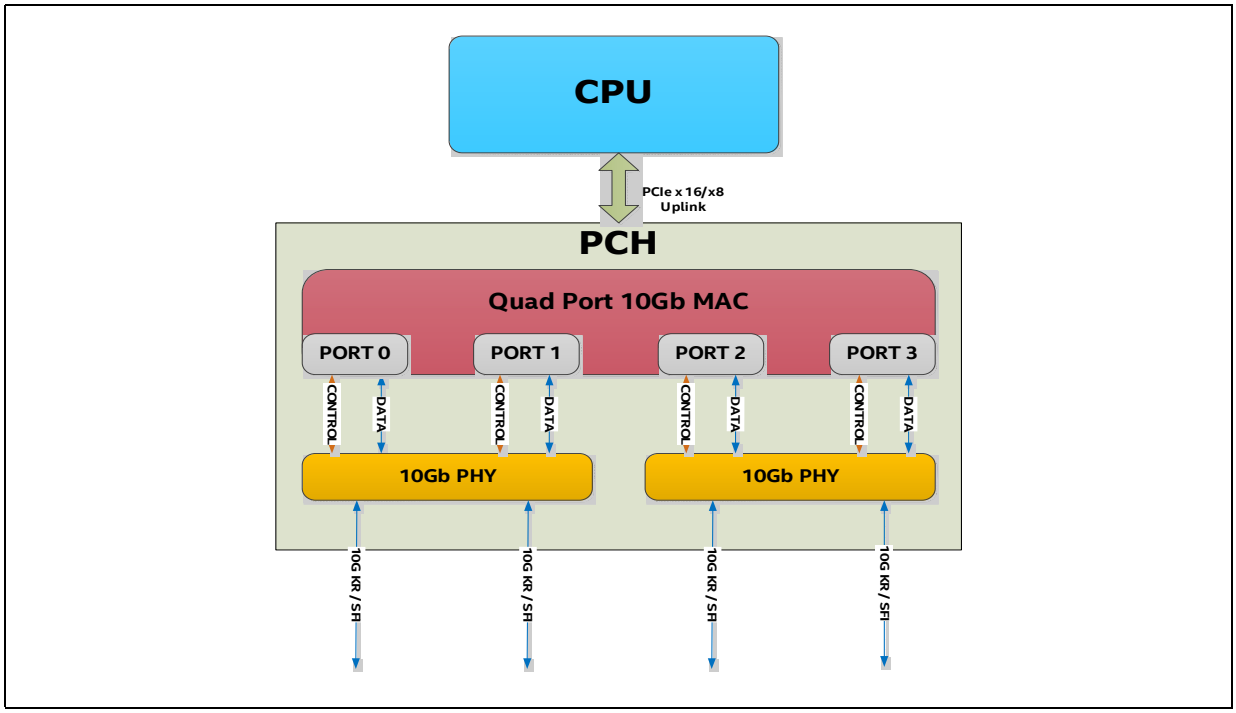
This section describes the external architecture (including device operation, pin descriptions, register definitions, etc.) for the quad-port, 10 GbE controller for the Platform Control Hub (PCH).

Information presented is intended as a reference for architects, logic designers, firmware and software device driver developers, board designers, test engineers, or anyone else who might need specific technical or programming information about the 10 GbE controller.

Note: Not all platforms support the 10 GbE controller even if SKUs that contain the controller. Please check the supported functions/features of the platform to see if the 10 GbE controller supported or not.

38.1 10 GbE Controller Overview

The 10GbE controller supports four XGMII-like interfaces to an integrated SerDes 10 GbE PHY enabling bandwidth of 10 GbE per port.



The 10 GbE controller supports 10 GbE / 1 GbE operations on its network data interface. Auto-negotiation, when configured for backplane Ethernet to automatically select between supported modes on each PHY as shown.

Each interface provides the following physical interfaces and electrical modes.

- Integrated modes of operation:
  - 10GBASE-KR for GbE backplane applications (IEEE* 802.3 clause 72)
  - 10GBASE-KR for backplane applications with Forward Error Correction (FEC) for BASE-R PHYs (IEE802.3 clause 74)
  - 10 GbE SFI SR/LR/DAC (SFF 8431)
  - 1000BASE-KX for backplane applications (IEEE 802.3 clause 74)
- Modes of operation requiring external PHY devices:
  - 10GBASE-T PHY (using an Intel® X557-AT2 (2-port)/ X557-AT4 (4-port) 10GBASE-T PHYs
  - 10 GbE SFI SR/LR/DAC (SFF 8431) re-timer option using Inphi* CS4227 series external PHY
  - 1000BASE-T using Marvell* 88E1514/88E1512/88E1543 external PHY

Note: These are the ONLY supported external PHYs.

Choose the appropriate configuration for your system configuration listed in Table 38-1.

Table 38-1 lists the supported operating modes and link partners.

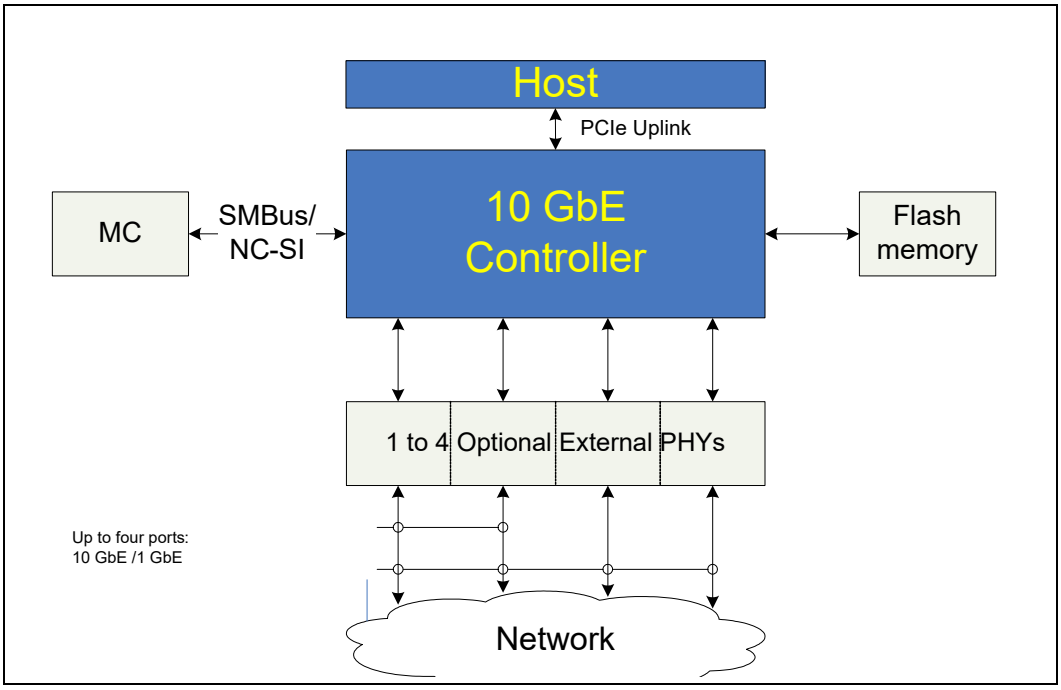
For detailed configurations, refer to the *Intel® X722 Lewisburg/Skylake-D 10 Gigabit Ethernet (GbE) Controller Reference Schematics*.

Table 38-1. 10 GbE LAN Configurations

Connection	Data Rate	Electrical Interface	3rd Party PHYs/Switches
Configurations Native to the PCH			
Backplane supports a direct connection to an aggregation switch.	10 Gb/s	10GBASE-KR	Intel FM5000/FM6000/FM10000 Ethernet switches and Broadcom* BCM5684x-series switches.
SFP+ / QSFP+.	10 Gb/s	SFI	Refer to the *Intel® Xeon® Processor Scalable Family Platform Design Guide* for a list of supported modules/cables.
Backplane supports a direct connection to an aggregation switch.	1 Gb/s	1000BASE-KX	Intel FM4000/FM5000/FM6000/ FM10000 Ethernet switches and Broadcom BCM5684x-series switches.
Configurations Requiring an External PHY			
10GBASE-T.	10 Gb/s	10GBASE-KR	Intel X557-AT2 (2-port)/ X557-AT4 (4-port).
SFP+ SR/LR/DAC for extended PCB reach.	10 Gb/s	10GBASE-KR	Inphi CS4227 (2-port) Inphi CS4223 (4-port)
1000BASE-T.	1 Gb/s	1000BASE-KX	Marvell 88E1514. Marvell 88E1512. Marvell 88E1543.

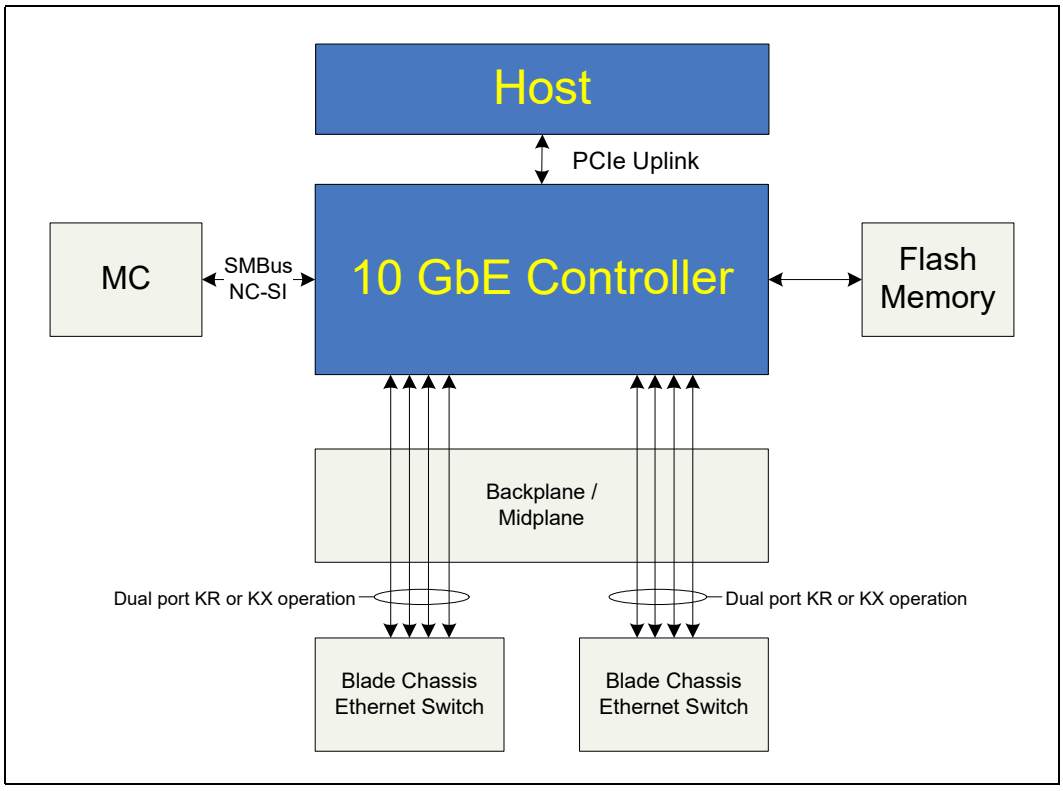
38.2 Typical System Configurations

Figure 38-1. Typical Rack / Pedestal System Configuration



As shown in Figure 38-1, the 10GbE controller is targeted for use in rack mounted or pedestal servers. Some types of Ethernet cables, such as SFP+ direct attach, can be driven directly by the 10 GbE controller, while other types, such as 10GBASE-T, require the external PHY component(s) shown.

Figure 38-2. Typical Blade System Dual-Redundant Star Configuration



As shown in Figure 38-2, the 10 GbE controller is also targeted for use in blade servers.

Blade backplanes typically connect Ethernet controllers in a dual-redundant star to two separate Ethernet switches as shown in Figure 38-2. In this configuration, the 10 GbE controller can be connected ports to each Ethernet switch or can be connected with only one port to an Ethernet switch, with two ports unused.

Figure 38-3 shows the 10 GbE controller's 10 GbE interfaces.

Figure 38-3. 10 GbE Controller Interfaces

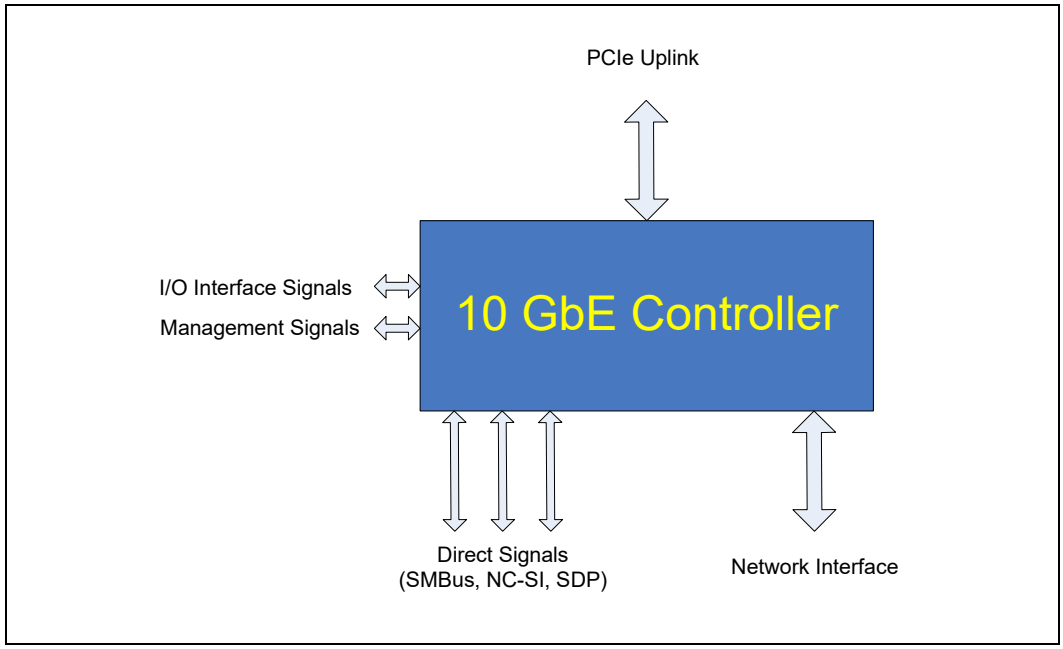


Figure 38-3 shows the 10 GbE controller as part of the Intel® C620 Series Chipset. The 10 GbE controller consists of:

- An integrated PHY interface:
- A digital I/O block that interfaces and manages various control I/O

38.3 High-Speed Signal Interface

The following table lists the 10 GbE controller network data interface pins.

Group Name	Signal Name	Type	Name and Function
(n = 0 through 3 for port#)			
Tx I/O and Rx I/O	LAN_TX_P<n>_P LAN_TX_P<n>_N	OUT	10 GbE / 1 GbE PHY Tx differential output
	LAN_RX_P<n>_P LAN_RX_P<n>_N	IN	10 GbE / 1 GbE PHY Rx Differential Input

38.4 Sideband Signals

The following sideband interfaces are available for the 10 GbE controller:

- MDIO
- I^2C
- LED Interface
- SDPs
- SMBus
- NC-SI

38.5 Low-speed Sideband Signal Interface

Group	Signal Name	I/O Pad Buffer Type	Internal Resistor PD	Internal Resistor PU	Description
(n = 0 through 3 for port#)					
Management Interface for Physical-Layer (PHY) Components. ***Note:*** MDIO and I^2C functions share the same interface pins, so only one can be operational at a time. These interfaces can be configured using NVM.	LAN_I2C_SCL_MDC_P[n]	I/O			I^2C clock of the 2-wire management interface used to access the management registers of an external optical module of port n. One clock pulse is generated for each data bit transferred. Management clock. Clock output for accessing the external PHY management registers. MDC clock frequency is proportional to link speed. At 10 Gb/s link speed, MDC frequency can be set up to 2.4 MHz.
	LAN_I2C_SDA_MDIO_P[n]	I/O			I^2C data of the 2-wire management interface used to access the management registers of an external optical module of port n. Stable during the high period of the clock (unless it is a start or stop condition). Management data. Bi-directional signal for serial data transfers between the MAC and the external PHY. Tri-state buffer requires an external pull up resistor.
LED	LAN_LED_P<n>_[1:0]	O			Two programmable LEDs per port that can be used for link status/link activity indications.
SDP	LAN_SDP_P<n>_[1:0]	I/O			Two general purpose SDP pins per port. Can be used to support IEEE 1588 auxiliary devices or input for external interrupts, etc.

Group	Signal Name	I/O Pad Buffer Type	Internal Resistor PD	Internal Resistor PU	Description
(n = 0 through 3 for port#)					
SMBus	LAN_SMBALRT_N	I/O			SMBus alert. Acts as an interrupt pin of a slave device on the SMBus.
	LAN_SMBCLK	I/O			SMBus clock. One clock pulse is generated for each data bit transferred.
	LAN_SMBDATA	I/O			SMBus data. Stable during the high period of the clock (unless it is a start or stop condition).
NC-SI	LAN_NCSI_ARB_ IN	I			NC-SI arbitration in
	LAN_NCSI_ARB_OUT	O			NC-SI arbitration out
	LAN_NCSI_CLK_IN	I			NC-SI reference clock input. Synchronous clock reference for receive, transmit, and control interfaces. It is a 50 MHz clock ±100 ppm.
	LAN_NCSI_CRS_DV	O			Carrier Sense/Receive Data Valid (CRS/ DV) to the Manageability Controller (MC). Indicates that the data transmitted from the 10 GbE controller to MC is valid.
	LAN_NCSI_RXD[1:0]	O			MC receive data. Data signals from the 10 GbE controller to the MC.
	LAN_NCSI_TX_EN	I			MC transmit enable. Indicates that received data from MC is valid.
	LAN_NCSI_ TXD[1:0]	I			MC transmit data. Data signals from the MC to the 10 GbE controller.

Group	Signal Name	I/O Pad Buffer Type	Internal Resistor PD	Internal Resistor PU	Description
(n = 0 through 3 for port#)					
Miscellaneous	PCI_DIS_N	I			This pin is a strapping pin latched while LAN_DIS_N, PE_RST_N or in-band PCIe reset are asserted. If this pin is not connected or driven high during initialization, then all PCI functions as configured from the NVM are enabled. If this pin is asserted/driven low during initialization, then all PCI functions that are allowed to be disabled as configured in the NVM are disabled. See Section 38.14 for more detail.
	LAN_DIS_N	I			This pin is a strapping option pin latched while LAN_DIS_N, PE_RST_N or in-band PCIe reset are asserted. This pin can be either used as a device disable or for disabling the LAN ports and associated functions based on the NVM configuration See Section 38.14 for more detail. If this pin is not connected or driven high during initialization, then all the LAN ports and associated functions as configured from the NVM are enabled for normal operation. If this pin is asserted/driven low during initialization then the LAN ports and associated functions as configured from the NVM are disabled. Asserting this pin disables the entire device if all the LAN ports are configured to be disabled. When the entire device is disabled, the PCIe link is in L3 state, the PHY is in power down mode, and the output buffers are tri-stated. See Section 38.14 for more detail.

38.6 Host Interface

The connection to the 10 GbE controller is enabled via a dedicated PCIe uplink (X8/X16). See Chapter 22 for more details about the PCIe host interface.

38.6.1 Host Memory Objects

The 10 GbE controller operating system drivers set up a wide variety of host memory objects that are comprehended and manipulated by the 10 GbE controller. All objects are set up in the context of a PCI function. This is important for at least two reasons:

- **Platform security:** For example, many types of the 10 GbE controller host memory objects are privileged, and are only allowed to be set up in the context of a PF. For example, they are dis-allowed in the context of a Virtual Function (VF), which operates at a lower privilege level than a PF.
- **Reliability:** If the operating system resets a PCI function, that function's host memory objects are lost, but the host memory objects of other PCI functions survive.
- **LAN Queue Pairs:** These are ring buffers (one transmit, one receive) for submitting commands to the LAN engine. Commands take the form of packets/data to be transmitted, descriptors for empty host memory buffers to be filled with received packets/data, etc. LQPs are typically mapped into OS kernel space. In a virtualized server, they can be assigned either to the VMM, or to VMs using SR-IOV. The 10 GbE controller supports up to 1536 LQPs that can be assigned to PFs or VFs as needed. The LQPs assigned to a particular PCI function can be used in these important ways:
  - For distributing packet processing work to the different processors in a multi-processor system. On the transmit side, this is done by simply dedicating an independent transmit queue for each CPU to use. On the receive side, packets are classified by the 10 GbE controller under operating system control into groups of conversations. Each group of conversations is assigned its own receive queue and receiving processor. Microsoft* RSS is one popular example of this method.
  - For assigning Traffic Class. Transmit queues assigned to different TCs are serviced at different rates by the 10 GbE controller transmit scheduler. Receive queues assigned to different TCs can be serviced at different rates by a Quality of Service (QOS0-enabled operating system and its software device drivers.
- **Protocol Engine Queue Pairs:** These are ring buffers (one Send Queue, one Receive Queue) for submitting commands to the Protocol Engine. Each Protocol Engine QP can be configured for either RDMA messages or for UDA packets. Commands on RDMA QPs take the form of RDMA messages to be transmitted, descriptors for empty host memory buffers to be filled with received RDMA messages, etc. Commands on UDA QPs take the form of packets/data to be transmitted, descriptors for empty host memory buffers to be filled with received packets/data, etc. Protocol Engine QPs are typically mapped into host *user space*. In a virtualized server, they can be assigned either to the VMM, or to Virtual Machine user space using SR-IOV. The 10 GbE controller supports up to 262,144 Protocol Engine QPs which are allocated to PFs and VFs as the 10 GbE controller initializes and drivers load, and are fixed after that.

- **Admin Queue Pairs:** Each PCI function maps an Admin Queue Pair: one Admin Send Queue and one Admin Receive Queue. The ASQ is a ring buffer used by the host driver for submitting commands to configure the 10 GbE controller. Commands submitted on the ASQ are serviced by the 10 GbE controller's Embedded Management Processor. Some examples are commands to reconfigure: the Tx scheduler, an Ethernet link, power management states like various internal switch and DCB settings, etc. The ARQ conveys events from the EMP to host driver that are not an immediate result of an ASQ command. The host driver posts empty buffers to the ARQ and the EMP fills them with events. Note that commands submitted on a VF ASQ are typically not directly serviced by the EMP, but rather are redirected to the associated PF ARQ. This enables the PF driver to inspect and authorize all VF ASQ commands which are often of a privileged nature.

38.6.2 Ethernet Media Access Controller

The 10 GbE controller integrates four IEEE Std 802.3 compliant Ethernet MACs that operate at 10 GbE. All 10 GbE controller MACs support transmission and reception of jumbo frames of up to 9728 bytes, and 802.3x flow control frames or 802.3bd priority-based flow control frames. The interface of each MAC is XGMII.

The 10 GbE controller also implements either four independent Management Data Input/Output interfaces or four independent I^2C interfaces for connection to external PHYs. These enable host software or the 10 GbE controller firmware to control connected external PHYs, including the ability to read and write PHY registers.

38.6.3 Transmit Scheduler

The 10 GbE controller provides management interfaces that allow each LAN transmit queue or protocol engine transmit queue to be placed into a *queue set*. A *queue set* is a list of transmit queues that belong to the same TC and are treated equally by the 10 GbE controller transmit scheduler. The 10 GbE controller supports a maximum of 384 Virtual Station Interfaces, and an average of four queue sets per VSI for use by LAN and two for use by the Protocol Engine. There is no limit to the number of transmit queues that can belong to a queue set.

Typically, one or more queue sets are assigned to a PCI function/VSI, according to the number of TCs it uses. Since each Protocol Engine transmit queue typically represents a single TCP/IP connection, it is easy to see how a Protocol Engine queue set will often have a large number of members. For LAN, the queue sets are often only one or a small handful of transmit queues, corresponding to the number of host CPUs assigned to generate traffic on that TC.

The 10 GbE controller transmit scheduler supports independent programming of a wide variety of controls that affect queue set behavior. Each queue set can be programmed with an independent static rate limit. Each group of queue sets assigned to a PCI function/VSI can be programmed with DCB Enhanced Transmission Selection settings, group static rate limit, and uplink bandwidth share.

The 10 GbE controller transmit scheduler also implements controls similar to those previously described for the various internal resources that comprise the 10 GbE controller's hierarchy of internal switching components. These internal resources include Virtual Ethernet Bridge, Virtual Ethernet Port Aggregator and SComp v-ports, as well as the physical Ethernet ports of the device.

38.6.4 Host Memory Cache

The 10 GbE controller LAN engine and Protocol Engine use host memory as a backing store for a variety of context objects. The HMC is responsible for caching and managing these context objects. Here are some examples:

- The LAN engine uses two HMC context objects per QP, one for the Transmit Queue and one for the Receive Queue. Parameters in the TQ context include a Transmit Segmentation Offload state. Parameters in the RQ context include a queue base address and associated pointers.
- For each RDMA connection, the Protocol Engine uses a *QP Context* object. For example, a TCP/IP connection context that stores TCP sequence numbers and IRRQ objects that buffer inbound RDMA Read Requests until their associated Read Responses are scheduled for transmit. For each RDMA Memory Region, the Protocol Engine uses a *Memory Region Table Entry* object to store region boundary and access rights information, and a set of *Physical Buffer List Entry* objects to store virtual-to-physical address translations for this region.

38.6.5 LAN Engine

The LAN engine implements the host programming interface for traditional LAN traffic in both virtualized and non-virtualized scenarios. The 10 GbE controller implements 384 VSIs (virtual-NICs) used to distribute traffic to PCI physical functions and virtual functions. These VSIs can connect to the host via 1536 LAN QPs.

The LAN engine also implements all of the 10 GbE controller's performance optimizations for traditional LAN traffic. This includes packet checksum offloads, Transmission Control Protocol/User Datagram Protocol (TCP/UDP Segmentation Offload), RSS and others.

38.6.6 Protocol Engine

The Protocol engine implements iWARP RDMA capability. The Protocol engine offloads TCP/IP processing from the host, places received data payloads directly into user buffers with no host CPU involvement, and eliminates user-kernel context switching when performing I/O by mapping the RDMA programming interface directly into application address space.

The Protocol engine implements the latest RDMA features, including Send Queue Push Mode. It also implements Userspace Direct Access, a means to accelerate select IP packet streams by conveying them directly to/from user space applications using RDMA-style host interfaces (QPs, CQ, etc).

38.6.7 System Management

The 10 GbE controller participates in system management by providing networking services to platform management controllers; also called BMC. The 10 GbE controller is also accessible to these devices to be managed as any platform resource that is managed by the system.

Networking services are provided through the Pass-Through functionality. Several sideband channels are provided to connect to a Management Controller:

- SMBus
- NC-SI
- Management Component Transport Protocol over PCIe.

The 10 GbE controller also supports communication between local software agents and the local BMC through an operating system to BMC capability.

38.6.8 Embedded Management Processor (EMP)

The EMP unit handles all management duties that cannot be performed by the 10 GbE controller's device drivers, and must be carried out on-chip. This includes performing parts of The 10 GbE controller power-on sequence, handling AQ commands, initializing the 10 GbE controller Ethernet ports, participating in various fabric configuration protocols such as DCBX and other Link Layer Discovery Protocols, fielding configuration requests received on one of the 10 GbE controller's BMC management interfaces such as NC-SI, and handling special configuration requests received off an Ethernet port.

38.6.9 Internal Switch

The internal switch handles the 10 GbE controller packet buffering and also implements all its internal Ethernet switching capabilities. The internal switch can logically support up to 16 VLAN-aware bridges, connecting up to 384 VSIs out to the 10 GbE controller Ethernet ports via optional EVB S-Components. The internal switch implements the filtering and forwarding behaviors expected by 802.1Q VLAN-aware bridges.

38.6.10 Receive Scheduler

The receive scheduler prioritizes received packets according to the ETS settings programmed at each Ethernet port.

38.6.11 Receive Filters

The 10 GbE controller receive filters implement all of the 10 GbE controller's logic for queue selection. For each received packet, the forwarding process carried out by the internal switch selects which VSI the packet is associated with. The receive filters then determine which engine(s) handle the packet (EMP, LAN engine, Protocol Engine) and which selected VSI RQs the packet is associated with. Receive filters include flow director, Quad Hash filtering, RSS filters, etc.

38.7 Various Interfaces

38.7.1 Shared Serial Flash Interface

The 10 GbE controller accesses the NVM via the SPI controller of the 10 GbE controller.

The NVM device is required for storage of device firmware, device configuration parameters, identifiers that vary per adapter (like MAC addresses), and register overrides that auto load automatically after reset. The 10 GbE controller assumes that at least 6 MB of the NVM are dedicated for these configurations.

38.7.2 SMBus Interface

SMBus is an optional interface for pass-through and/or configuration traffic between an external BMC and the 10 GbE controller. The 10 GbE controller's SMBus interface supports standard SMBus at 100 kHz and extensions up to a frequency of 1 MHz.

38.7.3 NC-SI Interface

NC-SI is an optional interface for pass-through and/or configuration traffic between an external BMC and the 10 GbE controller.

38.7.4 SDPs

SDPs are typically used to exchange information with or to control external devices (such as PHY devices) under the 10 GbE controller software driver control.

38.7.5 LEDs

The 10 GbE controller implements eight output drivers intended for driving external LED circuits. The 10 GbE controller can be configured so that either two of these outputs are allocated per port or four outputs are allocated per port (this later configuration is used when two ports are disabled). Each of the LED outputs can be individually configured to select which particular event, state, or activity it indicates. In addition, each LED can be individually configured for output polarity and for blinking versus non-blinking (steady-state) indications.

38.7.6 LAN_DIS

See Section 38.5 for more detail.

38.7.7 PCI-DIS

See Section 38.5 for more detail.

38.8 Features

This section lists the 10 GbE controller's feature set.

Table 38-2. PCIe* Host Interface Features (Sheet 1 of 2)

Description
Host primary interface 32 Bytes @ 400 MHz. Can be connected to a virtual switch port fabric, an RCIE fabric or a PCIe link layer fabric.
Requester Features • 64-bit address support for systems with more than 4 GB of physical memory • Maximum of 208 outstanding requests, allocated fairly as needed amongst PCI Functions • Maximum payload size supported: 512 bytes • Maximum read request size supported: 2 KB • All requests use TC TC0/VC0, the best effort service class for general purpose I/O Optimized support for relaxed Ordering transaction attribute • Optimized support for TLP Processing Hints • Optimized support for ID-based Ordering
Completer Features • Up to 4 Peripheral Component Interconnect PFs Device can be configured to disable/hide any number of these PFs. • Each PF implements these Base Address Registers (BARs) — Memory BAR for direct access to most Internal Registers. This BAR can be configured to define either a 32-bit or 64-bit base address. It's size is 256 KB or larger. — MSI-X BAR for direct access to MSI-X-related structures. This BAR can be disabled/hidden. It can be configured to define either a 32-bit or 64-bit base address and it's size is 32 KB. — Expansion ROM BAR for direct host access to one of the 10 GbE controller's option ROM images. This BAR can be disabled/hidden. It defines a 32-bit base address and it's size is 64 KB or larger. — I/O BAR for indirect access to most Internal Registers. This BAR can be disabled/hidden. It defines a 32-bit base address and it's size is 32 bytes. • PF capabilities list (some of these can be disabled/hidden) — PCI Power Management — MSI — MSI-X — PCIe — Vital Product Data: Each PF can access a single shared instance of VPD storage, with a size up to 256 bytes — Advanced Error Reporting — Device Serial Number — Alternative RID Interpretation — SR-IOV — TPH Requester — Access Control Services — Secondary PCIe
Reliability • AER • Support for optional End-to-End CRC generation and checking • Recovery from data poisoning • Completion timeout
Power Management • Supports the PCI Power Management specification and Section 5 of the *PCI Express Base Specification*. • Does not support D1 or D2 power management states. • Does support D0uninitialized, D0active, $D3_{HOT}$, and $D3_{COLD}$ power management states. • $D3_{HOT}$ transition to D0 does preserve configuration context (as indicated by the RO PCI configuration bit No_Soft_Reset=1b)
Interrupt Functionality • INTx: Supports four INTx interrupts conveyed using the PCIe INTx virtual wire signaling mechanism. A given PF can only map a single INTx. • MSI: Supports one Message-Signaled Interrupt per PF. • MSI-X: Supports up to 1168 MSI-X vectors, shared among PFs and VFs. Up to 129 MSI-X interrupts can be assigned to a single PF and up to 17 MSI-X interrupts can be assigned to a single VF, depending on the number of enabled functions. • Sophisticated interrupt moderation using Interrupt Throttling (ITR) and Interrupt Rate Limiting (INTRL).

Table 38-2. PCIe* Host Interface Features (Sheet 2 of 2)

Description
I/O Virtualization • Supports PCI-SIG SR-IOV Specification with up to 128 VFs. VFs can be flexibly assigned to PFs, although reassignment requires re-enumeration of the PCI bus hierarchy. • Each VF implements these BARs — Memory BAR for restricted access to Internal Registers. This BAR can be configured to define either a 32-bit or 64-bit base address. It's size is 16KB or larger. — MSI-X BAR for direct access to MSI-X-related structures. This BAR can be configured to define either a 32-bit or 64-bit base address. It's size is 16KB. • Per-VF Capabilities list (some of these can be disabled/hidden) — MSI-X — PCIe — AER — ARI — TPH requester — ACS

Table 38-3. Link Layer Ethernet Port Features

Description
Ethernet Speed and Interfaces (see Table 38-2). 10 Gb/s: Four ports of XGMII (supporting 10 GbE and 1 GbE) 1 Gb/s: 4 ports over XGMII (supporting 10 GbE and 1 GbE
The 10 GbE controller's Maximum Transmit Unit Size is 9728 - Ethernet header/CRC = 9728 - 18 = 9710 bytes (jumbo frames) MTU can be further reduced by additional header fields such as Virtual Local Area Network tag(s), etc.
Full-duplex operation at all supported speeds
Integrates support for IEEE Std 802.3 Clause 73 Auto-Negotiation for Backplane Ethernet

Table 38-4. Performance On The Network vs. Packet Size

Description
Maximize link capacity when operating at 4x10 Gb/s links with packets larger than 128 bytes at full duplex traffic
Maximize link capacity when operating at 2x10 Gb/s links or 1x10Gb/s link in all packet sizes at full duplex traffic

Table 38-5. Transmit And Receive Scheduling

Description
Queue Sets (see Section 38.6.3**)** • Supports up to 1024 RDMA/UDA Queue Sets (768 typically available) • Supports up to 1024 LAN queue sets (768 typically available) • Each queue set is assigned to the TC of a particular VSI • Round Robin bandwidth distribution between TQs assigned to same queue set • Round Robin bandwidth distribution between RDMA/UDA and LAN queue sets belonging to same TC of VSI
Quanta: • Transmit work is scheduled in units of configurable per chip Quanta bytes. • Quanta can be configured to be 1KB, 2KB, 4KB, 8KB, 16KB, 32KB or 64KB.
VSI: • Supports up to 384 VSIs with average of two TCs allocated per VSI • Independent ETS/SLA configuration per VSI • Configurable bandwidth limit per VSI.
VEB/VEPA: • Supports up to 16 VEB/VEPA switching components • Configurable bandwidth allocation among VEB/VEPA VSIs • Configurable bandwidth Limit of VEB/VEPA egress port
S-components: • Supports up to four S-components, one per physical port • Configurable bandwidth allocation among S-channels • Independent ETS configuration per S-component egress port • Configurable bandwidth limit of S-component egress port
Bandwidth Distribution: • Supports two bandwidth allocation modes — Relative bandwidth allocation - used for ETS, and bandwidth allocation between ingress ports of the switching component — Best effort bandwidth allocation - used for SLA • Supports three arbitration schemes — Weighted Round Robin — Weighted Strict Priority — Combination of WSP and weighted Round Robin • On any given virtual link, minimum bandwidth allocation is 1% of the virtual link bandwidth
Bandwidth Limit: • Configurable bandwidth limit in range of 10 Gb/s - 1 Mb/s with increment of 1 Mb/s • Configurable maximum bandwidth accumulation with a maximum of 200%

Table 38-6. LAN Engine Features

Description
VSI Support For each of the 10 GbE controller's 384 allocated VSIs, the LAN engine allocates the following independent resources: • The LAN engine supports a total of 1536 LQPs — A PF VSI can allocate and use up to 1536 LQPs — A VF VSI can allocate and use up to 16 LQPs • Statistics: One set of IEEE Std 802.3 Clause 30 hardware statistics counters per VSI • TSO context: For each TQ, the LAN engine allocates storage for TSO context like TCP sequence numbers and other header fields changed by the TSO process. This enables each TQ to make independent progress on its TSO operations. • RSS: RSS logic (key and lookup table) is implemented per VSI enabling the PF as any guest operating system assigned to a VF or VMDq2 VSI use their own RSS logic. The following resources are notably not allocated per VSI: • Interrupts: Interrupt vectors are typically allocated per CPU core. Software controls how a single vector is shared amongst interrupt causes (like LAN queues). Typically sharing is limited to a small number of LAN queues to minimize polling performance loss.
Programming Interface. • LAN TQ descriptors: There are different LAN TQ descriptors that define: packet data, transmit context (such as TSO information), flow director filter programming. All TQ descriptors are 16 bytes. The packet data descriptor defines one fragment. If a packet or TSO is comprised of multiple fragments, packet data descriptors can be chained, up to a limit of eight per transmitted Ethernet packet (TSOs are allowed more). TQ completions are signaled either by descriptor writeback, or by updating a software head pointer in host memory. Software controls this signaling mechanism on a per-TQ basis. Software controls signaling frequency on a per-descriptor basis. • LAN RQ descriptors: Each LAN RQ descriptor defines up to two fragments, one for the packet, and one for the header (used when optional header splitting is enabled). For a given RQ, the fragment lengths are fixed and programmed into RQ context. Maximum size of the packet fragment is 16 KB and maximum size of the header fragment is 2 KB. A received packet is allowed to span multiple RQ descriptors, up to a limit of five. When the LAN engine has filled a LAN RQ descriptor, it writes completion status back to the descriptor. RQ descriptors can be configured per queue to be either 16 or 32 bytes in size, depending on the amount of completion status detail desired. • LAN Q properties (TQ and RQ): LAN Qs are mapped into host memory as physically contiguous ring buffers. Maximum LAN Q depth is configurable on a per-Q basis. Supported values range from 8 to 8 KB entries. Maximum LAN TQ size is 8 KB entries x 16 bytes = 128 KB. Maximum LAN RQ size is 8 KB entries x 32 bytes = 256 KB. Each LAN Q can be individually disabled.
Performance Optimizations (available to every VSI) • Packet checksum offloads: Calculates IP, UDP, TCP, and Stream Control Transmission Protocol (SCTP) checksums for insertion into transmitted packets and for integrity checking on received packets. Includes support for UDP and TCP checksums in both IPv4 and IPv6 datagrams. • TCP/UDP segmentation offload, also referred to Large Send Offload for transmitted IPv4 and IPv6 packets. The maximum size of a TSO operation is 256 KB. • Received packet header splitting. This feature enables a received packet's header to be placed in a different host buffer than the packet payload. Each RQ can be configured independently to perform header splitting at a specified header layer: none, IP, TCP, UDP, etc. • RSS: PF instances of the RSS logic implement 256 entry indirection tables, supporting up to 64 RQs/CPUs.VSI instances of the RSS logic implement 64 entry indirection tables, supporting up to 16 RQs/CPUs.
Intel Virtual Machine Device Queues Functionality VMDq defines the hardware offloads that support Virtual Bridges implemented in software (usually in a VMM). The 10 GbE controller supports both VMDq version 1 and version 2. • Some of the major features of VMDq version 1 — Layer 2 filters for sorting packets based on MAC address — Layer 2 filters for sorting packets based on VLAN tags — 1536 LAN QPs that can be flexibly allocated to the PFs for VMDq flows — Default queue mechanism for non-matching packets — MSI-X interrupt per queue • Some of the major features of VMDq version 2 — Internal switching from a TQ to a RQ — Broadcast and multicast replication — Ability to sort packets based on a combination VLAN tag and MAC address filter — Anti-spoofing transmit filters (VLAN and MAC) — Dedicated RSS logic assigned to the VMDq version 2 VSI
Preboot Execution Environment Supports the PXE remote boot standard. Also supports Internet Small Computer System Interface boot via expansion ROM firmware.
Flow director filters: 8 K perfect-match filters stored on-chip

Table 38-7. Protocol Engine iWARP RDMA Features (Sheet 1 of 3)

Category	Description
TCP offload	**IP version** Support IPv4 and IPv6 TCP-offloaded RDMA connections. Certain complex packet formats are not supported with RDMA (such as TCP-IPv6 tunneled with UDP-IPv4 using Teredo protocol).
TCP offload	**Maximum Segment Size** TCP-offloaded RDMA connections MSS: 9728 - Ethernet header/CRC - IP header - TCP header IPv4 MSS = 9728 - 18 - 20 - 20 = 9670 bytes IPv6 MSS = 9728 - 18 - 40 - 20 = 9650 bytes MSS can be further reduced by additional header fields such as TCP option(s), etc.
TCP offload	**ARP Table** Each PCI Function enabled to use the Protocol Engine implements an ARP Table with up to 65536 entries. Each entry contains a six byte Ethernet address, plus control/status information for neighbor reachability detection. For all RDMA connections, the ARP Table supplies a destination Ethernet address for transmitted packets.
TCP offload	**Receive Window Size** Each RDMA connection has a configurable Receive Window with maximum size of 1GB-1B.
TCP offload	**TCP Timestamp** The TCP Timestamp option is supported.
TCP offload	**IP Datagrams/Fragmentation** IP datagrams with IP fragmentation that are received on RDMA connections are forwarded to the Host CPU for reassembly. In most cases, the Host CPU will return reassembled packets to the 10 GbE controller for further TCP and RDMA processing.
general	**Non-permissive IETF RNIC** Using a term coined in the *Marker PDU Aligned Framing for TCP* specification, the device is a Non-permissive IETF RNIC (an RNIC that implements the IETF protocols but not the RDMAC protocols).
general	**PCI Functions enabled to use the Protocol Engine** All PCI Physical Functions, and up to 32 Virtual Functions can be enabled to use the Protocol Engine.
general	**Marker PDU Aligned Protocol Support** Supports insertion of Transmit markers MPA CRC checking on inbound packets can be enabled or disabled per QP. MPA CRC is always generated for outbound iWARP packets.
QP	**Maximum QP Count** The Protocol Engine supports up to 262144 QPs which are allocated to PFs and VFs as the 10 GbE controller initializes and drivers load, and are fixed after that.
QP	**Work Queue Elements** Protocol Engine uses a similar WQE format for SQs, RQs and RQs Each WQE can vary in size, with these options: 32B, 48B, 64B, 80B, 96B, 112B, 128B Fragments per WQE: one to seven, corresponding to above sizes Fragments are specified with virtual addresses and are virtually contiguous. The Protocol Engine performs virtual-to-physical translation using its built-in Memory Management Unit (MMU) SQ RDMA Reads are limited to a single fragment by the iWARP specification A SQ WQE can optionally directly convey 112B of inline data for small message latency optimization.
QP	**Work Queues** Protocol Engine WQs are mapped into host memory as ring buffers. The Protocol Engine allows a WQ/ring buffer to be either physically or virtually contiguous. A physically contiguous WQ/ring buffer is preferred for performance applications because it eliminates a Virtual-to-Physical address translation. Maximum WQ size is 16K entries x 32B = 512 KB. Maximum WQ depth is configurable on a per-WQ basis. Supported values range from 4 to 16K 32B WQEs, in power-of-two increments. Dynamic WQ resizing is not supported.

Table 38-7. Protocol Engine iWARP RDMA Features (Sheet 2 of 3)

Category	Description
QP	**SQ Operations** All RDMA operation types defined by the PostSQ verb are supported and implemented by on-chip logic: Send, Send with Solicited Event, Send with Invalidate, Send with Solicited Event and Invalidate RDMA Write RDMA Read RDMA Read with Invalidate Local STag Bind Memory Window Fast-Register Non-Shared Memory Region Invalidate Local STag Fencing Blocking transmission when ORD limit is reached
QP	**Number of RDMA Reads** The number of outstanding inbound RDMA Reads can be configured independently per QP. This configuration parameter is called *Inbound RDMA Read Queue Depth*. The Protocol Engine allows IRD settings of 2, 8, 32, or 64. The HMC manages the temporary storage for inbound RDMA Reads, which is allocated in Host Memory. The Protocol Engine does not support modifying a QP's IRD after the QP has been created. The number of outstanding outbound RDMA Reads can also be configured independently per QP. This configuration parameter is called *Outbound RDMA Read Queue Depth*. The Protocol Engine allows ORD settings from 0 to 127, in single-step increments. The Protocol Engine *does* support modifying a QP's ORD after the QP has been created, if the proper quiesce conditions are met.
QP	**Maximum iWARP message size supported** Tx RDMA Writes, Tx Sends, Rx RDMA Reads: 2GB - 1 byte (limited by TCP sequence space) Rx Sends, Rx RDMA Writes, Tx RDMA Reads: 4 GB - 1 byte (limited by iWARP specification)
QP	**Send Queue Push Mode** To fully optimize small message latency, the Protocol Engine supports *Send Queue Push Mode*. In this mode, the Host CPU writes or pushes SQ WQEs with up to 112 bytes of inline data to Protocol Engine memory-mapped address space using CPU write-combining buffers. This mode achieves lower latency than can be achieved with standard inline data, because it eliminates a Protocol Engine SQ WQE read from host memory.
CQ	**Completion Queues** The Protocol Engine supports up to 131072 CQs which are allocated to PFs and VFs as the 10 GbE controller initializes and drivers load, and are fixed after that. A CQ/ring buffer can be either virtually or physically contiguous. A physically contiguous CQ/ring buffer is preferred for performance applications because it eliminates a Virtual-to-Physical address translation. Each CQE is 32B, and can optionally be padded to 64B cache-line boundary to avoid memory conflicts. Max CQ size is 1M entries x 32B = 32 MB (or 64 MB if optional 64B CQE size is used) Supports user-defined mapping of WQs to CQs. Supports CQ resizing, CQ size can be increased or decreased while the CQ is active. Supports CQ overflow detection.
EQ	**Event Queues** The Protocol Engine implements 256 Completion Event Queues, allocated amongst the Protocol Engine enabled PCI functions. Each CEQ reports all the Completion Events for a configurable group of CQs, making it easy for software to determine which CQ has new CQEs. The Protocol Engine implements one Asynchronous Event Queue per Protocol Engine enabled PCI function (up to a maximum of 48 AEQs). Each AEQ reports status and errors associated with the Protocol Engine Queue Pairs, Completion Queues, and ARP Table Entries of it's PCI Function. Typically, each active CEQ and AEQ is configured with a dedicated MSI-X vector. Other configurations are supported (e.g., sharing an MSI-X vector amongst a CEQ and an AEQ).
MMU	Protection Domains Each PF or VF enabled to use the Protocol Engine can define up to 32768 Protection Domains
MMU	**Memory Regions and Windows** Each PF or VF enabled to use the Protocol Engine can allocate up to 4M (i.e., 2^{22}) Memory Regions or Memory Windows A single Memory Region or Memory Window can be up to 32TB in size.
MMU	**Host Memory page sizes** Host Memory page sizes supported: 4 KB, 2 MB Each Memory Region can be independently configured for either page size

Table 38-7. Protocol Engine iWARP RDMA Features (Sheet 3 of 3)

Category	Description
MMU	**Physical Buffer List (PBL)** The Protocol Engine allows each Memory Region to be Physically Mapped (i.e., the region is physically contiguous in Host Memory) or Virtually Mapped with a one- or two-level PBL. Each PF or VF enabled to use the Protocol Engine can allocate up to 256M (i.e., 2^{28}) PBL Entries. A single PBL Entry maps a single Host Memory page. The 10 GbE controller does not support Block Mode as defined in the *RDMA Protocol Verbs Specification*
MMU	**Maximum Virtually Mapped Memory** The maximum virtually mapped memory a single PF or VF can register with the Protocol Engine is implementation-limited to 2^{45} bytes = 32 TB. The maximum may be further reduced depending on host page sizes used. For example, if a PF or VF allocates it's maximum limit of 256M PBL Entries, then... ...its maximum virtually mapped memory using 100% 2 MB host memory pages is: MIN(2^{45}, 2^{21} x 2^{28}) = 2^{45} bytes = 32 TB ...its maximum virtually mapped memory using 100% 4 KB host memory pages is: MIN(2^{45}, 2^{12} x 2^{28}) = 2^{40} bytes = 1 TB

Table 38-8. Protocol Engine User Space Direct Access Features

Description
User Space Direct Access Definition UDA enables select IP packet streams to be delivered directly to/from user space applications (bypassing the OS) using RDMA-style host interfaces (QPs, CQ, etc). Host interface resources such as QPs are shared between RDMA and UDA consumers.
UDP Accelerations When multiple UDA consumers wish to receive the same UDP multicast stream, each can register its QP with the Protocol Engine to be members of a multicast group. The Protocol Engine will replicate received UDP multicast datagrams to the QPs in a matching multicast group. Each PF enabled to use the Protocol Engine can define up to 65536 multicast groups.

Table 38-9. Internal Switching Features (Sheet 1 of 2)

Description
VSI Support The 10 GbE controller supports a total of 384 VSIs. VSI assignment is flexible, but the choice to support 384 VSIs is motivated by the following usage example: • 256 VSIs for VFs or VMDq2 • 32 VSIs for PFs (16 for LAN + 16 for RDMA)   — 32 VSIs for VFs with RDMA • 24 control ports (16 for VEBs/VEPAs, 4 for Ethernet ports, 4 for S-comp/E-comp) • 16 mirror ports • Four for EMP • 20 extras

EVB and Bridge Port Extension Functionality
• Supports EVB as defined in the IEEE P802.1Qbg specification
• The 10 GbE controller supports up to four S-components or four eBridge port extenders (one per port), up to 512 S-tags, and up to 384 S-channels.
• The 10 GbE controller supports up to 16 internal VEBs or VEPAs.
• A single Ethernet port can connect up to 384 S-channels. Connecting to these S-channels can be VEBs, VEPAs, or port extenders. The 10 GbE controller must be configured for either VEPA or port extenders (both are not supported simultaneously). Given these rules, plus the global constraints described in the preceding bullets, a single Ethernet port can connect either:
  — Up to 16 VEBs or VEPAs. This case only requires up to 16 S-channels.
  — Up to 384 EBPs or VEBs. Of the 384, only up to 16 can be VEBs.
• Each VEB is compliant with the IEEE 802.1Q Ethernet VLAN-aware bridge specification.
• Supports MAC address forwarding table with 1536 entries.
• Supports MAC and VLAN address forwarding table with 2048 pairs.
• Supports VLAN insertion and removal for up to 256 VLAN tags located anywhere in the 4K VLAN space.
• Supports Private VLANs.
• Supports port mirroring.
• Each VSI can be programmed for promiscuous reception of unicast, broadcast and multicast packets.

Table 38-9. Internal Switching Features (Sheet 2 of 2)

Description
Internal switching operates independently of the state of the LAN ports (also when LAN ports are down).

Table 38-10. System Manageability Features

Description
Sideband Interfaces for connection to an external BMC • SMBus operating at up to 1 Mb/s • DMTF-compliant NC-SI Interface at 100 Mb/s • PCIe (together with MCTP)
Sophisticated filters to select received packet flows for delivery or mirroring to the BMC. Each of the following filters is instantiated per-Ethernet port: • Four MAC filters • Eight VLAN filters • Four ether-type filters • Four to seven IPv4/IPv6 filters • 16 UDP/TCP port filters • One Flexible Total Cost of Ownership filter • Address Resolution Protocol filtering • Neighbor discovery filtering • Remote Management Control Protocol filtering • Internet Control Message Protocol filtering
Ability to internally switch packets for communication between an OS and BMC.
Statistics For each Ethernet port, the 10 GbE controller implements the statistics defined in the following standards: • IEEE Std 802.3 Clause 30 • RFC 2819 - RMON Ethernet statistics group For each VSI, the 10 GbE controller implements the statistics defined in the following standard: IEEE Std 802.3 Clause 30. For each VSI enabled to use the Protocol Engine, the 10 GbE controller silicon implements these additional statistics: • RFC4293: Objects for managing implementations of IP • RFC4022: Objects for managing implementations of TCP RFC4113: Objects for managing implementations of UDP

Table 38-11. Power Management Features

Description
Supports the PCIe power management features defined in the “Power Management” entry in Table 38-2.
Energy Efficient Ethernet Provides the MAC support for Energy Efficient Ethernet as defined in the IEEE P802.3az specification.
Low Power Link Up In a backplane Ethernet environment, when the 10 GbE controller is entering a low power state like $D3_{COLD}$, the 10 GbE controller’s Low Power Link Up logic attempts to re-auto-negotiate its backplane Ethernet ports to their lowest possible link speed.
Advanced Power Management Wake Up Supports APM wake upper-PF: • Magic packet filter. When a magic packet arrives, and if the Magic packet filter is enabled, this can trigger a wake up.

Table 38-12. General Features

Description
Serial Flash Interface (shared SPI interface with other IPs in the 10 GbE controller)
Configurable LED operation for software or Original Equipment Manufacturer customization of LED displays. Default Configuration by Non-volatile Memory for all LEDs for pre-driver functionality
Device disable capability
Watchdog timer
Time Sync (IEEE 1588)
Firmware and Expansion ROM Management • All firmware and expansion ROM code is stored in flash memory and is field upgradeable • Supports field upgrades via host software utility or BMC program • Supports independent update of these objects: EMP firmware and, Protocol Engine RDMA firmware, Expansion ROM code. • Supports secure authentication of these objects during an update procedure: EMP firmware, Expansion ROM code and Protocol Engine RDMA firmware.These objects are protected with a digital signature produced using SHA256 hash and 2048b RSA encryption. • The firmware and expansion ROM field upgrade procedure can be carried out while the 10 GbE controller is in normal operation. The 10 GbE controller does not load and use the new content until a reset occurs. • Failure of the firmware or expansion ROM field upgrade procedure (due to interrupting of Flash programming, or authentication failure) never prevents a reattempt of the field upgrade procedure. Normally the 10 GbE controller reverts to the previously saved firmware or expansion ROM content when a field upgrade fails.
SDPs and support interfaces: • There are 8 SDPs, 8 LEDs, MDIO and I^2C. • SDPs are typically used to exchange information with or to control external devices (such as PHY devices) under the 10 GbE controller software driver control. • Each SDP can be configured as an input, output, or interrupt source.
Device / Port / PCI Function Disable The 10 GbE controller implements two strapping pins that, together with NVM settings, enable the entire 10 GbE Ethernet Controller, or selected Ethernet ports and their associated PCI functions or selected PCI functions to be disabled. Disabling occurs before PCI enumeration, so the disabled resources are completely hidden.

38.8.1 Numbers And Number Bases

Hexadecimal numbers are written with a 0x prefix (like 0xFFFF or 0x1234ABDF). Binary numbers are written with a lowercase b suffix (like 1001b or 10b).

38.8.2 Byte Ordering

This section defines the internal organization of registers and memory structures that span multiple bytes. A few conventions to start with are:

1. Network Order - Ethernet always transmits multiple-bytes fields with the Most Significant (MS) byte first
2. Endian notation - defines how a logical entity (such as a MAC address) is stored in a given structure (like register or descriptor). Two options exist:
   a. Little Endian notation — The MS byte of the logical entity is mapped to the highest byte address of the structure
   b. Big Endian notation — The MS byte of the logical entity is mapped to the lowest byte address of the structure

A few examples follow:

Example 1: A 32-bit counter is equal to 0x01234567 (such as the sequence number in the TCP header). The counter is transferred on the wire as: 01 23 45 67 where 01 is the first byte on the wire and 67 is the last byte.

LE registers store this counter as (in bytes) as follows:

0x01 - highest byte address

0x23

0x45

0x67 - lower byte address

BE registers store this counter as (in bytes) as follows:

0x67 - highest byte address

0x45

0x23

0x01 - lower byte address

Example 2: An L2 type register that holds the value of IPv4 header is equal to 0x0800. The field is transferred on the wire as: 08 00 where 08 is the first byte on the wire.

LE registers store this counter as (in bytes) as follows:

0x08 - highest byte address

0x00 - lower byte address

BE registers store this counter as (in bytes) as follows:

0x00 - highest byte address

0x08 - lower byte address

Example 3: A 48-bit Ethernet MAC address equals to 0x00112348A9BE. The Ethernet MAC address is transferred on the wire as: 00 11 23 48 A9 BE where 00 is the first byte on the wire.

LE registers store this counter as (in bytes) as follows:

0x00 - highest byte address

0x11

0x23

0x48

0xA9

0xBE - lower byte address

BE registers store this counter as (in bytes) as follows:

0xBE - highest byte address

0xA9

0x48

0x23

0x11

0x00 - lower byte address

The following rules determine the Endian-ness of the 10 GbE controller structures:

- The general rule is that all structures are defined in LE notation unless defined otherwise. These structures include:
  - Registers
  - AQ commands
  - Structures in host memory (including any type of descriptors)
  - NVM
  - LAN and PE contexts
- The following structures are in BE notation:
  - Host memory buffers that are received or transmitted
  - Any structures that contains a MAC address (see exception for field vector)
  - Quad hash context programming registers (GL_SWT_LOFV_PE, GL_SWT_LOFV_SW and EMP_SWT_LOFV) are each defined in LE, while register 'n' is mapped to words (63 - 2*'n') and (62 - 2 *'n') in the field vector
  - IP addresses in Protocol Engine host memory
- The following structures have a mixed notation:
  - Field vector is presented in mixed BE/LE notation: words are ordered in BE notation and bytes within the words are presented in LE notation.
  - Type-length-value structures are stored in the NVM in mixed BE/LE notation: words are ordered in BE notation and bytes within the words are presented in LE notation.

38.9 Overview Of Standards

Table 38-13 lists standards relevant to the 10 GbE controller.

Table 38-13. Standards Supported By the 10 GbE controller (Sheet 1 of 6)

Category	Description
base	Title: Computing the Internet Checksum Document: http://www.ietf.org/rfc/rfc1071.txt Description: This RFC describes how to compute the Internet checksums used in IP, TCP and UDP.
base	Title: A TCP/IP Tutorial Document: http://www.ietf.org/rfc/rfc1180.txt Description: Bare bones tutorial of TCP/IP protocol suite: ARP, IP and TCP and Upper Layer Protocols (ULPs). This is included for informative purposes only.
base	Title: Implementing the Internet Checksum in Hardware. Document: http: //www.ietf.org/rfc/rfc1936.txt Description: Techniques for efficiently implementing the Internet checksum in hardware.
Ethernet	Title: "Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications" (IEEE Std 802.3-2012) Document: available from standards.ieee.org/getieee802 Description: Specifies the Ethernet MAC and PHY layers up to 100 Gb/s. Includes these now superseded docs (among many others): 802.3ae (10 Gb/s base specification), 802.3an (10GBASE-T), 802.3ap (backplane Ethernet, KX and KR), etc.
Ethernet	Title: "Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications – Amendment 8: MAC Control Frame for Priority-based Flow Control" (IEEE P802.3bd-2011) Document: available from standards.ieee.org/getieee802 Description: Defines a MAC control frame to support 802.1Qbb priority-based flow control.

Table 38-13. Standards Supported By the 10 GbE controller (Sheet 2 of 6)

Category	Description
Ethernet	Title: "IEEE Standard for Local and Metropolitan Area Networks – Media access control (MAC) Bridges" (IEEE Std 802.1D-2004) Document: available from standards.ieee.org/getieee802 Description: Base specification for Ethernet bridging.
Ethernet	Title: "IEEE Standards for Local and Metropolitan Area Networks - Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks" (IEEE Std 802.1Q-2011) Document: available from standards.ieee.org/getieee802 Description: Ethernet VLAN-aware bridge specification.
Ethernet	Title: "IEEE Standard for Local and metropolitan area networks— Link Aggregation" (IEEE Std 802.1AX-2008) Document: available from standards.ieee.org/getieee802 Description: Logic and protocols that enable aggregation of one or more Ethernet links into a single logical link. Until recently, link aggregation was defined in the 802.3 specification, but in the 2008 version it was moved to 802.1.
Ethernet	Title: "Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks — Amendment17: Priority-based Flow Control" (IEEE P802.1Qbb-2011) Document: available from standards.ieee.org/getieee802 Description: Priority-based Flow Control (PFC) is one of the specifications that comprise DCB. PFC enables flow control per TC on IEEE 802 point-to-point full duplex links. This is achieved by a mechanism similar to the IEEE 802.3 Annex 31B PAUSE, but operating on individual priorities.
Ethernet	Title: "Virtual Bridged Local Area Networks — Amendment 18: Enhanced Transmission Selection for Bandwidth Sharing Between Traffic Classes" (IEEE P802.1Qaz-2011) Document: available from standards.ieee.org/getieee802 Description: ETS is one of the specifications that comprise DCB. ETS enables arbitration of bandwidth between TCs. This specification also defines Data Center Bridging Exchange (DCBX) protocol. DCBX enables configuration of DCB features onto an Ethernet LAN.
Ethernet	Title: "Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks — Amendment 21: Edge Virtual Bridging" (IEEE P802.1Qbg-2012) Document: available from standards.ieee.org/getieee802 Description: EVB is one of the specifications that comprise DCB. EVB defines many of the virtual switching features on end stations like the 10 GbE controller. This includes definition of things like S-channels, which enable the multiplexing of multiple virtual channels on a single physical LAN, VSIs, VEBs, VEPAs, etc. It also defines new management infrastructure for administering the new features.
Ethernet	Title: "Virtual Bridged Local Area Networks — Bridge Port Extension" (IEEE P802.1BR/D3.3) Document: available from standards.ieee.org/getieee802 Description: BPE is a dedicated specification describing the bridge port extender element. It specifies the use of E-channels and a multicast replication service to extend bridge ports across multiple physical or virtual devices.
Ethernet	Title: "IEEE Standard for Local and metropolitan area networks— Station and Media Access Control Connectivity Discovery" (IEEE Std 802.1AB-2009) Document: available from standards.ieee.org/getieee802 Description: Defines Link Layer Discovery Protocol (LLDP) that enables a server to advertise it's identity, capabilities, and interconnections to other entities on an Ethernet fabric.
Ethernet	Title: "IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems" (IEEE Std 1588-2008) Document: available from standards.ieee.org Description: Defines a protocol that enables precise synchronization of clocks in systems communicating via packet networks.
Ethernet	Title: "SFF-8436 Specification for QSFP+ Copper and Optical Modules, rev 3.1, 4/22/2009" Document: ftp://ftp.seagate.com/pub/sff/SFF-8436.PDF (see www.sffcommittee.com) Description: Defines the Quad Small Form Factor Pluggable (QSFP+) module mechanicals, electrical interface and pinout, etc.
Ethernet	Title: "SFF-8431 Specifications for Enhanced Small Form Factor Pluggable Module SFP+, rev 4.1, 7/6/2009" Document: ftp://ftp.seagate.com/sff/SFF-8431.PDF (or see www.sffcommittee.com) Description: Defines the SerDes Framer Interface (SFI) high speed electrical interface to a Small Form-factor Pluggable (SFP+) optical module. Also defines the SFP+ management interface.

Table 38-13. Standards Supported By the 10 GbE controller (Sheet 3 of 6)

Category	Description
Ethernet	Title: RMII Specification, rev 1.2, 3/20/1998 Description: Reduced pin count interface used in place of IEEE standard Media Independent Interface (MII). The 10 GbE controller has an Reduced Media Independent Interface (RMII) interface for its NC-SI connection.
Ethernet	Title: Ethernet Alliance, Ethernet Jumbo Frames, ver 0.3, 11/17/2009 Document: EA-Ethernet Jumbo Frames v0 3.pdf Title: Extended Frame Sizes for Next Generation Ethernets Document: AlteonExtendedFrames_W0601.pdf Title: Extended Ethernet Frame Size Support Document: draft-kaplan-isis-ext-eth-02.txt Description: Documents describing jumbo frames. Included for informative purposes only.
test	Title: IEEE Standard Test Access Port and Boundary-Scan Architecture (IEEE Std 1149.1-2001) Document: available from standards.ieee.org Description: Defines a standard interface through which instructions and test data are communicated to an integrated circuit for component- and circuit board-level testing.
Ethernet/ IP	Title: Standard for the Transmission of IP Datagrams over Ethernet Networks Document: http://www.ietf.org/rfc/rfc894.txt Description: This specifies the method for transmitting IP datagrams over Ethernet.
Ethernet/ IP	Title: Standard for the Transmission of IP Datagrams over IEEE 802 Networks Document: http://www.ietf.org/rfc/rfc1042.txt Description: This specifies the method for transmitting IP datagrams over IEEE 802.3 networks.
iARP	Title: Address Resolution Protocol (ARP) Document: http://www.ietf.org/rfc/rfc0826.txt Description: Protocol to convert IP addresses to Ethernet addresses
IP	Title: Internet Protocol Document: http://www.ietf.org/rfc/rfc0791.txt Description: Base specification for IPv4.
IP	Title: Internet Protocol, Version 6 Document: http://www.ietf.org/rfc/rfc2460.txt Description: Base specification for IPv6.
IP	Title: Neighbor Discovery for IP version 6 (IPv6) Document: http://www.ietf.org/rfc/rfc4861.txt Description: IPv6 nodes use neighbor discovery to discover each other's presence, to determine each other's link-layer addresses, to find routers, and to maintain reachability information. This is an important standard for power management.
IP	Title: TCP Processing of the IPv4 Precedence Field Document: http: //www.ietf.org/rfc/rfc2873.txt Description: Corrects a conflict between TCP as defined in RFC793 and DiffServ in handling of the IPv4 precedence field.
TCP	Title: Transmission Control Protocol Document: http://www.ietf.org/rfc/rfc0793.txt Description: Base specification for TCP.
TCP	Title: Window and Acknowledgment Strategy in TCP Document: http://www.ietf.org/rfc/rfc813.txt
TCP	Title: Congestion Control in IP/TCP Internet works Document: http://www.ietf.org/rfc/rfc896.txt Description: Defines the Nagle algorithm, which specifies delaying transmission of small amounts of data when there are Acknowledgments (ACKs) outstanding.
TCP	Title: TCP Extensions for High Performance Document: http://www.ietf.org/rfc/rfc1323.txt Description: Defines the TCP window scale and timestamp options, Round Trip Time Measurement (RTTM) and Protect Against Wrapped Sequences (PAWS).

Table 38-13. Standards Supported By the 10 GbE controller (Sheet 4 of 6)

Category	Description
TCP	Title: Known TCP Implementation Problems Document: http://www.ietf.org/rfc/rfc2525.txt Description: This RFC describes implementation issues with various historical TCP/IP stacks. While it is included here for informative purposes only, it does serve as a good check list to avoid known problems. The 10 GbE controller Protocol Engine is designed to support its recommendations.
TCP	Title: TCP Congestion Control Document: http://www.ietf.org/rfc/rfc5681.txt Description: TCP slow start, congestion avoidance, fast retransmit, and fast recovery algorithms.
TCP	Title: The NewReno Modification to TCP's Fast Recovery Algorithm Document: http://www.ietf.org/rfc/rfc3782.txt Description: Update the fast recovery algorithm that is run after three duplicate ACKs have been received. Changes to the CWND during fast recovery, scheduling additional fast retransmitted packets if the received ACKs do not acknowledge all the data when fast recovery was entered.
TCP	Title: TCP Problems with Path MTU Discovery Document: http://www.ietf.org/rfc/rfc2923.txt Description: Discusses problems with existing RFC 1191 implementations that should be avoided. Serves as an implementation checklist.
TCP	Title: Computing TCP's Retransmission Timer Document: http://www.ietf.org/rfc/rfc2988.txt Description: Defines the standard algorithm that TCP senders are required to use to compute and manage their retransmission timer.
TCP	Title: Increasing TCP's Initial Window Document: http://www.ietf.org/rfc/rfc3390.txt Description: Specifies an optional standard for TCP to increase the permitted initial window from one or two segments to roughly 4KB.
TCP	Title: TCP Congestion Control with Appropriate Byte Counting (ABC) Document: http://www.ietf.org/rfc/rfc3465.txt Description: This experimental RFC defines a small modification to the way TCP increases its congestion window. Instead of increasing cwnd by a constant amount for each acknowledgment, cwnd is increased based on the number of previously unacknowledged bytes each ACK covers.
UDP	Title: User Datagram Protocol Document: http://www.ietf.org/rfc/rfc0768.txt Description: Base specification for UDP.
Tunneling	Title: NVGRE: Network Virtualization using Generic Routing Encapsulation Document: http://datatracker.ietf.org/doc/draft-sridharan-virtualization-nvgre/?include_text=1 Description: MAC in Generic Routing Encapsulation (GRE) over IP encapsulation
Tunneling	Title: VXLAN: A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks Document: http://datatracker.ietf.org/doc/draft-mahalingam-dutt-dcops-vxlan/?include_text=1 Description: MAC in UDP encapsulation
Storage	Title: SCSI Block Commands - 3 (SBC-3), 11/25/2009, T10/1799-D Document: drafts available from http://www.t10.org/ Description: Defines the SCSI Block Commands - 3 (SBC-3) command set, which maintains a high degree of compatibility with the SBC-2 command set, INCITS 405-2005. Of particular interest to the 10 GbE controller, SBC-3 defines protection information (also known as *Data Integrity Field* or DIF), an industry standard for data integrity that protects data between a block storage initiator and the storage device.
RDMA	Title: Marker PDU Aligned Framing for TCP Specification Document: http://www.ietf.org/rfc/rfc5044.txt Description: LLP is TCP, ULP is DDP. MPA is a framing protocol that enables the preservation of ULP record boundaries.
RDMA	Title: Direct Data Placement over Reliable Transports Document: http://www.ietf.org/rfc/rfc5041.txt Description: LLP is MPA, ULP is RDMAP. DDP protocol provides information to place incoming data directly into a ULP receive buffer without intermediate buffers.
RDMA	Title: A Remote Direct Memory Access Protocol Specification Document: http://www.ietf.org/rfc/rfc5040.txt Description: LLP is DDP, connects directly to the Host CPU/OS via the 10 GbE controller System Interface.

Table 38-13. Standards Supported By the 10 GbE controller (Sheet 5 of 6)

Category	Description
RDMA	Title: RDMA Protocol Verbs Specification, v1.0, April 2003 Document: rdmaconsortium.org/home/draft-hilland-iwarp-verbs-v1.0-RDMAC.pdf Description: Verbs describe the Host application/OS interface to an RNIC. This interface is implemented as a combination of the RNIC system interface, its associated firmware, and host software. It provides access to the RNIC queuing and memory management resources, as well as the underlying networking layers.
Mgmt	Title: System Management Bus (SMBus) Specification, v3.0, 12/20/2014 Document: http://www.smbus.org/specs/SMBus_3_0_20141220.pdf Description: A two-wire interface for communication of management information, based on the principles of operation of I^2C.
Mgmt	Title: Network Controller Sideband Interface (NC-SI) Specification, v1.0.1, 1/10/2013 Document: dmtf.org/sites/default/files/standards/documents/DSP0222_1.0.pdf Description: Standardizes the sideband communication interface between a NIC (the 10 GbE controller) and a BMC.
Mgmt	Title: Management Component Transport Protocol (MCTP) Base Specification, rev 1.1.0, 4/22/2010 Document: dmtf.org/sites/default/files/standards/documents/DSP0236_1.1.0.pdf Description: Specifies MCTP protocol.
Mgmt	Title: Management Component Transport Protocol (MCTP) SMBus/I2C Transport Binding Specification, rev 1.0.0, 7/28/2009 Document: dmtf.org/sites/default/files/standards/documents/DSP0237_1.0.0.pdf Description: Describes the binding of MCTP over SMBus.
Mgmt	Title: Management Component Transport Protocol (MCTP) PCIe VDM Transport Binding Specification, rev 1.0.1, 12/11/2009 Document: dmtf.org/sites/default/files/standards/documents/DSP0238_1.0.1.pdf Description: Describes the binding of MCTP over PCIe.
Mgmt	Title: Management Component Transport Protocol (MCTP) IDs and Codes, rev 1.1.0, 11/3/2009 Document: dmtf.org/sites/default/files/standards/documents/DSP0239_1.1.0.pdf Description: Describes constants used by MCTP specifications.
Mgmt	Title: NC-SI Over MCTP Specification, rev 0.1, 8/17/2010 Document: dmtf.org/sites/default/files/standards/documents/DSP0261_1.0.pdf Description: Describes the encapsulation of NC-SI packets in MCTP.
Power Mgmt	Title: Magic Packet Technology, November 1995 Document: http://support.amd.com/TechDocs/20213.pdf Description: Defines a method for waking up a sleeping networked PC using a specific Ethernet frame (a Magic packet).
Power Mgmt	Title: PCI Bus Power Management Interface Specification, rev 1.2, 3/3/2004 Document: available from www.pcisig.com Description: Defines a standard set of PCI peripheral power management hardware interfaces and behavioral policies.
Power Mgmt	Title: proxZZZy for sleeping hosts, February 2010 (ECMA-393) Document: www.ecma-international.org/publications/files/ECMA-ST/ECMA-393.pdf Description: Defines a low-power proxy that handles key network tasks for a high-power device, thus allowing the high-power device to sleep when not in active use.
Statistics	Title: The Interfaces Group MIB Document: http://www.ietf.org/rfc/rfc2863.txt Description: Describes objects used for managing network interfaces.
Statistics	Title: Remote Network Monitoring MIB Document: http://www.ietf.org/rfc/rfc2819.txt Description: Defines objects for managing remote network monitoring devices. Includes the popular packet size histogram counters.
Statistics	Title: Microsoft NDIS 6.0 OID_GEN_STATISTICS Document: msdn.microsoft.com/en-us/library/ff569640(VS.85).aspx Description: Adapter statistics counters defined by Microsoft for NDIS 6.0.
Statistics	Title: MIB for the Internet Protocol (IP) Document: http://www.ietf.org/rfc/rfc4293.txt Description: IP version-independent IP MIB. Obsoletes RFC 2011, 2465, 2466.

Table 38-13. Standards Supported By the 10 GbE controller (Sheet 6 of 6)

Category	Description
Statistics	Title: MIB for the Transmission Control Protocol (TCP) Document: http://www.ietf.org/rfc/rfc4022.txt Description: IP version-independent TCP MIB. Obsoletes RFC 2012, 2452.
Statistics	Title: MIB for the User Datagram Protocol (UDP) Document: http://www.ietf.org/rfc/rfc4113.txt Description: Objects for managing implementations of UDP. Obsoletes RFC 2013.
Statistics	Title: Microsoft kernel-mode Network Direct requirements Document: typedef struct _NDIS_NDK_PERFORMANCE_COUNTERS Description: Protocol Engine statistics are defined to support Microsoft kernel-mode Network Direct requirements.
PCI	Title: PCI Local Bus Specification, rev 3.0, 2/3/2004 Document: available from www.pcisig.com Description: Compliant with select sections, such as Appendix I Vital Product Data.
PCI	Title: PCI Hot-Plug Specification, rev 1.1, 6/20/2001 Document: available from www.pcisig.com Description: Defines how PCI add-in cards are installed and removed while the system is running.
PCI	Title: PCI Firmware Specification, rev 3.0, 6/20/2005 Document: available from www.pcisig.com Description: Defines the firmware interface for managing PCIe systems in a host computer. Describes the format, contents, and code entry points for expansion ROMs.
PCI	Title: PCI Express Base Specification, rev 3.0, 11/10/2010 Document: available from www.pcisig.com Description: Contains the technical details of the PCIe architecture, protocol, link layer, physical layer, and software interface. Also defines some important power management features, including Optimized Buffer Flush/ Fill (OBFF).
PCI	Title: PCI Express Card Electromechanical Specification, rev 2.0, 4/11/2007 Document: available from www.pcisig.com Description: Mechanical and electrical specifications for an Evo card form factor.
PCI-IOV	Title: Single Root I/O Virtualization and Sharing Specification, rev 1.1, 1/20/2010 Document: available from www.pcisig.com Description: The SR-IOV specification defines extensions to the PCIe specification that enable the VMs in a virtualized server to efficiently share PCI adapter hardware resources.
software	Title: Microsoft specification for “Receive Side Scaling” (RSS) from MSDN. Document: msdn.microsoft.com/en-us/library/ff567236%28v=VS.85%29.aspx Description: RSS is a network driver technology that enables the efficient distribution of network receive processing across multiple CPUs in multiprocessor systems.

38.10 PCI-Express (PCIe)

38.10.1 Requirements

The 10 GbE controller supports Rev. 3.0 of the PCIe base specification.

In addition to the capabilities required by the PCIe specifications, the 10 GbE controller also supports the following optional functionality as described in this section:

- All PCI functions are native PCIe functions
- Transaction layer mechanisms
  - 64-bit and 32-bit memory address spaces
  - Removal of I/O BAR (optional)
  - Relaxed ordering
  - Flow control update timeout mechanism

  - ID-based ordering (IDO)
  - Packet sizes: Maximum payload size: 2 KB, Maximum read request size: 4 KB
  - Extended tags
  - Function-Level Reset (FLR)
  - TLP Processing Hints (TPH)
- Reliability
  - Advanced Error Reporting (AER)
  - End-to-End CRC (ECRC) generation and checking
  - Recovery from data poisoning
  - Completion timeout
- Power management:
  - Active state power management (L0s and L1 states)
  - Wake capability
- DFT and DFM support for high-volume manufacturing
- The 10 GbE controller supports the following extended capabilities:
  - AER
  - Device Serial Number (DSN)
  - Alternative RID Interpretation (ARI)
  - Single Root I/O Virtualization (SR-IOV)
  - TPH Requester
  - Access Control Services (ACS)

38.10.2 Transaction Layer

38.10.2.1 Size Of Target Accesses

38.10.2.1.1 Memory Accesses

Rules for accesses to the CSR space (both memory BAR and MSI-X BAR):

- Write accesses:
  - CSR writes are 32 bit or 64 bit only
  - Zero-length writes have no internal impact (nothing written, no effect such as clear-by-write). The transaction is treated as a successful operation (no error event)
  - Other accesses (partial writes, larger writes) are handled as completer abort - data is dropped and an error is generated per PCIe rules
- Read accesses:
  - CSR reads are 32 bit or 64 bit only
    - Some 64-bit reads are handled atomically such as not interleaved with any other read requests. This applies mainly to reading counters, where all 64 bits need to be read simultaneously. Such registers are explicitly marked in their description.
  - Partial reads with at least one byte disabled are handled internally as a full read. That is, any side effect of the full read (such as clear by read) is also applicable to partial reads. The completion on PCIe adheres to the specification rules regarding the number of bytes reported in the completion.

  - Zero-length reads generate a completion, but the register is not accessed and undefined data is returned
  - Larger CSR read requests are handled as completer abort - the completion includes a CA status and an error is generated per PCIe rules

Rules for accessing the doorbell space in the memory BAR:

- Write accesses:
  - Write accesses are supported within the limit of the respective PCIe credits
  - A write access that crosses over into the Flash address space is handled as a completer abort
  - Zero-length writes have no internal impact (nothing written, no effect such as clear-by-write). The transaction is treated as a successful operation (no error event)
- Read accesses:
  - Read accesses are replied with stale data. No error indication is provided

Rules for accessing the Flash space in the memory BAR or the expansion ROM BAR:

- Write accesses:
  - Writes to Flash are 8-bit wide only
  - Any larger write accesses are handled as completer abort - data is dropped and an error is generated per PCIe rules
- Read accesses:
  - Reads to Flash are 32-bit wide
  - Partial reads with at least one byte disabled are handled internally as a full read. That is, any side effect of the full read (such as clear by read) is also applicable to partial reads. The completion on PCIe adheres to the specification rules regarding the number of bytes reported in the completion
  - Zero-length reads generate a completion, but the Flash is not accessed and undefined data is returned
  - Larger CSR read requests are handled as completer abort - the completion includes a CA status and an error is generated per PCIe rules

38.10.2.2 I/O Accesses

Rules for accesses to the I/O BAR:

- Write accesses:
  - Write accesses are 32-bit wide
  - Zero-length writes have no internal impact (nothing written, no effect such as clear-by-write). The transaction is treated as a successful operation (no error event)
  - Other accesses (partial writes, larger writes) are handled as completer abort - data is dropped and an error is generated per PCIe rules
- Read accesses:
  - Reads to the I/O BAR are 32-bit wide
  - Partial reads with at least one byte disabled are handled internally as a full read. That is, any side effect of the full read (such as clear by read) is also applicable to partial reads. The completion on PCIe adheres to the specification rules regarding the number of bytes reported in the completion

— Larger CSR read requests are handled as completer abort - the completion includes a CA status and an error is generated per PCIe rules

38.10.2.2.1 Messages

MCTP messages might contain a payload of up to 64 bytes.

38.10.2.2.2 Support For Dynamic Changes

The 10 GbE controller captures the bus number and device number per each configuration write request. However, a dynamic change of the bus number or device number is not supported. Rather, the PCIe link should be quiescent prior to such a change, including reception of all completion for previous requests.

38.10.2.3 Transactions Initiated By the 10 GbE controller

Note: If Extended tags are not enabled, the performance of the device may degrade. In order to ensure usage of extended tags, all functions must enable them in the Extended Tag field Enable field.

38.10.2.3.1 Data Alignment

Requests must never specify an address/length combination that causes a memory space access to cross a 4 KB boundary. The 10 GbE controller therefore breaks requests into 4 KB-aligned requests (if needed). This does not pose any requirement on software. However, if software allocates a buffer across a 4 KB boundary, hardware issues multiple requests for the buffer. Software should consider aligning buffers to a 4 KB boundary in cases where it improves performance.

The general rules for packet alignment are as follows. Note that these apply to all 10 GbE Controller requests:

- The length of a single request does not exceed the PCIe limit of MAX_PAYLOAD_SIZE for write and MAX_READ_REQUEST_SIZE for read.
- A single request does not span across different memory pages as noted by the 4 KB boundary alignment previously mentioned.

If a request can be sent as a single PCIe packet and still meets the general rules for packet alignment, then it is not broken at the cache line boundary but rather sent as a single packet. However, if any of the general rules require that the request is broken into two or more packets, then the request is broken at the cache line boundary.

For requests with data payload, if the payload size is larger than (MAX_PAYLOAD_SIZE - CACHELINE_SIZE), then the request is broken into multiple TLPs staring at the first cache-line boundary following the (MAX_PAYLOAD_SIZE - CACHELINE_SIZE) bytes. For example, if MAX_PAYLOAD_SIZE = 256B and CACHELINE_SIZE = 64 bytes, a 1 KB request starting at address 0x...10 is broken into TLPs such that the first TLP contains 240bytes of payload (since 240bytes + 0x10 = 256bytes is on the cache-line boundary).

The system cache line size is controlled by the GLPCI_CNF2.CACHELINE_SIZE bit, loaded from the NVM Cache Line Size field. Note that the Cache Line Size register in the PCI configuration space is not related to the GLPCI_CNF2.CACHELINE_SIZE and is solely for software use.

38.10.2.4 Messages

Table 38-14 lists the response to messages sent to the device. Unlisted messages are not supported by the device and are treated as an unsupported request.

Table 38-14. Messages In the 10 GbE controller (As a Receiver)

Message Code [7:0]	Routing r2r1r0	Message	10 GbE Controller Response
0x00	011b	Unlock	Silently drop.
0x40 0x41 0x43 0x44 0x45 0x47 0x48	100b	Ignored messages (used to be hot-plug messages)	Silently drop.
0x50	100b	Slot power limit support (has one Dword data)	Silently drop.
0x7E	000b 010b 011b 100b	Vendor_defined type 0	Drop and handle as an unsupported request.
0x7F	100b	Vendor_defined type 1	Silently drop.
0x7F	000b 010b 011b	Vendor_defined type 1 See Section 38.10.2.4.1	Send to MCTP reassembly if Vendor ID = 0x1AB4 (DMTF) and VDM code - 0000b (MCTP). Otherwise, silently drop.

Table 38-15 lists the messages sent by the device.

Table 38-15. Messages In the 10 GbE controller (As a Transmitter)

Message code [7:0]	Routing r2r1r0	Message
0x7F	000b 010b 011b	Vendor Defined Messages (VDM). See Section 38.10.2.4.1.

38.10.2.4.1 VDM

The following vendor defined message is supported: DMTF MCTP.

MCTP VDMs

MCTP VDMs are supported as both master and target. The following header fields are involved:

- Fmt - Set to 11b to indicate a 4-Dword header with data
- Type:
  - [4:3] - Set to 10b to indicate a message
  - [2:0] - Routing r2r1r0 = 000b, 010b or 011b
- Traffic Class - Set to 000b
- TLP Digest - Set to 0b (no ECRC)
- Error Present - Set to 0b
- Attributes[1:0] - Set to 01b (no snoop)

- Tag field - Indicates this is an MCTP packet and the size of padding to Dword alignment added
- Message code = 0x7F (Type 1 VDM)
- Destination ID - captures the target B/D/F for route by ID. Otherwise, reserved
- Vendor ID = 0x1AB4 (DMTF)

38.10.2.5 Transaction Attributes

Figure 38-4. Traffic Class (TC) and Virtual Channels (VC)

The 10 GbE controller only supports TC = 0b and VC = 0b (default).

38.10.2.5.1 TLP Processing Hints (TPH)

The 10 GbE controller supports the TPH capability defined in the PCI Express specification. It does not support extended TPH requests.

Existence of a TLP Process Hint (TPH) is indicated on the PCIe link by setting the TH bit in the TLP header. Using the PCIe TLP Steering Tag (ST) and Processing Hints (PH) fields, the 10 GbE controller can provide hints to the root complex about the destination (socket ID) and about data access patterns (locality in cache) when executing DMA memory writes or read operations.

The 10 GbE controller exposes a PCIe TPH capability structure with no steering table.

Required steps to enable TPH usage:

1. For a given function, the "*TPH Requester Enable"* field in the PCIe configuration *TPH Requester Control register* should be set to either 01b or 11b and the "ST Mode Select" field should be set to one of the two supported values: 000b (No Table Mode) or 010b (Device Specific Mode). If this is not the case, *the PF driver should not enable the TPH in the transmit and receive queue contexts*.
2. Appropriate TPH enable bits in the receive or transmit queue context should be set.
3. Processing hints should be programmed in the *GLTPH_CTRL.Desc_PH* and *GLTPH_CTRL.Data_PH* Processing Hints (PH) fields.
4. Steering information should be programed in the CPUID fields in the receive or transmit queue context.

Note: In order to enable TPH usage, all the memory reads are done without setting any of the byte enable bits.

Steering tag and processing hint programming

Each type of DMA traffic uses a different policy to define how the steering tag (socket ID) and processing hints are generated:

- The policy for LAN traffic.
- The policy for offloaded traffic.
- Accesses to the host memory cache do not use TPH hints.
- Accesses to the admin command queues do not use TPH hints.

38.10.2.6 Device Ordering Rules

The 10 GbE controller meets the PCIe ordering rules as follows:

Deadlock avoidance – The 10 GbE controller meets the PCIe ordering rules that prevent deadlocks:

1. Posted writes overtake stalled read requests. This applies to both target and master directions. For example, if master read requests are stalled due to lack of credits, master posted writes are allowed to proceed. On the target side, it is acceptable to timeout on stalled read requests in order to allow later posted writes to proceed.
2. Target posted writes overtake stalled target configuration writes.
3. Completions overtake stalled read requests. This applies to both target and master directions. For example, if master read requests are stalled due to lack of credits, completions generated by the 10 GbE controller are allowed to proceed.

Consistency of data:

1. Descriptor/Data Ordering — The 10 GbE controller insures that a Rx descriptor is written back on PCIe only after the data that the descriptor relates to is written to the PCIe link.
2. Target NP read requests might pass target posted writes addressing different PCI functions.
3. Completions for target reads (memory, I/O, configuration) do not pass previous posted requests. Here are some specific usages of this rule:
   — Flush following a reset (such as FLR, BME, D3 entry, VFE clear) - When the system issues a reset event, it needs to identify when the device stops sending new posted requests from the function(s) under reset. The completion to the config write of these reset events will be sent after all requests/completions related to that function(s) are flushed out. The device is expected not to issue any new posted transactions from the function(s) under reset.
   — MSI and MSI-X Ordering Rules – System software can change the MSI or MSI-X tables during run-time. Software expects that interrupt messages issued after the table has been updated are using the updated contents of the tables.
      - Since software does not know when the tables are actually updated in the 10 GbE controller, a common scheme is to issue a read request to the MSI or MSI-X table after an update to the table (a PCI configuration read for MSI and a memory read for MSI-X). Software expects that any message issued following the completion of the read request, is using the updated contents of the tables.
      - Once an MSI or MSI-X message is issued using the updated contents of the interrupt tables, any consecutive MSI or MSI-X message does not use the contents of the tables prior to the change.

Independence between target and master accesses:

1. The acceptance of a target posted request does not depend upon the transmission of any TLP.
2. The acceptance of a target non-posted request does not depend upon the transmission of a non-posted request.
3. Accepting a completion does not depend upon the transmission of any TLP.

38.10.2.6.1 Processing Of Target Accesses

The 10 GbE controller meets the specification requirements regarding target accesses as described in the previous section.

In addition, the following behaviors apply:

- Target accesses from different functions might be processed in a different order than the order they arrive
- Completions that belong to requests from different PCI functions might be issued in a different order than the order of the respective requests.

38.10.2.6.2 Relaxed Ordering

The 10 GbE controller takes advantage of the relaxed ordering rules in PCIe. By setting the relaxed ordering bit in the packet header, the 10 GbE controller enables the system to optimize performance in the following cases:

1. Relaxed ordering for LAN descriptor and data reads — When the 10 GbE controller issues a read transaction, its split completion has no ordering relationship with the writes from the CPUs (same direction). It should be allowed to bypass the writes from the CPUs.
   - The *GLLAN_RCTL.RXDESCRDROEN* bit (loaded from NVM) enables relaxed ordering for Rx descriptor reads.
   - The *GLLAN_TCTL.TXDESCRDROEN* bit (loaded from NVM) enables relaxed ordering for Tx descriptor reads.
   - The *GLLAN_TCTL.TXDATARDROEN* bit (loaded from NVM) enables relaxed ordering for Tx data reads.
2. Relaxed ordering for LAN Rx data writes — When the 10 GbE controller issues Rx data writes, it also enables them to bypass each other in the path to system memory because software does not process this data until their associated descriptor writes are done.
   - The *GLLAN_RCTL.RXDATAWRROEN* bit (loaded from NVM) enables relaxed ordering for Rx data writes.
3. The 10 GbE controller does not relax ordering for the following requests:
   - LAN descriptor writes
   - LAN Tx head write back
   - Interrupt messages
   - MCTP messages
   - Protocol Engine traffic
   - HMC requests
   - EMP requests
   - Any other requests not previously mentioned

Relaxed ordering is globally enabled in the 10 GbE controller by clearing the *GLPCI_CNF2.RO_DIS* bit, originally loaded from NVM. It is further controlled through the *Enable Relaxed Ordering* bit in the PCIe Device Control register.

38.10.2.6.3 Id-based Ordering (Ido)

ID-based ordering was introduced in the PCIe rev. 2.1 specification. When enabled, the 10 GbE controller sets IDO in all applicable TLPs defined in the PCIe specification. IDO is not set for MCTP packets.

IDO is enabled when all of the following conditions are met:

- IDO is not disabled from the NVM. Device default is enabled. The value loaded from the NVM is reflected in the GLPCI_CAPSUP register.
- The PCIe *IDO Request Enable* bit (for requests) or the *IDO Completion Enable* bit (for completions) in the Device Control 2 register is set.

38.10.2.7 End-to-End CRC (ECRC)

The 10 GbE controller supports ECRC as defined in the PCIe specification. The following functionality is provided:

- Inserting ECRC in transmitted TLPs:
  - The 10 GbE controller indicates support for inserting ECRC in the *ECRC Generation Capable* bit of the PCIe Configuration registers. This bit is loaded from the global *ECRC Generation Capable* NVM bit.
  - Inserting ECRC is enabled per function by the *ECRC Generation Enable* bit of the PCIe Configuration registers. VFs follow the behavior of their PF.
  - ECRC is not added to MCTP messages (per the MCTP specification).
- ECRC is checked on all incoming TLPs. A packet received with an ECRC error is dropped. Note that for completions, a completion timeout occurs later (if enabled).
  - The 10 GbE controller indicates support for ECRC checking in the *ECRC Check Capable* bit of the PCIe Configuration registers. This bit is loaded from the global *ECRC Check Capable* NVM bit.
  - Checking of ECRC is enabled by the *ECRC Check Enable* bit of the PCIe Configuration registers. ECRC checking is done if enabled by at least one physical function (enablement is not done via VFs).
- ECRC errors are reported on all Physical Functions (PFs) enabled for ECRC checking.
- System software can configure ECRC independently per each physical function.

38.10.3 Error Events And Error Reporting

This section describes error reporting and advanced error reporting.

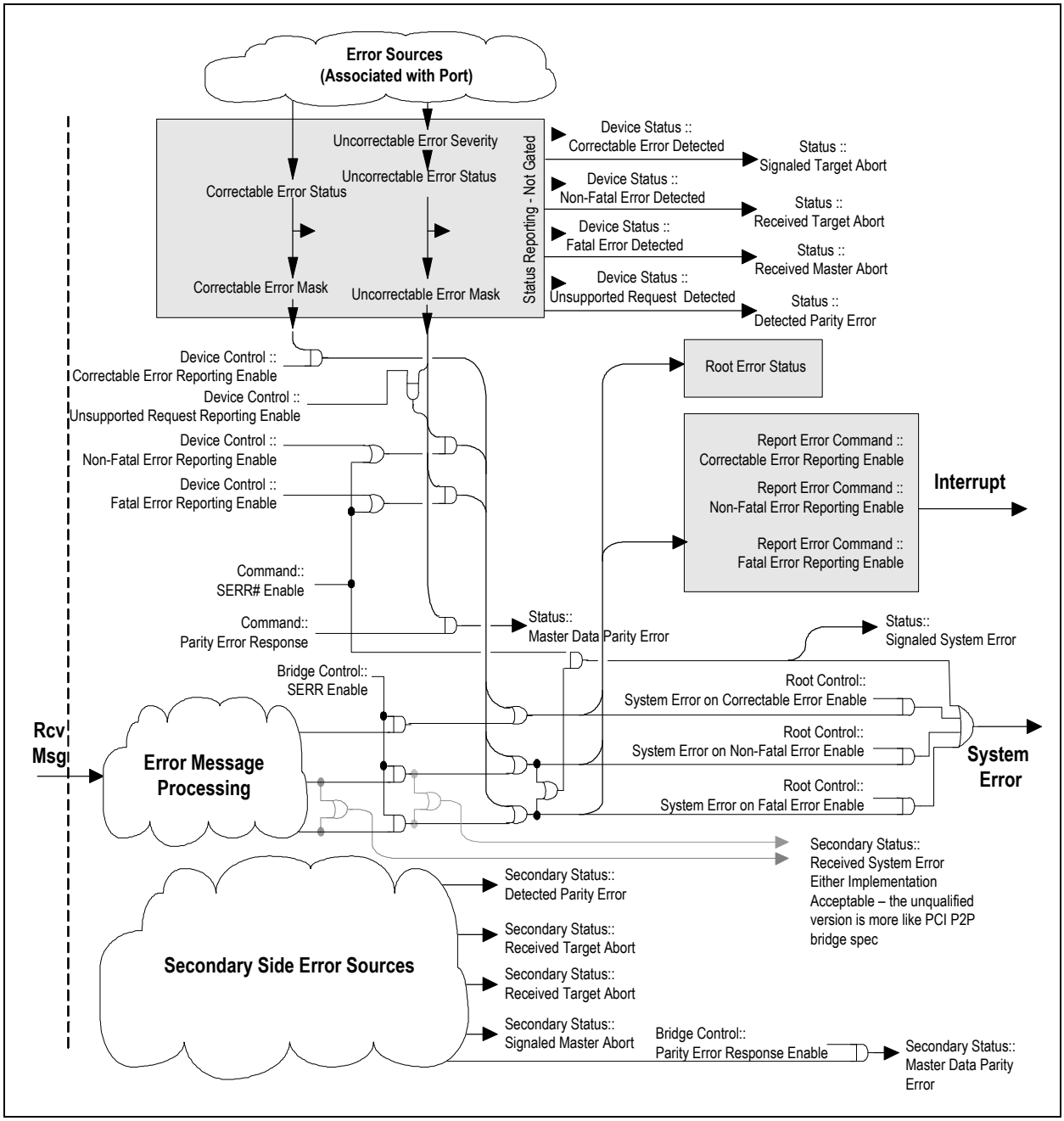
38.10.3.1 General Description

PCIe defines two error reporting paradigms: the baseline capability and the Advanced Error Reporting (AER) capability. The baseline error reporting capabilities are required of all PCIe devices and define the minimum error reporting requirements. The AER capability is defined for more robust error reporting and is implemented with a specific PCIe capability structure. Both mechanisms are supported by the 10 GbE controller, but the AER capability needs to be enabled in the NVM.

The *SERR# Enable* and the *Parity Error* bits from the Legacy Command register also take part in the error reporting and logging mechanism.

In a multi-function device, PCIe errors that are not related to any specific function within the device are logged in the corresponding status and logging registers of all functions in that device (see Section 6.2.4 in the *PCIe Base Specification*). Figure 38-5 shows, in detail, the flow of error reporting in PCIe. See also Figure 6-2 in the PCIe base specification.

Figure 38-5. Error Reporting Mechanism



38.10.3.2 Error Events

Table 38-16 lists the error events identified by the 10 GbE controller and the response in terms of logging, reporting, and actions taken. Refer to the PCIe specification for the effect on the PCI Status register. All the errors reported in SoCs are related to the Transaction layer.

Table 38-16. Response and Reporting Of PCIe Error Events (Sheet 1 of 2)

Error Name	Error Events	Default Severity	Action
Poisoned TLP Received	TLP With Error Forwarding	Uncorrectable ERR_NONFATAL Log Header	See Section 38.10.3.4 for more details. If error is defined as non-fatal (default severity): • Treat as an advisory non-fatal error: Send an ERR_COR Message If error is defined as fatal: • Send ERR_FATAL message
ECRC Check Failed	Failed ECRC check	Uncorrectable ERR_NONFATAL Log Header	See Section 38.10.2.7 for more details. If error is defined as non-fatal (default severity): • Send an ERR_NONFATAL Message If error is defined as fatal: • Send ERR_FATAL message
Unsupported Request (UR)	Receipt of TLP with unsupported Request Type Receipt of an Unsupported Vendor Defined Type 0 Message Invalid Message Code Wrong Function Number Received TLP Outside BAR Address Range Receipt of a Request TLP during $D3_{HOT}$, other than Configuration and Message requests	Uncorrectable ERR_NONFATAL Log header	Send Completion With UR f error is defined as non-fatal (default severity): • Treat as an advisory non-fatal error: Send an ERR_COR Message If error is defined as fatal: • Send ERR_FATAL message
Completion Timeout	Completion Timeout Timer Expired	Uncorrectable ERR_NONFATAL	See Section 38.10.3.3 for more details. If error is defined as non-fatal (default severity): • Treat as an advisory non-fatal error: Send an ERR_COR Message If error is defined as fatal: • Send ERR_FATAL message
Completer Abort	Received Target Access with illegal data size per Section 38.10.2.1	Uncorrectable. ERR_NONFATAL Log header	Send completion with CA If error is defined as non-fatal (default severity): • Treat as an advisory non-fatal error: Send an ERR_COR Message If error is defined as fatal: • Send ERR_FATAL message

Table 38-16. Response and Reporting Of PCIe Error Events (Sheet 2 of 2)

Error Name	Error Events	Default Severity	Action
Unexpected Completion	Received Completion Without a Request For It (Tag, ID, etc.)	Uncorrectable ERR_NONFATAL Log Header	Discard TLP If error is defined as non-fatal (default severity): • Treat as an advisory non-fatal error: Send an ERR_COR Message If error is defined as fatal: • Send ERR_FATAL message
Receiver Overflow	Received TLP Beyond Allocated Credits	Uncorrectable ERR_FATAL	Receiver Behavior is Undefined
Flow Control Protocol Error	Minimum Initial Flow Control Advertisements Flow Control Update for Infinite Credit Advertisement	Uncorrectable. ERR_FATAL	Receiver Behavior is Undefined
Malformed TLP (MP)	Data Payload Exceed Max_Payload_Size Received TLP Data Size Does Not Match Length Field TD field value does not correspond with the observed size PM messages that do not use TC0. Usage of Unsupported VC Target request crosses a 4KB boundary	Uncorrectable ERR_FATAL Log Header	Drop the Packet, Free FC Credits
Completion with Unsuccessful Completion Status		No Action (already done by originator of completion)	Free FC Credits

38.10.3.3 Completion Timeout Mechanism

The 10 GbE controller supports completion timeout as defined in the *PCIe Specification*.

The 10 GbE controller controls the following aspects of completion timeout:

- Disabling or enabling completion timeout
  - The PCIe *Completion Timeout Disable Supported* bit in the Device Capabilities 2 register is hardwired to 1b to indicate that disabling completion timeout is supported
  - The PCIe *Completion Timeout Disable* bit in Device Control 2 register controls whether completion timeout is enabled
- A programmable range of timeout values
  - The 10 GbE controller supports all four ranges as programmed in the *Completion Timeout Ranges Supported* field of the Device Capabilities 2 register. The actual completion timeout value is written in the *Completion Timeout Value* field of Device Control 2 register.

The following sequence takes place when completion timeout is detected:

- The appropriate message is sent on PCIe as listed in Table 38-16.
- The affected queue or client takes action based on the nature of the original request:
  - If the original request was for Tx packet data, the request and any partial packet completions are dropped

  - — Else, the request is handled same way as malicious requests. An interrupt is issued to the respective PF.
- Software might identify the source of the event (whether due to TLP poisoning, to a completion timeout, or an actual malicious event) by reading the error reporting counters or the performance and statistics counters.

38.10.3.4 Error Forwarding (TLP Poisoning)

If a TLP is received with an error-forwarding trailer, the packet is dropped and is not delivered to its destination.

The following sequence takes place when a poisoned TLP is received:

- The appropriate message is sent on PCIe as listed in Table 38-16.
- An interrupt is issued.
- If the TLP is a completion, a completion timeout follows at some later time. Processing continues as described in Section 38.10.3.3.

System logic is expected to trigger a system-level interrupt to inform the operating system of the problem. Operating systems can then stop the process associated with the transaction, re-allocate memory to a different area instead of the faulty area, etc.

38.10.3.5 Completion With Unsuccessful Completion Status

A completion arriving with unsuccessful completion status (either UR or CA) is dropped and not delivered to its destination. A completion timeout follows at some later time. Processing continues as described in Section 38.10.3.3.

38.10.3.6 Error Pollution

Error pollution can occur if error conditions for a given transaction are not isolated to the error's first occurrence. If the PHY detects and reports a receiver error, to avoid having this error propagate and cause subsequent errors at the upper layers, the same packet is not signaled at the data link or transaction layers. Similarly, when the data link layer detects an error, subsequent errors that occur for the same packet are not signaled at the transaction layer.

38.10.3.7 Blocking on Upper Address

The *GLPCI_UPADD* register blocks master accesses from being sent out on PCIe if the TLP address exceeds some upper limit. Bits [31:1] correspond to bits [63:33] in the PCIe address space, respectively.

When a bit is set in *GLPCI_UPADD[31:1]*, any transaction, in which the corresponding bit in its address is set, is blocked and not sent over PCIe. If all register bits are cleared, there is no effect (for example, a packet is sent unconditionally).

Processing a blocked transaction:

- Write transaction
  - — The transaction is dropped.
  - — Set the exceeded upper address limit (write requests) event in the PCIe errors register.
  - — An interrupt is issued.
- Read transaction

  - The transaction is dropped.
  - Set the exceeded upper address limit (read requests) event in the PCIe errors register.
  - The originating internal client is notified.
  - The affected queue or client takes action based on the nature of the original request. An interrupt is issued to the respective PF.

38.10.4 Performance And Statistics Counters

The 10 GbE controller incorporates counters to track the behavior and performance of the PCIe interconnect. The 10 GbE controller implements several types of counters:

- Transaction layer event counters - Section 38.10.4.1
- Link and Physical layer event counters - Section 38.10.4.2
- Latency counters - Section 38.10.4.3
- Bandwidth counters

General characteristics of the counters:

- Software can reset, stop, or start the counters.
- The counters are shared by all PCI functions (service mode of sharing).

Part of the registers that manage the operation of the performance counters are accessed via the GLPCI_LCBADD and GLPCI_LCBDATA register pair.

Reading a register via the GLPCI_LCBADD/GLPCI_LCBDATA pair is done as follows:

- Write the following values into the GLPCI_LCBADD register:
  - ADDRESS - the 18-bit register address. See below for the specific address per each register
  - BLOCK_ID - Defines the sub-unit where the register resides. Use the value 0x7F to access registers mentioned in this section
  - LOCK - use if need to gain access in case of multiple agents accessing the GLPCI_LCBADD/GLPCI_LCBDATA registers
- Read the GLPCI_LCBDATA register.
  - Note that although GLPCI_LCBDATA is a 32-bit register, the registers that maintain the actual count are read as atomic 64-bit reads. The GLPCI_LCBADD contains the address of the low DW, and reading GLPCI_LCBDATA returns a 64-bit value.

Writing a register via the GLPCI_LCBADD/GLPCI_LCBDATA pair is done as follows:

- Write the following values into the GLPCI_LCBADD register:
  - ADDRESS - the 18-bit register address. See below for the specific address per each register
  - BLOCK_ID - Defines the sub-unit where the register resides. For actual values, consult the text below
  - LOCK - use if need to gain access in case of multiple agents accessing the GLPCI_LCBADD/GLPCI_LCBDATA registers
- Write to the GLPCI_LCBDATA register.

38.10.4.1 Event Counters - Transaction Layer

Counters operate in one of the following modes:

- Count mode — the counter increments when the respective event occurred
- Leaky bucket mode — the counter increments only when the rate of events exceeded a certain value. See Section 38.10.4.1.2 for more details.

The list of events supported by the 10 GbE controller are listed in Table 38-17.

Table 38-17. PCIe* Statistic Events Encoding

Events	Event Mapping (Hex)	Description
Cycles	0x00	Increment on each integrated I/O clock tick.
Transaction Layer events		
Bad Request TLPs	0x10	Number of bad TLPs arriving to the transaction layer. These include: • Request caused UR. • Request caused CA. • Malformed TLP.
Bad Completions	0x11	Number of bad completions received. These include: • Unexpected completion. • UR status. • CA status.
Completion Timeout	0x12	Number of completion timeout events.
Poisoned TLP	0x13	Number of TLPs received with poisoned data or data parity.
ECRC Check	0x14	Number of TLPs that fail ECRC check.

38.10.4.1.1 Count Mode

The following CSR fields control operation of the Count mode:

- Four 32-bit counters GLPCI_GSCN_0_3 track events and increment on each occurrence of an event.
  - The four 32-bit counters can also operate in a two 64-bit mode to count long intervals or large payloads.
    - Registers *GLPCI_GSCN_0_3[0]* and *GLPCI_GSCN_0_3[1]* form the first 64-bit counter. Registers *GLPCI_GSCN_0_3[2]* and *GLPCI_GSCN_0_3[3]* form the second 64-bit counter.
    - The GLPCI_GSCL_1.GIO_64_BIT_EN selects between 32-bit and 64-bit modes.
- The GLPCI_GSCL_1.GIO_COUNT_EN_[3:0] bits enable each of the 4 counters.
  - The enable bits for the two 64-bit counters are *GLPCI_GSCL_1.GIO_COUNT_EN_0* and *GLPCI_GSCL_1.GIO_COUNT_ EN_2*, respectively.
- The GLPCI_GSCL_1.GIO_COUNT_START bit starts event counting of enabled counters.
- The GLPCI_GSCL_1.GIO_COUNT_STOP bit stops event counting of running counters.
- The GLPCI_GSCL_1.GIO_COUNT_RESET bit resets the event counters.
- The GLPCI_GSCL_2 associates an event with each of the 4 counters.
  - In 64-bit mode, the GIO_EVENT_NUM_[2,0] fields are used.

38.10.4.1.2 Leaky Bucket Mode

Each of the counters can be configured independently to operate in a leaky bucket mode. When in leaky bucket mode, the following functionality is provided:

- One of four 16-bit Leaky Bucket Counters (LBC) is enabled via the *LBC_ENABLE_[3:0]* bits in the PCIe Statistic Control register #1.
- The LBC is controlled by the *GIO_COUNT_START, GIO_COUNT_STOP, GIO_COUNT_RESET* bits in the PCIe Statistic Control register #1.
- The LBC increments every time the respective event occurs.
- The LBC is decremented every T ms as defined in the LBC_TIMER_N field in the PCIe Statistic Control registers #5...#8 (*GLPCI_GSCL_5_8*).
- When an event occurs and the value of the LBC meets or exceeds the threshold defined in the LBC_THRESHOLD_N field in the PCIe Statistic Control registers #5...#8 (*GLPCI_GSCL_5_8*), the respective statistics counter increments, and the LBC counter is cleared to zero.

38.10.4.2 Event Counters - Link And Physical Layers

This section describes the performance events for the Link and Physical layers and how to manage the counters associated with these events.

The registers responsible for the Link and Physical layers counters are accessed via the GLPCI_LCBADD and GLPCI_LCBDATA register pair.

Two events can be counted concurrently. The event counters include two sets of registers, each managing one event counter. Such pairs are documents as <register_name>[1:0].

The following procedures manage the operation of the event counters (when writing to part of the register, make sure other fields are written with their existing values):

- Resetting the counters configuration
  - — Set the XPPERFCON.GRST bit.
  - — Clear the XPPERFCON.GRST bit (otherwise the logic stays in reset).
- Setting an event:
  - — Write 0x0...0 to the XPPMCL[1:0] registers.
  - — Set the XPPMR[1:0].CENS field to 0x1.
  - — Set the XPPMR[1:0].CNTMD field to 0x1.
  - — Set the XPPMER[1:0].XPRSCA field to 0x1.
  - — Set the event according to Table 38-18.
- Starting a count:
  - — Set the XPPERFCON.GCE bit.
- Stopping a count:
  - — Clear the XPPERFCON.GCE bit.
- Reading the count; reading the counter clears their values:
  - — Read the respective XPPMDH[1:0] and XPPMDL[1:0] register pair in a single 64-bit aligned access.

Table 38-18 lists the Link and Physical Layer events.

Table 38-18. Link And Physical Layer Performance Events

Event	Description	Register Field
Uncorrectable Errors	Counts the total number of uncorrectable errors.	XPPMER[1:0].CNTUCERR
Correctable Errors	Counts the total number of correctable errors.	XPPMER[1:0].CNTCERR
Tx L0s State Utilization	Counts the number of entries to L0s on the Tx lanes.	XPPMER[1:0].TXL0SU
Rx L0s State Utilization	Counts the number of entries to L0s on the Rx lanes.	XPPMER[1:0].RXL0SU
Link Utilization	Counts clocks that a port is receiving data. If one counter counts receiver errors and another counter counts Link Utilization, a bit error rate can be calculated.	XPPMER[1:0].LNKUTIL
Recovery State Utilization	Counts the number of entries to recovery state.	XPPMER[1:0].RECOVERY
ASPM L1 State Utilization	Counts the number of entries to ASPM L1 state (like initiated by the device).	XPPMER[1:0].L1
SW L1 State Utilization	Counts the number of entries to L1 state initiated by software.	XPPMER[1:0].SWL1
Tx and Rx L0s Utilization	Counts number of events where both Tx and Rx are in L0s state.	XPPMER[1:0].RXL0STXL0SU
NAK DLLP Received	Counts number of received NAK DLLPs.	XPPMER[1:0].NAKDLLP

38.10.4.2.1 Bandwidth Counters

The bandwidth counters measure total payload bytes transferred over the PCIe link. Counting is provided per each traffic type (posted, non-posted, completions) per direction (upstream, downstream).

The mechanisms described above hold for the bandwidth counters with the following differences:

- Setting an event:
  - Set the XPPMR[1:0].CENS field to 0x1.
  - Set the XPPMR[1:0].EGS field to 0x2.
  - Set the XPPMR[1:0].FCCSEL field to the desired traffic type (posted, non-posted, completions, or all).
  - Set the XPPPMER[1:0].TXRXSEL field to desired values.
  - Set the XPPPMER[1:0].XPRSCA field to 0x1.
  - Set the XPPERFCON.GCE field to 0x1.

Registers fields used exclusively by the bandwidth counters:

- XPPMR[1:0].FCCSEL - Selects the desired traffic type (posted, non-posted, completions, or all)
- XPPPMER[1:0].TXRXSEL - Selects between monitoring downstream traffic, upstream traffic, or both

38.10.4.2.2 Registers Map

The register fields that control the Link and Physical Layer events are as follows.

Table 38-19. XP PM Compare Low Bits Register (XPPMCL[1:0]) (0x3288, 0x328c)

Field	Bit(s)	Init.	Description
CMPL	31:0	0xFF...F	PM compare low value. Low order bits [31:0] for PM compare register[1:0].

Table 38-20. XP PM Compare Low Bits Register (XPPMDL[1:0]) (0x32e8, 0x32f0)

Field	Bit(s)	Init.	Description
CNTL	31:0	0x00...00	PM data counter low value. Low order bits [31:0] for PM data counter[1:0].

Note: XPPMDL must be read together with the respective XPPMDH register as a single 64-bit aligned read. The registers are simultaneously cleared on read.

Table 38-21. XP PM Data High Bits Register (XPPMDH[1:0]) (0x32ec, 0x32f4)

Field	Bit(s)	Init.	Description
RSVD	31:4	0x0...0	Reserved.
CNTH	3:0	0x0	PM data counter high value. High order bits [35:32] for PM data counter[1:0].

Note: XPPMDH must be read together with the respective XPPMDL register as a single 64-bit aligned read. The registers are simultaneously cleared on read.

Table 38-22. XP PM Response Control Register (XPPMR[1:0]) (0x3294, 0x3298)

Field	Bit(s)	Init.	Description
EGS	20:19	0x0	Event group selection.
CNTMD	15:14	0x0	Count mode.
CENS	13:11	0x0	Counter enable source.

Table 38-23. XP PM Resource Events Register (XPPMER[1:0]) (0x32AC, 0x32B0)

Field	Bit(s)	Init.	Description
NAKDLLP	29	0b	NAK DLLP received - set to enable.
RXL0STXL0SU	28	0b	Tx and Rx L0s utilization - set to enable.
SWL1	27	0b	SW L1 state utilization - set to enable.
L1	26	0b	ASPM L1 state utilization - set to enable.
RECOVERY	25	0b	Recovery state utilization - set to enable.
CNTUCERR	24	0b	Count uncorrectable errors - set to enable.
CNTCERR	23	0b	Count correctable errors - set to enable.
TXL0SU	22	0b	Tx L0s state utilization event - set to enable.
RXL0SU	21	0b	Rx L0s state utilization event - set to enable.
XPRSCA	20:17	0x0	XP Resource Assignment 0001b = Set. Else = Reserved.
LNKUTIL	16:13	0x0	Link utilization - set to 0x1 to enable.
FCCSEL	4:2	0x0	Flow Control Class Select This field selects which flow class for resource event. xx1 = Posted. x1x = Non-posted. 1xx = Completion. ***Note:*** Setting to 111b counts posted, non-posted and completion traffic combined.
TXRXSEL	1:0	0x0	Tx/Rx Select (TXRXSEL) This field selects the traffic direction to monitor. 1x = Transmit. x1 = Receive (from PCIe bus). 11b = Either transmit or receive direction.

Table 38-24. Performance Monitor Local Control Register (XPPERFCON) (0x32C4)

Field	Bit(s)	Init.	Description
RSVD	31:2	0x0	Reserved.
GCE	1	0b	Global count enable.
GRST	0	0b	Global reset.

Table 38-25. XP PM Data Low Bits Register (XPPMDL[1:0]) (0x3280, 0x3284)

Field	Bit(s)	Init.	Description
CNTL	31:0	0x0	PM data counter low value (CNTL). Low order bits [31:0] for PM data counter[1:0].

38.10.4.3 Latency Counter

The latency counter measures the min, max, or average read latency.

Note: Completion timeout events are ignored when the latency counter is enabled.

The latency counters are managed via a set of register fields described below (see also Table , Table 38-27, and Table 38-28). Each of the following sources of traffic has its separate set of registers and counters:

- 0x0 - Rx LAN descriptor fetch
- 0x1 - Tx LAN descriptor fetch
- 0x2 - Protocol engine Rx requests
- 0x3 - Protocol engine Tx requests
- 0x4 - Internal cache load
- 0x5 - Internal management engine read
- 0x6 - Tx LAN packet fetch

The registers are accessed via the GLPCI_LCBADD and GLPCI_LCBDATA registers.

The register fields that control the latency counter operation are:

Table 38-26. NPQ Control Register - NPQC (0x00000)

Field	Bit(s)	Init.	Description
Reserved	31:10	0	Reserved
RTMNTREN	9	0	Latency Counting Enable This bit should set in order to enable latency counters. When set, completion timeout events are ignored.
PERFMNTREN	8	0	Performance Monitor Enable This bit should set in order to enable latency counters. Clearing this bit clears the latency counters.
PERFMNTRAVG	7:4	1	Performance Monitor Average Rate This field sets the averaging rate for all latency average monitors. See definition of NPQRTDLY1.ARTDLY. This field, divided by 16, is the weight (W) in an exponential moving average. The possible values are 1, 2, 4 or 8, which correspond to averaging rates of 0.0625, 0.125, 0.25 or 0.5, respectively.
Reserved	3:0	0x4	Reserved.

Table 38-27. NPQ Round-trip Delay 1 Register - NPQRTDLY1 (0x00030; RO)

Field	Bit(s)	Init.	Description
Reserved	31:16	0	Reserved
ARTDLY	15:0	0	Average Read Requests Round-trip Delay. Captures the average read latency experienced since the last counter reset. Latency is measured from time the read request starts until the time the completion starts to arrive. Average is calculated as exponential moving average. That is, the new average M(n) at sample n equals M(n) = (W/16) * new_sample + (16-W)/16*M(n-1), where W is defined in the NPQC.PERFMNTRAVG field.

Table 38-28. NPQ Round-trip Delay 2 Register - NPQRTDLY2 (0x00034; RO)

Field	Bit(s)	Init.	Description
MAXRTDLY	31:16	0	Maximum Read Requests Round-trip Delay. Captures the maximal read latency experienced since the last counter reset. Latency is measured from time the read request starts until the time the completion starts to arrive.
MINRTDLY	15:0	0	Minimal Read Requests Round-trip Delay. Captures the minimal read latency experienced since the last counter reset. Latency is measured from time the read request starts until the time the completion starts to arrive.

Latencies are measured in cycle counts, where a cycle duration is per Table 38-29.

Table 38-29. Resolution Of The Latency Counters

PCIe Operation Speed	Setting of the GLPCI_CLKCTL.PCI_CLK_DYN Bit	PCIe Operational Link Width	Cycle Duration (ns)
Gen 1 (2.5G)	0b	x	8
Gen 2 (5.0G)	0b	x	4
Gen 3 (8.0G)	0b	x	2
Gen 1 (2.5G)	1b	8 lanes	16
Gen 2 (5.0G)	1b	8 lanes	8
Gen 3 (8.0G)	1b	8 lanes	4
Gen 1 (2.5G)	1b	1 or 4 lanes	32
Gen 2 (5.0G)	1b	1 or 4 lanes	16
Gen 3 (8.0G)	1b	1 or 4 lanes	8

The NPQC register serves other proposes, so using it should be done in the following manner:

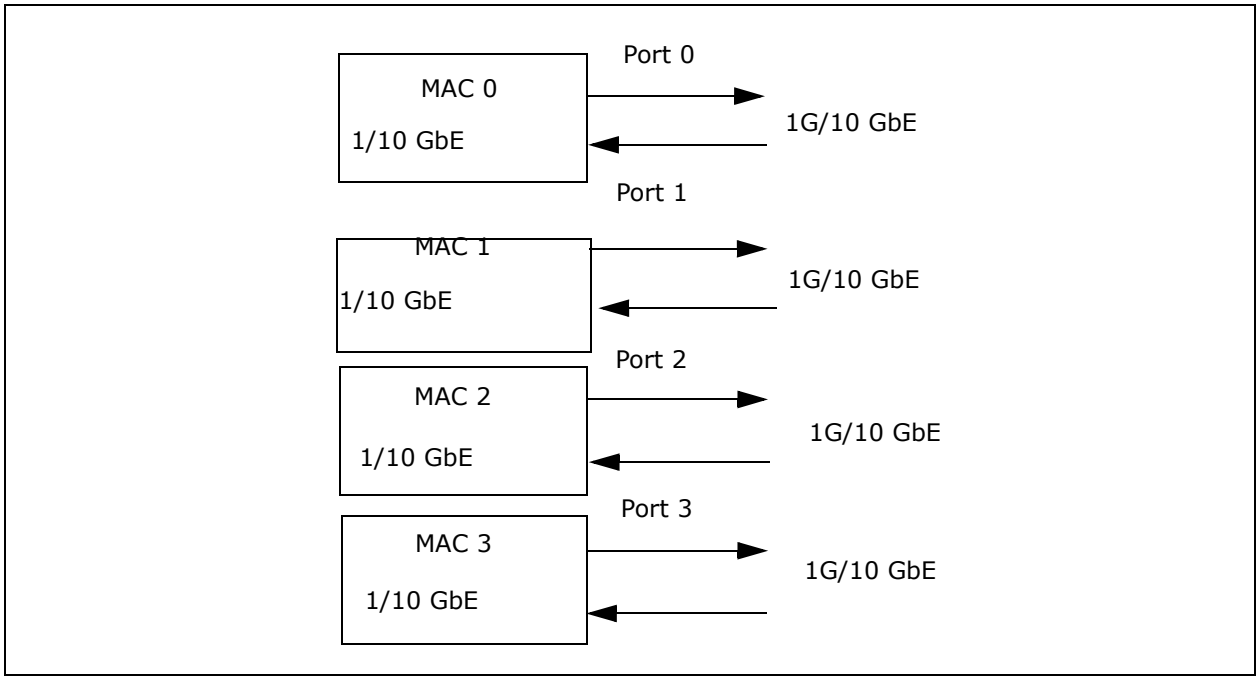
- Read the relevant NPQC register
- Modify the relevant fields from the list above in a local copy
- Write the local copy back to the device

Clearing the latency counters is done by setting the NPQC.PERFMNTREN bit to 0b.

38.10.5 Media Access Control (MAC) Layer

The 10 GbE controller supports up to four full duplex Ethernet MAC ports compliant with IEEE Std802.3-2012 standard (Clause 4 and Annex 4A). The MAC ports can be configured to operate at different speeds of operation as shown in Figure 38-6.

Figure 38-6. 10 GbE Controller MAC Configuration



Each MAC port is associated with a corresponding Media Access Unit (MAU) that provides the physical layer interfaces as explained in Section 38.10.5. The MAUs need to be configured to operate with appropriate physical layer protocols based on the MAC operating speed. Physical layers supported by the 10 GbE controller for different speeds of operation are explained in Section 38.10.7.

38.10.6 Requirements

Following lists the requirements for MAC layer.

38.10.6.1 Port Configurations And Speeds

- Port configuration ID defines the maximum number of enabled ports and the link speeds supported for each. the 10 GbE controller supports two configuration IDs:
  - — Two-port settings — Each port can be independently operated at 10 Gb/s, 1 Gb/s, or disabled
  - — Four-port settings — Each port can be independently run at 10 Gb/s, 1 Gb/s or disabled
- Configuration ID selection is defined in the NVM.
- The 10 GbE controller is able to switch between port configuration IDs (different number of ports) using only a core global reset (no need for POR).
- The 10 GbE controller supports dynamic speed change on each of its ports independently per PHY indication:

— Port 0/1 — 10 Gb/s and 1 Gb/s

— Port 2/3 — 10 Gb/s and 1 Gb/s

38.10.6.2 Port Mapping

- Each PCIe PHY can be flexibly mapped to any of the device ports. Note that a change of mapping requires a reset.
- Firmware reports PMC on link status or port status changes.

38.10.6.3 External PHY Support

- The 10 GbE controller's firmware supports an external PHY configuration pre-operating system mode (through the NVM) and responds to link events.
- The 10 GbE controller's firmware provides an Admin Queue interface so the LAN software device driver can communicate with an external PHY.

38.10.6.4 Pacing

- When a link is set to a speed of 10 Gb/s, support pacing of Tx packets such that the average data rate is lower than the link speed. Pacing is defined at 1Gb/s granularity and is implemented by extending the IPG.
- When a link is set to speed of 10 Gb/s or higher, support pacing of Tx packets such that the duration from one packet to the next does not exceed the equivalent of a given packet rate. Pacing is defined at 10 Mp/s granularity. It is implemented by extending each IPG to insure a certain duration from the first byte of a packet to the first byte of the next packet.
- The 10 GbE controller has a (per port configuration) programmable fixed IPG extension (on top of specification requirements) for all Tx packets to enable a connection through bump on the wire devices.

38.10.6.5 Loopback

The 10 GbE controller is able to run MAC loopback with no dependency on an external line clock or link indication.

38.10.6.6 Link Management

- The 10 GbE controller supports 5 x I^2C/MDIO interfaces in order to have a dedicated one-per-port in the case of four-port SFP+ and an additional one for thermal sensor or an external NVM.
- MDIO/I^2C interfaces can be flexibly assigned and shared between ports by firmware control.

38.10.6.7 MAC Speed Configuration

Table 38-30 lists the possible speed configurations available on each MAC port.

Table 38-30. Mac Port And Possible Speed Configurations

Port	1 Gb/s	10 Gb/s
MAC 0	Y	Y
MAC 1	Y	Y
MAC 2	Y	Y
MAC 3	Y	Y

When LAN ports are disabled in multi-port system configurations, corresponding PCIe functions need to be disabled through the NVM or external disable pins.

The 10 GbE controller supports auto-negotiation protocol when configured for certain physical interfaces. The link speed on each port can either be configured through Link Configuration Admin commands (see Section 38.11.3) or auto selected when auto-negotiation is enabled on the respective ports.

38.10.6.8 Ethernet CRC Generation and Stripping

The 10 GbE controller MAC supports CRC generation (Ethernet frame check sequence). The MACs can be independently programmed to generate and append the 32-bit CRC (FCS) in the transmit direction by setting the *CRC Enable* field in the Set MAC Config AQ command.

The default value is set for the MAC to append the CRC.

Note: When CRC generation is disabled in the MAC, automatic padding in the MAC does not work correctly, therefore all transmitted packets must be at least 64 bytes long.

38.10.6.9 Transmit Padding

The minimum frame size for Ethernet as specified by IEEE Std 802.3 standard is 64 bytes. The 10 GbE controller MAC pads, with ZEROs, Ethernet packets that are smaller than 64 bytes during transmit. Refer to the padding rules for received packets and loop-back packets section for more details.

When interfacing to an external crypto engine, padding is done prior to sending the packet to it meaning that the crypto engine receives packets that are at least 60 bytes long.

38.10.6.10 Jumbo Frame Support

The 10 GbE controller MAC supports transmission and reception of frames of up to 9.5 KB (9728 bytes). Maximum receive and transmit frame size is configured through the Set MAC Config <Max Frame Size > field.

38.10.6.11 Ethernet Flow Control (FC) through Standard Packets

Note: The 10 GbE controller supports receiving/transmitting an FC indication through a side interface (as defined by ONPI specification).

This interface is enabled only when it is not connected directly to an external link (RPT, CLR...).

The 10 GbE controller supports flow control (pause) as defined in 802.3x (IEEE Std 802.3-2008 Annex 31B), as well as the specific operation of asymmetrical flow control (asymmetric pause) defined by 802.3z (IEEE Std 802.3-2008 Annex 28B). The 10 GbE controller also supports Priority Flow Control (PFC) as defined in IEEE P802.1Qbb, sometimes referred to as Class Based Flow Control or (CBFC), as part of the DCB architecture.

Note: A 10 GbE controller port can either be configured to receive 802.3x Link Flow Control (LFC) packets or 802.1Qbb/802.3bd PFC packets. It does not support the reception of both types of packets simultaneously over the same port.

Flow control is implemented to reduce receive buffer overflows, which result in the dropping of received packets. Flow control also allows for local controlling of network congestion levels. This can be accomplished by sending an indication to a transmitting station of a nearly full receive buffer condition at a receiving station.

The implementation of asymmetric flow control allows for one link partner to send flow control packets while being allowed to ignore their reception. For example, not required to respond to PAUSE frames.

The following registers define the basic control functionality. Refer to the registers specified in Section 38.24.1.1.6, Section 38.24.1.2.10, and Section 38.24.2.2 for the other programming related to flow control. In DCB mode, some of the registers are duplicated per Traffic Class (TC), up to eight duplicate copies of the registers. If DCB is disabled, index [0] of each register is used.

- MAC Flow Control (PRTDCB_MFLCN) register — Enables flow control and passing of MAC control packets to the host.
- Flow Control Configuration (PRTDCB_FCCFG) — Determines mode for Tx flow control (no FC, LFC, or versus PFC).
- Flow Control Transmit Timer Value (PRTDCB_FCTTVN[3:0]) — a set of 16-bit timer values to include in transmitted PAUSE frame. A single timer is used in LFC mode and up to eight timers are used in PFC mode.
- Flow Control Refresh Threshold Value (PRTDCB_FCRTV) — 16-bit PAUSE refresh threshold value (in legacy FC PRTDCB_FCRTV must be smaller than PRTDCB_FCTTVN[0]).
- PRTDCB_TC2PFC.TC2PFC bitmap shall be set to 0xFF in LFC mode.

38.10.6.11.1 Mac Control Frames And Reception Of Flow Control Frames

Mac Control Frame — Other Than FC

IEEE 802 reserved the Ether-type value of 0x8808 for MAC control frames as listed in Table 38-31. The MAC control frame format is specified in IEEE 802.3 Clause 31.

Table 38-31. Mac Control Frame Format

DA	The *Destination Address* field can be an individual or multicast (including broadcast) address. Permitted values for the *Destination Address* field can be specified separately for a specific control opcode such as FC packets.
SA	Port Ethernet MAC address (6 bytes).
Type	0x8808 (2 bytes).
Opcode	The MAC control opcode indicates the MAC control function.
Parameters	The *MAC Control Parameters* field must contain MAC control opcode-specific parameters. This field can contain none, one, or more parameters up to a maximum of minFrameSize =20 bytes.
Reserved Field = 0x00	The *Reserved* field is used when the MAC control parameters do not fill the fixed length MAC control frame.
CRC	4 bytes.

MAC Control Frame Receive Identification

Packets with:

- MAC DA = 01-80-C2-00-00-01.
- Ether-type value of 0x8808 are considered to be control frames.
- UC MAC DA (PRTGL_SAL and PRTGL_SAH).

Structure of 802.3x FC Packets

802.3x FC packets are defined by the following three fields (see Table 38-32):

1. A match on the six-byte multicast address for MAC control frames or a match to the station address of the device. The 802.3x standard defines the MAC control frame multicast address as 01-80-C2-00-00-01. A match on the *Type* field. The *Type* field in the FC packet is compared against an IEEE reserved value of 0x8808.
2. A match of the *MAC Control Opcode* field has a value of 0x0001.

Frame-based flow control differentiates XOFF from XON based on the value of the PAUSE *Timer* field. Non-zero values constitute XOFF frames while a value of zero constitutes an XON frame. Values in the *Timer* field are in units of pause quanta (slot time). A pause quanta lasts 64 byte times, which is converted in to an absolute time duration according to the line speed.

Note: XON frame signals the cancellation of the pause that was initiated by an XOFF frame. For example, a pause for zero pause quanta.

Table 38-32. 802.3x Link Flow Control Frame Formats

DA	01_80_C2_00_00_01 (6 bytes).
SA	Port Ethernet MAC address (6 bytes).
Type	0x8808 (2 bytes).
Opcode	0x0001 (2 bytes).
Time	XXXX (2 bytes).
Pad	42 bytes.
CRC	4 bytes.

802.3x Frame Receive Identification

Received frames that are identified as control frames can further be classified as 802.3x if the frames opcode = 0x0001 as listed in Table 38-32. See MAC Control Frame Receive Identification for more details.

The MAC supports the following configuration to further manage 802.3x FC frames' Rx identification:

- UC MAC DA (PRTGL_SAL and PRTGL_SAH).

Priority Flow Control (PFC)

DCB introduces support for multiple TCs assigning different priorities and bandwidth per TC. LFC stops all the TCs. PFC specified in IEEE P802.1Qbb enables more granular flow control on the Ethernet link in an DCB environment as opposed to the PAUSE mechanism defined in 802.3x. The PFC frame format is specified in IEEE P802.3bd.

Table 38-33. PFC Packet Format

DA	01_80_C2_00_00_01 (6 bytes).
SA	Port Ethernet MAC address (6 bytes).
Type	0x8808 (2 bytes).
Opcode	0x0101 (2 bytes).
Priority Enable Vector	0x00XX (2 bytes).
Timer 0	XXXX (2 bytes).
Timer 1	XXXX (2 bytes).
Timer 2	XXXX (2 bytes).
Timer 3	XXXX (2 bytes)
Timer 4	XXXX (2 bytes).
Timer 5	XXXX (2 bytes).
Timer 6	XXXX (2 bytes).
Timer 7	XXXX (2 bytes).
Pad	26 bytes.
CRC	4 bytes.

Table 38-34. Format Of Priority Enable Vector

	ms octet	ls octet
Priority enable vector definition	0	e[7]...e[n]...e[0]
e[n] =1 => time (n) valid e[n] =0 => time (n) invalid		

Each of the eight timers refers to a specific User Priority (UP). For example, Timer 0 refers to UP 0, etc. The 10 GbE controller binds a UP (and therefore the timer) to one of its TCs according to the UP-to-TC binding tables. Refer to the PRTDCB_TUP2TC register for the binding of received PFC frames to Tx TCs, and to the PRTDCB_RUP2TC register for the binding of transmitted PFC frames to Rx TCs.

When a PFC frame is formatted by the 10 GbE controller, the same values are replicated into every *Timer* field and priority enable vector bit of all the UPs bound to the concerned TC. These values as configured in the PRTDCB_RUP2TC register.

The following rule is applicable for the case of multiple UPs that share the same TC as configured in the PRTDCB_TUP2TC register. When PFC frames are received with different timer values for the previous UPs, the traffic on the associated TC must be paused by the highest XOFF timer's value.

PFC Frame Receive Identification

Received frames that are identified as control frames can further be classified as PFC if the frames opcode = 0x0101 as listed in Table 38-33. See MAC Control Frame Receive Identification for more details.

The MAC supports the following configurations to manage PFC frames' identification on Rx:

- UC MAC DA (PRTGL_SAL and PRTGL_SAH).

Operation and Rules

The 10 GbE controller operates in either LFC mode or in PFC mode. Enabling both modes concurrently is not allowed.

The MAC implements the following configurations:

- LFC (PAUSE) is enabled by the *RFCE* bit in the PRTDCB_MFLCN register.
- PFC is enabled by the *RPFCE* bit in the PRTDCB_MFLCN register.

Once the receiver has validated the reception of an XOFF, or PAUSE frame, the device performs the following:

- Increments the appropriate statistics register(s).
- Initialize the pause timer based on the packet's PAUSE *Timer* field (overwriting any current timer's value).
  - In case of PFC, this is done per TC. If several UPs are associated with a TC, then the device sets the timer to the maximum value among all enabled timer fields associated with the TC.
- Disable packet transmission or schedule the disabling of transmission after the current packet completes.
  - In case of PFC, this is done per paused TC.

Resumption of transmission can occur under the following conditions:

- Expiration of the PAUSE timer.
  - In case of PFC, this is done per TC.
- Reception of an XON frame (a frame with its PAUSE timer set to 0b).
  - In case of PFC, this is done per TC.

Both conditions clear the relevant TXOFF status bits in the Transmit Flow Control Status (PRTDCB_TFCS) register and transmission can resume. Hardware records the number of received XON frames.

Timing Considerations

When operating at 10 Gb/s line speed, the 10 GbE controller must not begin to transmit a (new) frame more than 60 pause quanta after receiving a valid XOFF frame, as measured at the wires. When connected to an external 10GBASE-KR PHY with FEC or to an external 10GBASE-T PHY, the response time requirement increases to 74 pause quanta, because of extra delays consumed by these external PHYs.

When operating at 1 Gb/s line speed, the 10 GbE controller must not begin to transmit a (new) frame more than 2 pause quanta after receiving a valid XOFF frame, as measured at the wires.

The IEEE P802.1Qbb draft 2.3, specifies that the tolerated response time for priority XOFF frames is 614.4 ns (equivalent of 12 pause quanta at the link speed of 10 Gb/s). This extra budget in addition to the link delay is aimed to compensate the fact that decision to stop new transmissions from a specific TC must be taken earlier in the transmit data path than for the LFC case.

38.10.6.11.2 PAUSE and MAC Control Frames Forwarding

At 10 Gb/s and Lower Speeds

Two bits in the PRTDCB_MFLCN register control the transfer of PAUSE and MAC control frames to the host. These bits are Discard PAUSE Frames (DPF) and Pass MAC Control Frames (PMCF). Note also that any packet must pass the L2 filters as well.

- The *DPF* bit controls transfer of PAUSE packets to the host. The same policy applies to both LFC and PFC packets as listed in Table 38-35. Note that any packet must pass the L2 filters as well.
- The *PMCF* bit controls transfer of non-PAUSE packets to the host. Note that when LFC frames are not enabled (RFCE = 0b) then LFC frames are considered as MAC Control (MC) frames for this matter. Similarly, when PFC frames are not enabled (RPFCE = 0b) then PFC frames are considered as MC frames as well.

Note: When virtualization is enabled, forwarded control packets are queued according to the regular switching procedure.

Table 38-35. Transfer of PAUSE Packet to Host (DPF bit) in 10 Gb/s MAC

RFCE	RPFCM	DPF	LFC Handling	PFC Handling
0b	0b	X	Treat as MC (according to PMCF setting).	Treat as MC (according to PMCF setting).
1b	0b	0b	Accept.	Treat as MC (according to PMCF setting).
1b	0b	1b	Reject.	Treat as MC (according to PMCF setting).
0b	1b	0b	Treat as MC (according to PMCF setting).	Accept.
0b	1b	1b	Treat as MC (according to PMCF setting).	Reject.
1b	1b	X	Unsupported setting.	Unsupported setting.

38.10.6.11.3 Transmitting PAUSE Frames

The 10 GbE controller generates PAUSE packets to insure there is enough space in its receive packet buffers to avoid packet drop. The 10 GbE controller monitors the fullness of its receive FIFOs and compares it with the contents of a programmable threshold. When the threshold is reached, the 10 GbE controller sends a PAUSE frame. The 10 GbE controller supports both LFC and PFC but not both concurrently (at the same physical port). When DCB is enabled, it sends only PFC, and when DCB is disabled, it sends only LFC.

Note: Similar to the reception of flow control packets previously mentioned, software can enable flow control transmission by setting the PRTDCB_FCCFG.TFCE field only after it is negotiated between the link partners (possibly by auto-negotiation).

PFC

Like Rx flow control, Tx flow control operates in either a link 802.3x compliant mode or in PFC mode, but not in both at the same time.

The same flow control mechanism is used for PFC and for 802.3x flow control to determine when to send XOFF and XON packets. When PFC is used in the receive path, Priority PAUSE packets are sent instead of 802.3x PAUSE packets. See 802.3x Frame Receive Identification for the format of priority PAUSE packets.

A specific consideration for generating PFC packets:

- When a PFC packet is sent, the packet sets all the UPs that are associated with the relevant TC (UP-to-TC association in receive is defined in PRTDCB_RUP2TC register).

Operation And Rules

At 10 Gb/s and lower speeds, The *TFCE* field in the Flow Control Configuration (PRTDCB_FCCFG) register enables transmission of PAUSE packets as well as selects between the LFC mode and the PFC mode.

Refer to Section 38.24.1.2.4 for the criteria used by the device for sending a XOFF frame to the neighbor. The 10 GbE controller sends an additional PAUSE frame if it has previously sent one and the FIFO overflows. This is intended to minimize the amount of packets dropped if the first PAUSE frame did not reach its target.

From the time it has issued a PAUSE frame to the neighbor the 10 GbE controller starts counting down in an internal shadow counter that is used to mirror the pause time-out counter at the partner's end. When this internal counter reaches the value set in PRTDCB_FCRTV register then, if the PAUSE condition is still valid (meaning that the buffer(s) fullness is still above the relevant watermarks), an XOFF message is sent again.

Once the receive buffer fullness reaches the relevant low watermarks, the 10 GbE controller sends an XON message (a PAUSE frame with a timer value of zero). Refer to Section 38.24.1.2.7 for the criteria used by the device for sending a XON frame to the neighbor.

38.10.6.12 Inter Packet Gap (IPG) Control And Pacing

The 10 GbE controller supports transmission pacing by extending the IPG (the gap between consecutive packets). The pacing mode enables the average data rate to be slowed in systems that cannot support the full link rate (10 Gb/s or 1 Gb/s). As listed in Table 38-36, the pacing modes work by stretching the IPG in proportion to the data sent. In this case, the data sent is measured from the end of preamble to the last byte of the packet. No allowance is made for the preamble or default IPG when using pacing mode.

Example 1:

Consider an example of a 64-byte frame. To achieve a 1 Gb/s data rate when link rate is 10 Gb/s and packet length is 64 bytes (16 Dwords), programmers need to add an additional IPG of 144 Dwords (nine times the packet size to reach 1 Gb/s). Which when added to the default IPG gives an IPG of 147 Dwords.

Example 2:

Consider an example of a 65-byte frame. To achieve a 1 Gb/s data rate when link rate is 10 Gb/s and packet length is 65 bytes (17 Dwords when rounded up) programmers need to add an additional IPG of 153 Dwords (nine times the packet duration in Dwords). Which when added to the default IPG gives an IPG of 156 Dwords. Note that in these case, where the packet length counted in Dwords is not an integer, programmers need to count any fraction of a Dword as a whole Dword for computing the additional IPG.

Table 38-36 lists the pacing configurations supported by the 10 GbE controller at link rates of 10 Gb/s. When operating at lower link speeds the pacing speed is proportional to the link speed. Pacing is configured in the *Pacing Config* field in the Set MAC Config admin command.

Table 38-36. Pacing Speeds At 10 Gb/s Link Speed

Pacing Speeds (Gb/s)	Delay Inserted Into IPG	*Pacing Config* field in the Set MAC Config Admin Command
10 (LAN)	None	0000b
9.0	1 Dword for 9 transmitted	1001b
8.0	1 Dword for 4 transmitted	1000b
7.0	3 Dwords for 7 transmitted	0111b
6.0	2 Dwords for 3 transmitted	0110b
5.0	1 Dwords for 1 transmitted	0101b
4.0	3 Dwords for 2 transmitted	0100b
3.0	7 Dwords for 3 transmitted	0011b
2.0	4 Dwords for 1 transmitted	0010b
1.0	9 Dwords for 1 transmitted	0001b
10	None	Default

38.10.6.13 MAC Speed Change At Different Power Modes

Auto-negotiation enables establishment of link speed at the Highest Common Denominator (HCD). During certain low power modes power saving might be more important than link performance. The 10 GbE controller supports an additional mode of operation, where it can configure the PHY to a fixed speed, starting from the lowest speed and checking if the link can be up. This method sets the link speed to the Lowest Common Denominator (LCD) link speed. The link-up process enables the link to come up at any possible speed in cases where power is more important than performance.

38.10.6.14 MAC Errors

The 10 GbE controller's MAC supports identification of the following erroneous packets:

- **L2 CRC Error**
  Packet's FCS check resulted in an error.
- **Undersized Or Oversized Packets**
  Received frame size is smaller than 64 bytes or larger than the configured max. frame size, which was either loaded from the NVM or set using the Set MAC Config command.
- **Illegal Byte Error**
  An illegal control byte is any value that is not legal. Refer to the IEEE802.3 specification for legal symbols, such as /I/,/ E/,/T/, or /D/, /S/, /Seq/, or /LPI/).
- **Error Byte Error**
  Indication that a packet was received with the error control byte (/E/) on the XGMII interface. Indicates that the PCS has encountered an error during the packet's reception.
- **802.3 Length Error**
  <Length> field information in the 802.3 header does not match the packet's actual size.

Packet's containing any one of the previous errors can be filtered by the device or forwarded to a pre-configured VSI based on the SBP flag in the PRT_SBPVSI register.

Note: Packet's shorter than 64 bytes are always filtered by the MAC layer and are not affected by the Store Bad Packets (SBP) configuration.

38.10.6.14.1 MAC Error Counters

Table 38-37 lists the different MAC error counters supported by the device.

Table 38-37. MAC Error Counters

Counter Name	Description
CRC Error	Counts the number of packets received with CRC errors that are not fragments.
Receive Length Error	Counts the number of packets received with error in length field comparison in the 802.3 header.
Receive Undersize	Counts the number of packets with good CRC that are smaller than 64 bytes.
Receive Fragment	Counts the number of packets with bad CRC that are smaller than 64 bytes
Receive Oversize	Counts the number of packet that are larger than the configured max frame size.
MAC Short Packet Discard	Counts the number of packets smaller than 32 bytes that were discarded by the MAC layer.

38.10.7 Link Management Interfaces

The 10 GbE controller supports either MDIO or I^2C interfaces for control plane connection between the MAC (master side) and external PHY devices. The MDIO or I^2C interfaces enable both MAC and firmware access to the PHY for monitoring and controlling of the PHYs functionality. The 10 GbE controller MDIO is compliant with the IEEE Std 802.3 Clause 45 as well as IEEE Std 802.3 Clause 22 frame formats and register address space for accessing legacy PHY devices. The 10 GbE controller I^2C is compliant with the I^2C bus specification.

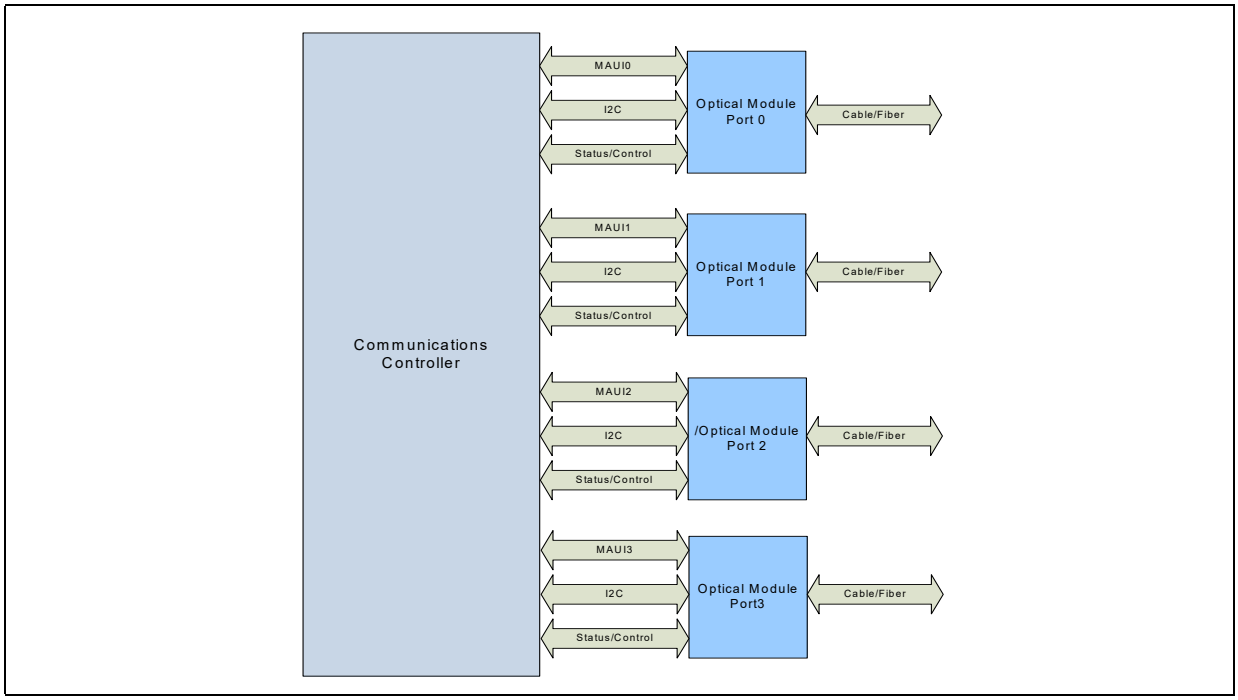
The 10 GbE controller supports up to four management interfaces (one per port) to control external PHYs devices. Depending on the PHY type to manage, the management interface can be configured for either MDIO interface or 2-wire management interface (I^2C). The configuration is done through control bits in the NVM.

Note: Section 38.10.7.2 includes the details on I^2C interface.

In order to manage multi-port PHYs, an I^2C/MDIO interface can be configured to control a quad port PHY or two I^2C/MDIO interfaces can be configured to control two dual port PHYs. The PHY configuration registers for each port are mapped into the respective I^2C/MDIO address space. The 10 GbE controller provides hardware acceleration of MDIO accesses over the 2-wire management interface. The software device driver manages the external devices using the Admin commands (see Section 38.11.3). The software device driver does not have direct access to the MDIO bus except for diagnostic purposes. Firmware performs direct access to MDIO interface for reading and writing to the PHY device as described in the paragraphs that follow.

Figure 38-7 shows the basic connectivity between the MAC and an external PHY/ module.

Figure 38-7. Basic PHY/Module Connectivity



38.10.7.1 MDIO Interface

The MDIO interface is a simple two-wire serial interface between MAC and PHY and is used to access Control and Status registers inside the PHY. The interface is implemented using two 3.3V I/Os:

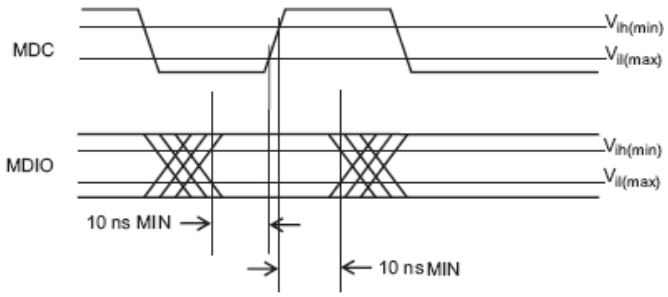
1. MDC — MDIO-interface clock signal driven by a MAC (STA) device.
2. MDIO — Read/write data between MAC and PHY.

38.10.7.1.1 MDIO Timing Relationship To MDC

The MDC clock toggles during a read/write operation at a fixed clock frequency of 2.441 MHz.

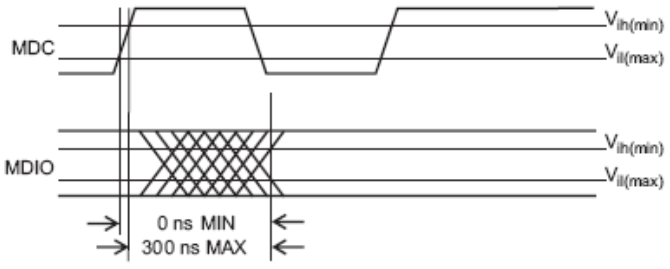
MDIO is a bidirectional signal that can be sourced by the Station Management Entity (STA) or the PHY. When the STA sources the MDIO signal, the STA must provide a minimum of 10 ns of setup time and a minimum of 10 ns of hold time referenced to the rising edge of MDC, as shown in Figure 38-8 (measured at the MII connector).

Figure 38-8. MDIO Timing Sourced By the MAC



When the MDIO signal is sourced by the PHY, it is sampled by the MAC (STA) synchronously with respect to the rising edge of MDC. The clock to output delay from the PHY, as measured at the MII connector, must be a minimum of 0 ns, and a maximum of 300 ns, as shown in Figure 38-9.

Figure 38-9. MDIO Timing Sourced by the PHY



38.10.7.1.2 IEEE Std 802.3 Clause 22 and Clause 45 Differences

IEEE Std 802.3 Clause 45 provides the ability to access additional device registers while still retaining logical compatibility with interface defined in Clause 22. Clause 22 specifies the MDIO frame format and uses an ST code of 01 to access registers. In Clause 45, additional registers are added to the address space by defining MDIO frames that use a ST code of 00.

Clause 45 (MDIO interface) Major Concepts:

1. Preserve management frame structure defined in IEEE 802.3 Clause 22.
2. Define mechanism to address more registers than specified in IEEE 802.3 Clause 22.
3. Define ST and OP codes to identify and control the extended access functions.

38.10.7.1.3 MDIO Management Frame Structure

The MDIO interface frame structure defined in IEEE802.3 Clause 22 and Clause 45 are compatible so that the two systems supporting different formats can co-exist on the same MDIO bus. The 10 GbE controller supports both frame structures to enable interfacing PHYs that support either protocol.

The basic frame format as defined in IEEE802.3 Clause 22 can optionally be used for accessing legacy PHY registers is listed in Table 38-38.

Table 38-38. Clause 22 Basic MDIO Frame Format

	Management Frame Fields							
Frame	**Pre**	**ST**	**OP**	**PRTAD**	**REGAD**	**TA**	**Data**	**Idle**
Read	1...1	01	10	PPPPP	RRRRR	Z0	DDDDDDDDDDDDDDDD	Z
Write	1...1	01	01	PPPPP	RRRRR	10	DDDDDDDDDDDDDDDD	Z

The MDIO interface defined in Clause 45 uses indirect addressing to create an extended address space enabling access to a large number of registers within each MDIO Managed Device (MMD). The MDIO management frame format is listed in Table 38-39.

Table 38-39. Clause 45 Indirect Addressing MDIO Frame Format

	Management Frame Fields							
Frame	**Pre**	**ST**	**OP**	**PRTAD**	**DEVAD**	**TA**	**Address / Data**	**Idle**
Address	1...1	00	00	PPPPP	EEEEE	10	AAAAAAAAAAAAAAAA	Z
Write	1...1	00	01	PPPPP	EEEEE	10	DDDDDDDDDDDDDDDD	Z
Read	1...1	00	11	PPPPP	EEEEE	Z0	DDDDDDDDDDDDDDDD	Z
Post-Read Increment Address	1...1	00	10	PPPPP	EEEEE	Z0	DDDDDDDDDDDDDDDD	Z

To support clause 45 indirect addressing each MMD (PHY — MDIO managed device) implements a 16-bit address register that stores the address of the register to be accessed by data transaction frames. The address register must be overwritten by address frames. At power up or device reset, the contents of the address register are undefined. Write, read, and post-read-increment-address frames must access the register whose address is stored in the address register. Write and read frames must not modify the contents of the address register. Upon receiving a post-read-increment-address frame and having completed the read operation, the MMD increments the Address register by one (up to a value of 0xFFFF). Each MMD supported implements a separate address register, so that the MMD's address registers operate independently of one another.

Idle Condition (IDLE) — The IDLE condition on MDIO is a high-impedance state. All three state drivers must be disabled and the PHY's pull-up resistor pulls the MDIO line to a logic one.

Preamble (PRE) — At the beginning of each transaction, the station management entity must send a sequence of 32 contiguous consecutive one bits on MDIO with 32 corresponding cycles on MDC to provide the PHY with a pattern that it can use to establish synchronization. A PHY must observe a sequence of 32 contiguous consecutive one bits on MDIO with 32 corresponding cycles on MDC before it responds to any transaction.

Start of Frame (ST) — The ST is indicated by:

- <00> pattern for clause 45 compatible frames for indirect access cycles.
- <01> pattern for clause 22 compatible frames for direct access cycles.

These patterns ensure a transition from the default value of one on the MDIO signal, and identifies the start of frame.

Operation **Code (OP)** — The *OP* field indicates the type of transaction being performed by the frame.

For Clause 45 compatible frames:

- A <00> pattern indicates that the frame payload contains the address of the register to access.
- A <01> pattern indicates that the frame payload contains data to be written to the register whose address was provided in the previous address frame.
- A <11> pattern indicates that the frame is an indirect read operation.
- A <10> pattern indicates that the frame is an indirect post-read-increment-address operation.

For Clause 22 compatible frames:

- A <10> pattern indicates a direct read transaction from a register.
- A <01> pattern indicates a direct write transaction to a register.

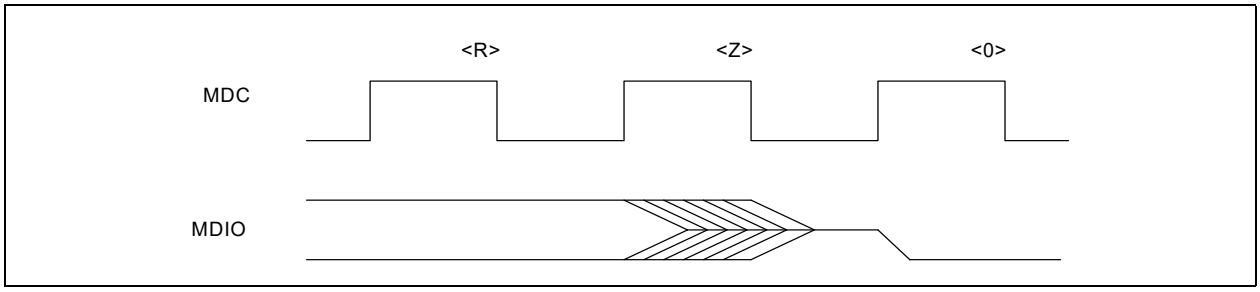
Port Address (PRTAD) — The PRTAD is five bits, allowing 32 unique PHY port addresses. The first PRTAD bit to be transmitted and received is the MSB of the address. A station management entity must have prior knowledge of the appropriate port address for each port to which it is attached, whether connected to a single port or to multiple ports.

Device Address (DEVAD) — The DEVAD is five bits, allowing 32 unique MMDs per port. The first DEVAD bit transmitted and received is the MSB of the address. This field is relevant only in clause 45 compatible frames (ST=<00>).

Register Address (REGAD) — The REGAD is five bits, allowing 32 individual registers to be addressed within each PHY. The first REGAD bit transmitted and received is the MSB of the address. This field is relevant only in clause 22 compatible frames (ST=<01>).

Turnaround (TA) — The TA time is a 2-bit time spacing between the DEVAD field and the Data field of a management frame. This is to avoid contention during a read transaction. For a read or post-read-increment-address transaction, both the STA and the PHY must remain in a high-impedance state for the first bit time of the TA. The PHY must drive a zero bit during the second bit time of the TA of a read or post read-increment-address transaction. During a write or address transaction, the STA must drive a one bit for the first bit time of the TA and a zero bit for the second bit time of the TA. Figure 38-10 shows the behavior of the MDIO signal during the *TA* field of a read transaction.

Figure 38-10.Behavior of MDIO During TA Field of a Read Transaction



- Clause 45 compatible frames have 16-bit address/data fields. For an auto-negation address cycle, it contains the address of the register to be accessed on the next cycle. For the data cycle of a write frame, the field contains the data to be written to the register. For a read or post-read-increment-address frame, the field contains the contents of the register. The first bit transmitted and received must be bit 15.
- Clause 22 compatible frames have 16-bit data fields. The first data bit transmitted and received must be bit 15 of the register being addressed.

38.10.7.1.4 MDIO Direct Access

The software device driver manages the PHYs using the Admin commands (See Section 38.11.3). The device driver does not have direct access to the MDIO bus except for diagnostic purposes. Firmware performs direct access to the MDIO interface for reading and writing to the PHYs as described in the following paragraphs.

The MDI is accessed through registers GLGEN MSCA and GLGEN MSRWD. A single management frame is sent by setting bit GLGEN MSCA.MDICMD to 1b after programming the appropriate fields in the MSCA and MSRWD registers. The GLGEN MSCA.MDICMD bit is auto cleared after the read or write transaction completes. To execute Clause 22 format write operations, the following steps should be done:

1. Data to be written is programmed in field GLGEN MSRWD.MDIWRDATA.
2. Register GLGEN MSCA is initialized with the appropriate control information (start, code, etc.) with bit GLGEN MSCA.MDICMD set to 1b.
3. Wait for bit GLGEN MSCA.MDICMD to reset to 0b when indicating that the transaction on the MDIO interface is complete.

The steps for Clause 22 format read operations are identical to the write operation except that the data in field GLGEN MSRWD.MDIWRDATA is ignored and the data read from the external device is stored in register field GLGEN MSRWD.MDIRDDATA bits. Clause 45 format read/write operations must be performed in two steps. The address portion of the pair of frames is sent by setting register field GLGEN MSCA.MDIADD to the desired address, field GLGEN MSCA.STCODE to 00b (start code that identifies Clause 45 format), and register field GLGEN MSCA.OPCODE to 00b (Clause 45 address register write operation). A second data frame must be sent after the address frame completes. This second frame executes the write or read operation to the address specified in the PHY address register.

38.10.7.2 Two-wire I^{2}C Management Interface

The I^{2}C interface operates via the GLGEN_I2CCMD and GLGEN_I2CPARAMS register set. Since this register set can be used by either software or firmware in alternation, its ownership must be acquired/released via the semaphore ownership taking/release flows.

The I^{2}C interface can be used in two methods, a hardware based access, where the device initiates a transaction following a software device driver or firmware request via the GLGEN_I2CCMD register or a software controlled bit banging using the GLGEN_I2CPARAMS register.

38.10.7.2.1 Hardware-Based I^{2}C Access

The following flows should be used to access an I^{2}C register.

As part of device initialization, or anytime before the actual access, the following parameters should be set:

- *GLGEN_I2CPARAMS.PHYADD* — the address of the device to access.
- *GLGEN_I2CPARAMS.ACCESS_WIDTH* — the width of the data to read or write (byte or word).

Note: The *GLGEN_I2CPARAMS* register should not be modified during an I^{2}c transaction.

To execute a write access, the following steps should be done:

1. Check that register is ready: Poll GLGEN_*I2CCMD.R* bit until it is read as 1b.
2. Command — the *GLGEN_I2CCMD* register is initialized with the appropriate PHY register address in *REGADD* field, the data to write in the *DATA* field and the operation (write) to the *OP* field (0b).
   a. If an interrupt is required, set the GLGEN_*I2CCMD.I* field.
3. Check that command completes: Poll the *GLGEN_I2CCMD.R* bit until it is read as 1b.
   a. Check that no error is indicated in the *GLGEN_I2CCMD.E* field.

To execute a read access, the following steps should be done:

1. Check that register is ready: Poll *GLGEN_I2CCMD.R* bit until it is read as 1b.
2. Command — The *GLGEN_I2CCMD* register is initialized with the appropriate PHY register address in *REGADD* field, and the operation (read) to the *OP* field (1b).
   a. If an interrupt is required, set the *GLGEN_I2CCMD.I* field.
3. Check that command completes: Poll *GLGEN_I2CCMD.R* bit until it is read as 1b.
   a. Check that no error is indicated in the *GLGEN_I2CCMD.E* field.

4. Read the data returned from the *GLGEN_I2CCMD.DATA* field. If a byte access is done (*GLGEN_I2CPARAMS.ACCESS_WIDTH* = 0), only *DATA[7:0]* is valid.

See Section 38.10.7.2.2 for the I^2C commands supported when using the built in read and write commands. All the transactions uses a clock of 100 kHz. When using the bit bang method any command can be given to the I^2C device.

38.10.7.2.2 Bit Bang Based I^2C Access

In this mode, the software device driver or the firmware controls the I^2C interface directly using the *GLGEN_I2CPARAMS* register according to the following table.

Pad	Field Controlling the Output Value	Field Reflecting the Input Value	Field Controlling the Output Enable Value[1]
SDPx_2 I^2C clock)	CLK_OUT	CLK_IN	CLK_OE_N
SDPx_3 (I^2C data)	DATA_OUT	DATA_IN	DATA_OE_N

Notes:
1. 0b = Pad is output. 1b = Pad is input.

38.10.7.2.3 Supported Commands

Note: The gray columns in the tables that follow denote cycles driven by the I^2C device. White columns denote cycles driven by the 10 GbE controller.

When a word read command (*GLGEN_I2CPARAMS.ACCESS_WIDTH* = 1b, *GLGEN_I2CCMD.OP* =1b) is given the following sequence is done by the 10 GbE controller:

Table 38-40. I^2C Read Transaction - Dummy Write

1	7	1	1	8	1	
S	Device Address	Wr	A	Register Address	A	
	From GLGEN_*I2CCMD.PHYADD*	0	0	From GLGEN_*I2CCMD.REGADD*	0	

Table 38-41. I^2C Read Transaction - Word Read

1	7	1	1	8	1	8	1	1
S	Device Address	Rd	A	Data	A	Data	A	P
	From *GLGEN_I2CPARAMS.PHYADD*	1	0	Stored in *GLGEN_I2CCMD.DATA[7:0]*	0	Stored in *GLGEN_I2CCMD.DATA[15:8]*	0	

Note: When GLGEN_*I2CPARAMS.I2C_BYTE_ORDER* field is set, then the read data is considered as a 16 bit word. Thus, the first byte read is stored in GLGEN_*I2CCMD.DATA[15:8]* and the second byte read is stored in GLGEN_*I2CCMD.DATA[7:0]*.

When a byte read command (GLGEN_*I2CPARAMS.ACCESS_WIDTH* = 0b, GLGEN_*I2CCMD.OP* =1b) is given the following sequence is done by the 10 GbE controller.

Table 38-42. I^2C Read Transaction - Dummy Write

1	7	1	1	8	1
S	Device Address	Wr	A	Register Address	A
	From *GLGEN_I2CPARAMS.PHYADD*	0	0	From *GLGEN_I2CCMD.REGADD*	0

Table 38-43. I²C Read Transaction - Byte Read

1	7	1	1	8	1	1
S	Device Address	Rd	A	Data	A	P
	From GLGEN_*I2CPARAMS.PHYADD*	1	0	Stored in GLGEN_*I2CCMD.DATA[7:0]*	0	

When a word write command (GLGEN_*I2CPARAMS.ACCESS_WIDTH* = 1b, GLGEN_*I2CCMD.OP* =0b) is given the following sequence is done by the 10 GbE controller.

Table 38-44. I²C Write Transaction - Word Write

1	7	1	8	1		8	1	8	1	1
S	Device Address	Wr	Register Address	A		Data	A	Data	A	P
	From GLGEN_*I2CPARAMS.PHYADD*	0	From GLGEN_*I2CCMD.REGADD*	0		From in GLGEN_*I2CCMD.DATA[7:0]*	0	From in GLGEN_*I2CCMD.DATA[15:8]*	0	

Note: When *GLGEN_I2CPARAMS.I2C_BYTE_ORDER* field is set, then the data written is considered as a 16 bit word. Thus, the first byte written is taken from *GLGEN_I2CCMD.DATA[15:8]* and the second byte written is taken from *GLGEN_I2CCMD.DATA[7:0].*

When a byte write command (*GLGEN_I2CPARAMS.ACCESS_WIDTH* = 0b, *GLGEN_I2CCMD.OP* =0b) is given the following sequence is done by the 10 GbE controller.

Table 38-45. I²C Write Transaction - Byte Write

1	7	1	8	1	8	1	1
S	Device Address	Wr	Register Address	A	Data	A	P
	From GLGEN_*I2CPARAMS.PHYADD*	0	From GLGEN_*I2CCMD.REGADD*	0	From in GLGEN_*I2CCMD.DATA[7:0]*	0	

38.11 Link Management Topologies

The 10 GbE controller supports the MDIO and 2-wire management interface (I^2C) for connectivity to external modules, re-timers and PHYs. For example, SFP+ or QSFP+ optical and direct attached copper PHYs.

The 10 GbE controller supports up to four management interfaces (one per port) to control external PHY devices. Depending on the PHY type to manage these can be configured for either MDIO or 2-wire management interface. The configuration is done through control bits in the NVM.

Figure 38-11 and Figure 38-12 show the basic examples of connectivity between the MAC and an external module.

Figure 38-11.Basic Example of Module Connectivity (1 of 2)

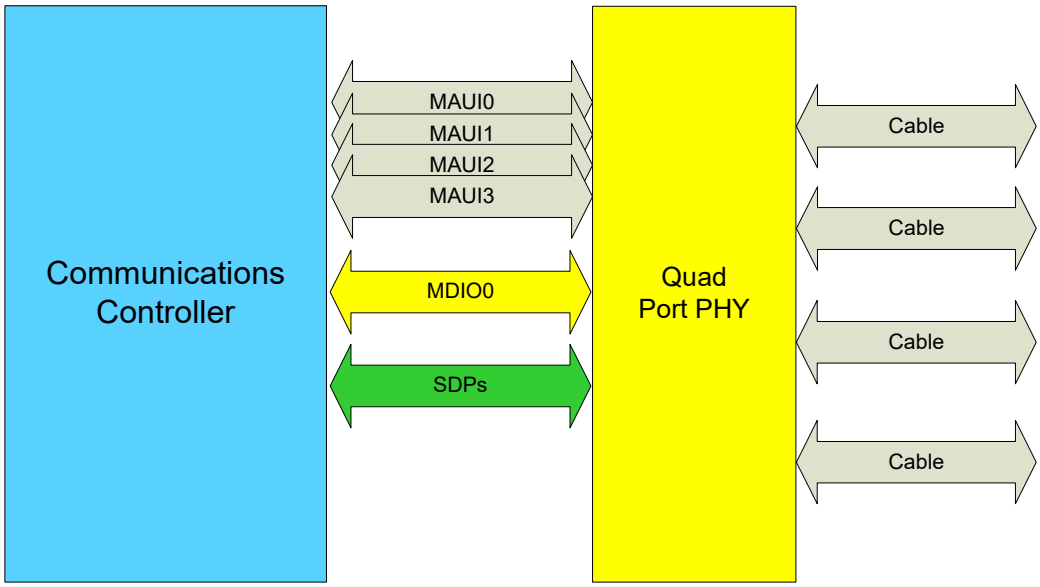
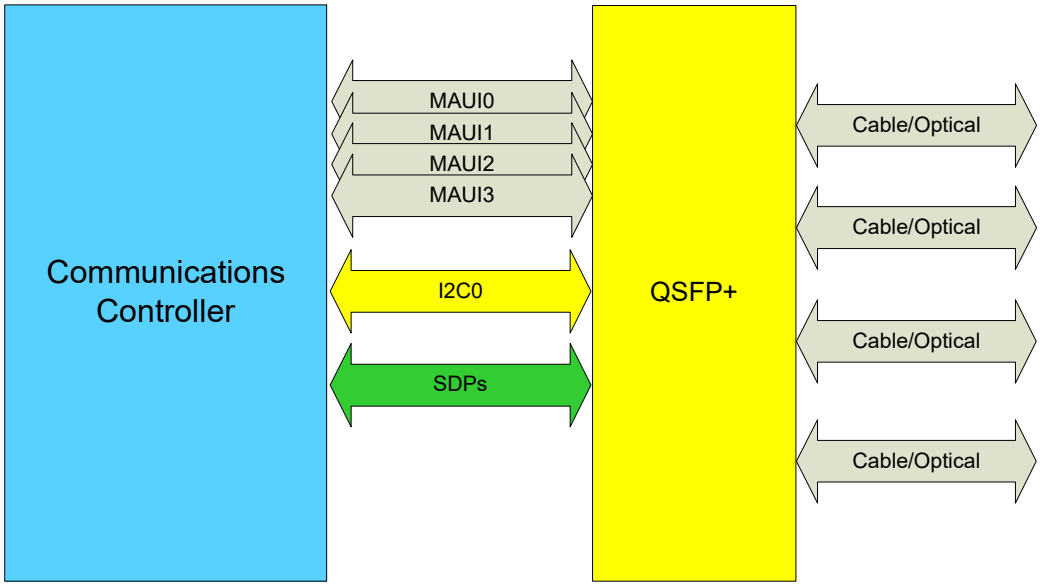


Figure 38-12.Basic Example Of Module Connectivity



The 10 GbE controller provides hardware acceleration of I^2C accesses over the two-wire management interface. The device driver manages the SFP+/QSFP+ modules using the Admin commands (see Section 38.11.3). The device driver does not have direct access to the I^2C bus except for diagnostic purposes. Firmware performs direct access to I^2C interface for reading and writing to the PHY modules.

The I^2C bus is accessed through registers GLGEN_I2CCMD and GLGEN_I2CPARAMS. I^2C accesses through the hardware controller can be performed through the GLGEN_I2CCMD register. I^2C accesses can also be performed using bit banging through

GLGEN_I2CPARAMS register if needed. Firmware can execute I^2C write/read operations by writing to the I^2C command register. The status of I^2C cycle completion can be performed by reading the GLGEN_I2CCMD register. Refer to the steps that follow to appropriately set the fields:

1. I^2C register address is placed in GLGEN_I2CCMD.DEVADD field.
2. PHY address is placed in GLGEN_I2CCMD.PHYADD field (typically this is 0xA0 or 0xA2 for modules).
3. Command for read or write operation is placed in the GLGEN_I2CCMD.OP field.
4. For a write operation, data to be written is placed in the GLGEN_I2CCMD.DATA field.
5. Successful completion of a write or read operation is indicated by the 10 GbE controller through *Ready* bit (GLGEN_I2CCMD.R). A read error is indicated if GLGEN_I2CCMD.E bit is set along with GLGEN_I2CCMD.R.
6. For a read operation, data can be read from GLGEN_I2CCMD.DATA field when the *Ready* (GLGEN_I2CCMD.R) bit is set.
7. A reset sequence can be sent to the I^2C bus before the actual write/read operation when GLGEN_I2CCMD.Reset bit is set.

38.11.1 External PHY Management Connectivity

The 10 GbE controller has designated 8 GPIO pins as SDPs for use with ports 0 through 3. However, depending on the board layout, system designers have the flexibility to configure any of the GPIO pins, irrespective of their name, as SDPs. This section describes the use of SDP pins for the management of external PHYs and optical/copper modules.

Firmware controls the SDP pins when used for specific hardware functions for PHY management. The device driver uses Admin commands (see Section 38.11.3) to configure and manage the PHYs; the device driver does not directly access the SDP pins except for diagnostic purposes.

Table 38-46 lists the recommended configurations for connecting the 10 GbE controller SDPs, MDIO/I^2C pins to external PHYs or optical/copper modules. The configurations include connectivity to QSFP+ modules, SFP+ modules and 10GBASE-T PHYs, in single, dual or quad port configurations. If mapping of these SDP pins to a specific hardware function is not required then the pins can be used as GPIOs.

Any of the SDP pins configured as an input for dedicated hardware functions can be configured to cause an interrupt to the firmware. The firmware in turn reads the appropriate PHY status registers (through MDC or I^2C interfaces) and might post an event and provide necessary status information to the device driver through Admin commands (see Section 38.11.3).

Table 38-46. Example for SDP, MDIO and I^2C Ports Usage for SFP+ and QSFP+ Modules (Sheet 1 of 2)

Port Num	Pin	SFP+	QSFP+	Quad 10GBASE-T Copper PHY
0	SDP0_0	Mod_ABS	ModPrsL	PhyInt
	MDIO_SDA0 MDC_SCL0	SDA SCL	MDIO_SDA0 MDC_SCL0	

Table 38-46. Example for SDP, MDIO and I²C Ports Usage for SFP+ and QSFP+ Modules (Sheet 2 of 2)

Port Num	Pin	SFP+	QSFP+	Quad 10GBASE-T Copper PHY
1	SDP1_0	Mod_ABS	ModPrsL	PhyInt
	MDIO_SDA1 MDC_SCL1	SDA SCL		
2	SDP2_0	Mod_ABS	ModPrsL	PhyInt
	MDIO_SDA2 MDC_SCL2	SDA SCL		
3	SDP3_0	Mod_ABS	ModPrsL	PhyInt
	MDIO_SDA3 MDC_SCL3	SDA SCL		

38.11.1.1 Transceiver Module Support

The 10 GbE controller MAUI interface with additional usage of low speed interface pins (SDP, I²C and MDIO I/Os) supports connection to transceiver modules compliant with the following Multi Source Agreements (MSAs):

- SFP+ — SFF-8431 Specifications for Enhanced 8.5 and 10 Gigabit Small Form Factor Pluggable Module SFP+ rev 1.0
- QSFP+ — SFF 8436 Rev 3.4 specifications for quad small form factor pluggable module (4 x 10 GbE)

38.11.1.1.1 10 GbE Controller's SFP+ Connectivity Scheme

The following diagrams show the low speed digital signals used when connecting the 10 GbE controller to SFP+ modules.

Note: The diagram shows the minimal connectivity required.

Figure 38-13.Quad 10 GbE SFP+

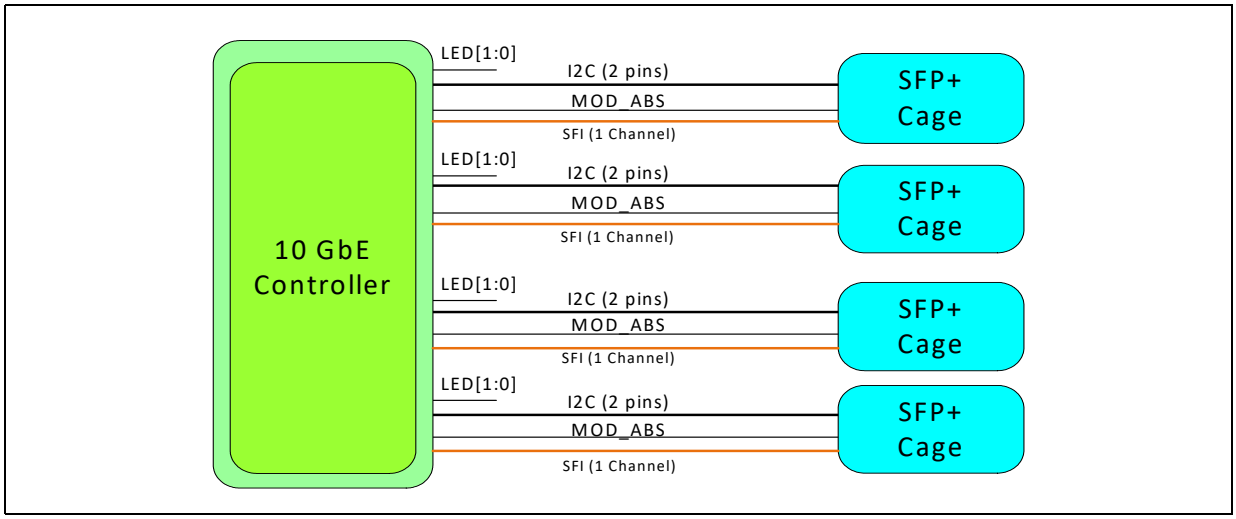
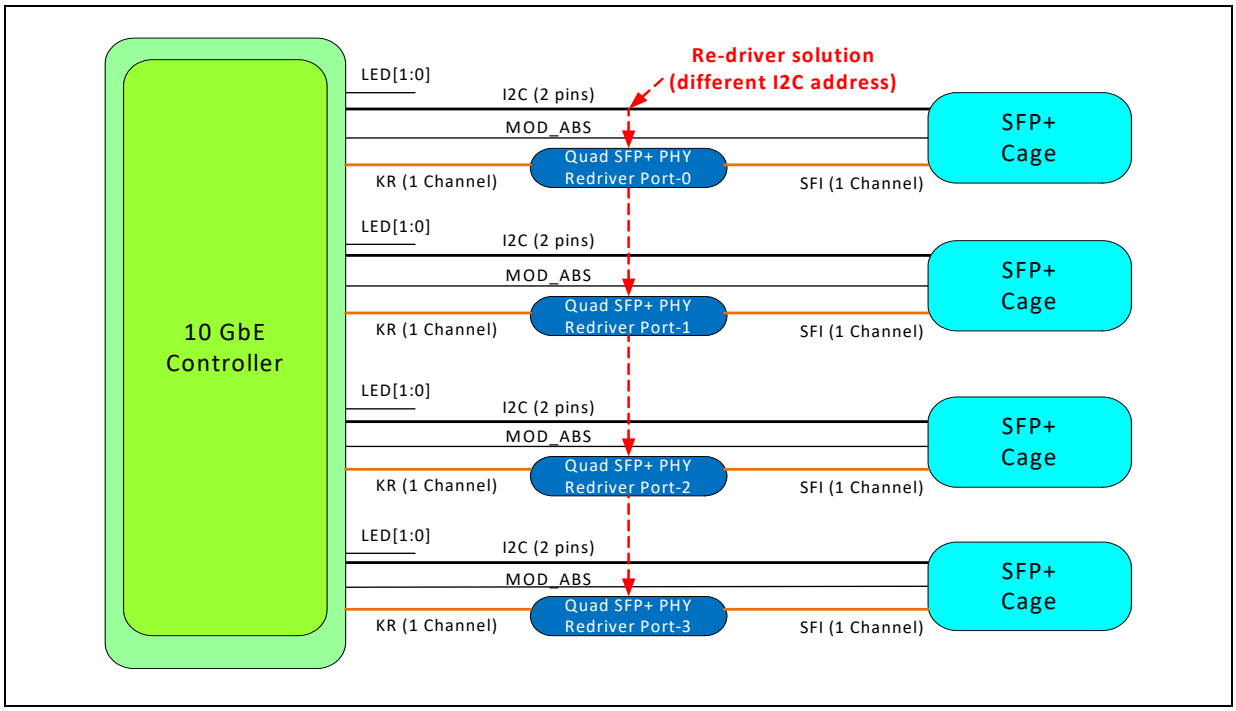


Figure 38-14.Quad 10 GbE SFP+ with Re-Driver PHY

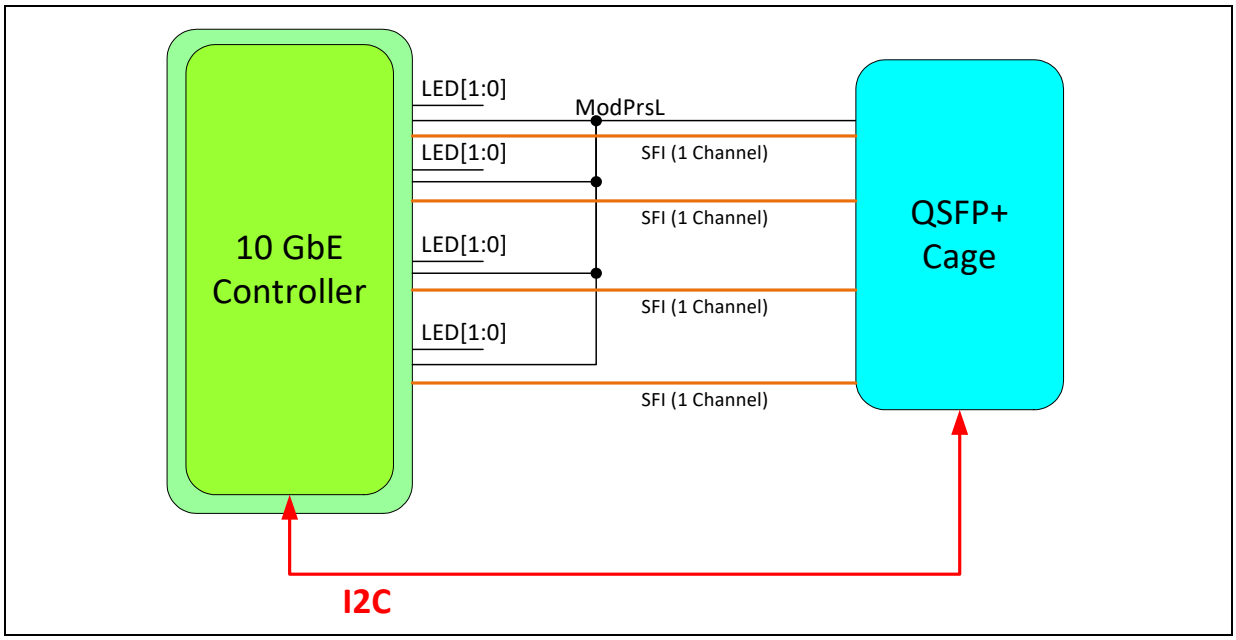


38.11.1.1.2 10 GbE Controller's QSFP+ Connectivity Scheme

The following diagram shows the low speed digital signals used when connecting the 10 GbE controller to QSFP+ modules.

Note: The diagram shows the minimal connectivity required.

Figure 38-15.Quad 10 GbE QSFP+

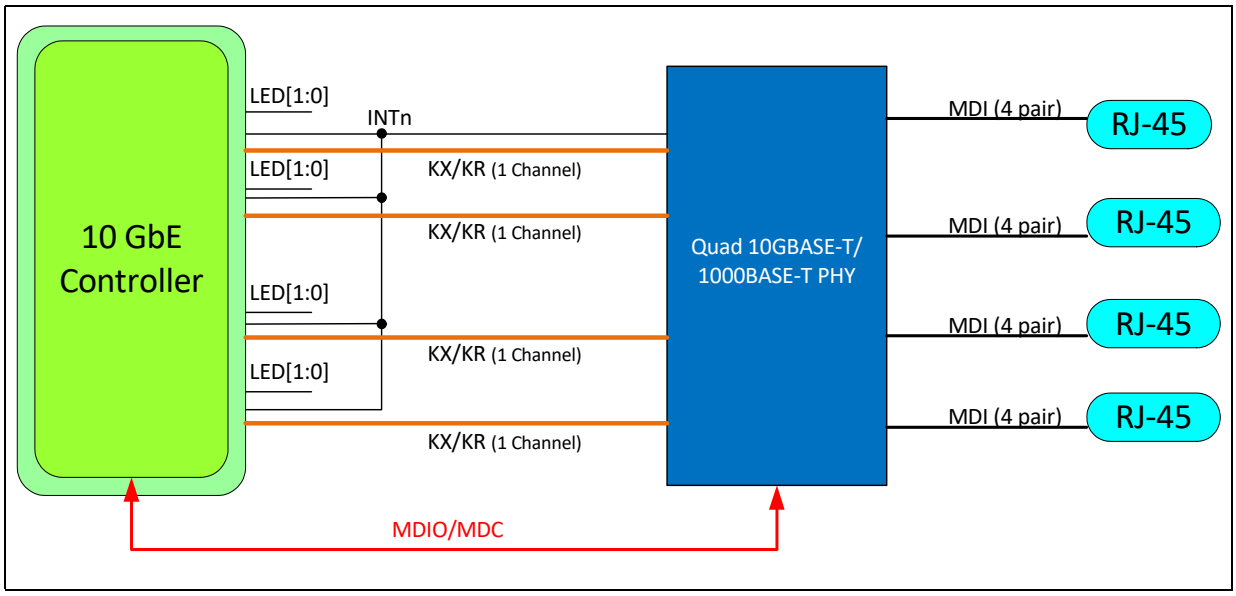


38.11.1.1.3 10GBASE-T/1000BASE-T Connectivity Scheme

The following diagram shows the low speed digital signals used when connecting the 10 GbE controller to 10GBASE-T/1000BASE-T PHYs.

Note: The diagram show the minimal connectivity required.

Figure 38-16.Quad 10GBASE-T/1000BASE-T



38.11.1.1.4 OCP Support

OCP specification (http://www.opencompute.org/wiki/Server/SpecsAndDesigns) defines a Mezz card form factor for NIC cards. This type of card is used to implement NICS based on standalone products. A variant of these cards that includes only a PHY can connect to an integrated MAC.

In the stand alone case, the NIC comes with its own NVM and thus can reflect to the system the currently connected PHY. In the case of PHY cards however, the NVM is tied to the chipset and is independent of the currently connected PHY.

This section describes the capability to the the 10 GbE controller MAC to detect the currently connected OCP PHY and expose it to the system and software adequately.

Supported Configurations

The following configurations are supported:

1. Quad port OCP connected to all 4 lanes of the 10 GbE controller.
2. Dual port OCP connected to lanes 0 and 1 of the 10 GbE controller. Lanes 2 and 3 can be connected to a LOM or left unconnected.
3. Dual port OCP connected to lanes 2 and 3 of the 10 GbE controller. Lanes 0 and 1 can be connected to a LOM or left unconnected.

Note: The port-to-lane mapping might change the ports mapped to the OCP. The I^2C associated to each port should follow the lanes. So, if port 0 is mapped to lane 1 and lane 1 is connected to OCP port 1, I^2C 0 should also be connected to I^2C 1 of the OCP.

Note: The 10 GbE controller supports only OCPs where port pairs are identical.

Use Cases

There are a few use cases expected when implementing an OCP card.

Table 38-47. OCP Use Cases

Use Model	Characteristics
OxM MFG Line	OxM assembles and configures system in their manufacturing flow NVM programming/configuration handled in MFG line Likely includes multiple boot cycles
Distributor	Stocks OEM standard components; configure/ship to order Mainboard contains OEM default NVM image Assembles system from bag-of-parts May be bare-metal assembly. For example, might not load or boot to the operating system)
IHV	Builds system-level solutions to specification for target applications or customers Assembles system from bag-of-parts; programs NVM image during system bring-up flow Expected to have system-level testing prior to shipment to end user
End-user	End user changes mezz card Expected to be a small subset of the usage models; mezz changes would be a relatively rare event. For example, boot flows should be optimized for a case where the mezz has not been changed since last boot)

OCP Card Detect By Firmware

As part of power on flow, if the *OCP Enable* NVM bit is set, firmware should read the first 3 words from the address as defined in the NVM field I^2C auto-detect address on the I^2C port defined by first *I2C auto-detect port* field in NVM and compare it with the signature value stored in the EMP SR section. The first two words contain the ASCII signature OCP_, to make sure that an OCP NVM is being read. The value in the EMP SR should be OCP_nn, where nn is the ASCII value (two bytes) that reflects the OCP card ID. Firmware should first compare the two-word signature to the EMP SR signature to verify that there is a valid OCP NVM at the auto-detect I^2C port and address. The third word is also compared in order to make sure that the OCP card was not replaced with another type of OCP card.

If the OCP ID EMP SR value matches the third word in the OCP card, then the NVM is already updated, and the init flow can proceed normally.

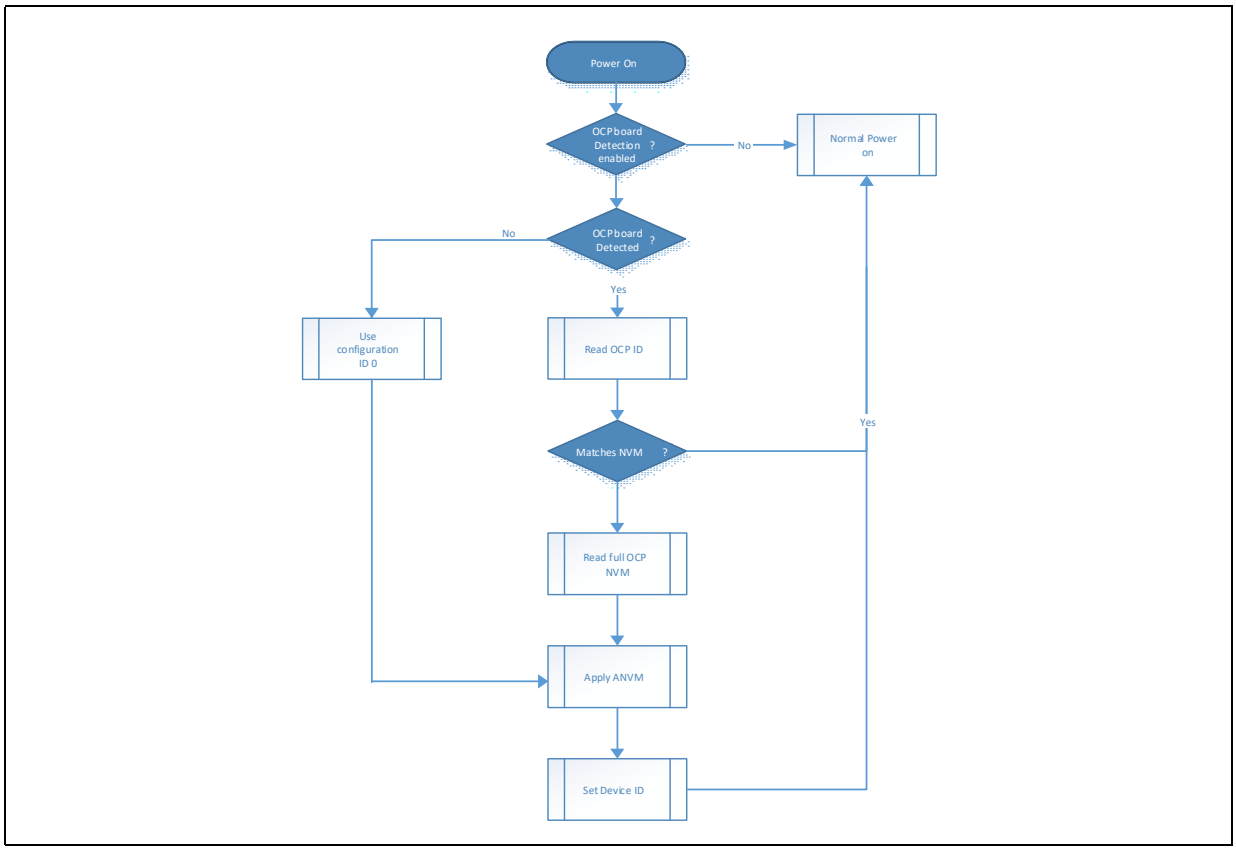
If firmware fails to read from the first I^2C port or reads different signature from the OCP_ signature in EMP SR (such as a valid OCP NVM is not detected), firmware should proceed and try searching for an OCP card at the second I^2C auto-detect port field as defined in NVM.

If a signature is valid, but the OCP ID does not match the EMP SR value, then firmware proceeds to update the NVM as described in the next section.

If an OCP card is expected (*OCP Enable* NVM bit is set), but no OCP card is detected (no valid signature read or I^2C read is not successful on both auto-detect ports), firmware should return the ENOSYS error in the Get PHY abilities AQC for ports that are defined as OCP ports.

Note: The initial value of the signature in the EMP SR might be set to OCP_00, which forces firmware to detect the OCP card as new card, as 0x0 is a reserved card ID that does not represent any valid card.

Figure 38-17.PHY Auto-Detect Flow



NVM Update of Auto-load

The following flow is used to update the NVM:

1. Add to the PCIe alternate auto-load section, an auto load of the device ID of function 0 to 0xFFFF (address - 0x000BE080. PFPCI_DEVID[0]), to indicate to the BIOS that an update is in process.

Note: The BIOS reads the ID approximately 30 seconds after boot; however, the update should finish in less than 30 seconds. Also, step 1 is required for future compatibility to BIOS. This step should be done before the PCI auto-load process. If a PCI auto-load occurs before the auto detect, the value is also 0xFFFF.

2. Read the OCP NVM and check the checksum.
   a. If it does not match, do not change the NVM. Firmware should report an ENOSYS error in the Get PHY abilities AQC for ports that are defined as OCP, but the OCP NVM is corrupted.
   b. Otherwise, continue.
3. In the shadow RAM area, update the following entries:
   a. New 10 GbE Controller signature should be updated to OCP_nn where nn is the ASCII value read from the OCP card.

b. The value of the super feature IDs should be updated according to the value read from the OCP NVM. If an OCP card is not present, index 0 of the configuration should be used. The super feature ID to apply each index to is set according to the mapping of *Port #n OCP connection* fields. For example, if *Port #2 OCP connection* = 1 (port 1), then the index read from offset 0x16:0x17 (OCP port #2) should be applied to super feature ID #F0FB (NIC port #1).

c. Value of the global super-feature configuration ID should be updated according to the values read from the OCP ID EEPROM, the number of ports in the system, number of OCP ports in the system, and number of ports on the OCP card, according to the following flow:

i. If total number of ports in system is two, choose 2-port global configuration.

ii. Else, if number of OCP ports in system is four and number of OCP ports in OCP card is two, choose 2-port global configuration.

iii. Else, choose 4-port global configuration (remaining cases are hybrid configurations, and pure OCP platforms with four OCP port cards).

Table 38-48. OCP NVM Content

Offset	Content
0x8:0x9	OCP global configuration — 4-port
0x10:0x11	OCP global configuration — 2-port
0x12:0x13	Port #0 of OCP configuration
0x14:0x15	Port #1 of OCP configuration
0x16:0x17	Port #2 of OCP configuration
0x18:0x19	Port #3 of OCP configuration

Note: the port #n configuration available entries are defined according to the number of ports field (offset 0xE:0xF). The 10 GbE controller POR supports up to four ports.

d. If the sub-vendor and subsystem IDs are not 0xFFFF, update the sub-vendor and subsystem IDs immediate fields.

4. Apply Adaptive Non-volatile Memory (ANVM)-like flow on the new shadow RAM content.

If one of the indexes is not found, remove report card.

5. Flush shadow RAM to NVM.
6. Remove the entry added at step 1 to PCIe Alternate auto-load section.

BIOS Flow

As part of enumeration of the 10 GbE controller, if the device ID of port 0 is 0xFFFF, firmware is in process of an ANVM-like flow. BIOS waits/revisits the 10 GbE controller later when the device ID has a valid value.

OCP ID NVM

The OCP card includes an I^2C accessible NVM (http://www.atmel.com/devices/AT34C02D.aspx) or equivalent containing the following information:

Table 38-49. OCP NVM content (Sheet 1 of 2)

Offset (byte)	Bits	Description
0x0:0x3	31:0	OCP NVM signature. ASCII value of OCP_.
0x4:0x5	15:0	OCP ID in ASCII. See the table that follows.
0x6:0x7	15:0	OCP Card Version

Table 38-49. OCP NVM content (Sheet 2 of 2)

Offset (byte)	Bits	Description
0x8:0x9	15:0	OCP Global Configuration ID (4-port)
0xA:0xB	15:0	OCP Global Configuration ID (2-port)
0xC:0xD	15:0	Subvendor ID
0xE:0xF	15:0	Subsystem ID
0x10: 0x11	15:0	Number of ports
0x12: 0x13	15:0	Port #0 Configuration ID
0x14: 0x15	15:0	Port #1 Configuration ID
0x16: 0x17	15:0	Port #2 Configuration ID
0x18: 0x19	15:0	Port #3 Configuration ID
0x1A:0xFD		Reserved. Should be zero
0xFE:0xFF	15:0	16-bit Checksum

OCP ID

This field is an ASCII decode of an ID identifying a given type of OCP cards. Two OCPs can have the same ID if and only if the content of all the other fields in the NVM are the same.

The following table defines the ID for the planned OCP cards.

Table 38-50. OCP Cards IDs

Card	ID	Description
No card	0x0	Reserved value to indicate no card. Might be used to indicate that the NVM is not configured with active card.
4-port 10GBASE-T	0x1	QP RJ-45; X557 QP PHY, KR input
2-port 10GBASE-T	0x2	DP RJ-45; X557 DP PHY, KR input
4-port 10G SFP+	0x3	QP Port SFP+, external PHY, KR input
2-port 10G SFP+	0x4	DP Port SFP+, external PHY, KR input
4-port 1000BASE-T	0x5	QP Port RJ-45, external PHY, KR input
2-port 1000BASE-T	0x6	DP Port RJ-45, external PHY, KR input

OCP Card Version

This field gives the OCP card version. The 8 MSBs are major version, while the 8 LSBs are the minor version. Currently, only major version = 1 cards are supported and if the major version changes, an error should be returned. The minor version tracks additive features that might be added to the OCP, without breaking the current functionality.

For example, all 1.x versions uses the same OCP ID format, versions higher than 1.0 might include additional fields describing parameters of the OCP. Firmware capable of reading only 1.0 format, is able to read 1.x, but ignores the additional fields.

Note: The major version has changed to two. Other version cards result in an error (default configuration 0x0 is chosen.

Full OCP Configuration ID

This field gives an ID identifying the ANVM configuration to apply to configure OCP-wide configurations. This typically includes the number of ports, etc. This value selects an option in the ANVM super-feature ID 0xF0FE.

Sub-vendor and Subsystem IDs

These fields enable the OCP vendor to override the value reported in the subsystem and sub-vendor fields in the PCIe configuration space. A value of 0xFFFF is ignored.

A configuration where only one of these two fields is equal to 0xFFFF is not valid.

Number of Ports

This field is used to define how many Port # Configuration ID entries are present. This field is not used to configure the number of exposed functions/ports. This number should be part of the full OCP configuration ANVM feature.

Port #n configuration ID

This field gives an ID identifying the ANVM configuration to apply to configure port #n OCP configurations. This typically includes the PHY type, the function device ID, LED configuration, etc. This value selects an option in the ANVM super-feature ID 0xF0FA/B/C/D according to the port number.

Checksum

Contains a checksum of the NVM calculated according to the following algorithm:

```
#define FIXED_SIZE 254/2
#define EEPROM_SUM 0xBABA

u16 checksum = 0;
u16 length = 0;
u16 word = 0;
int i;
for (i = 0; i < FIXED_SIZE; i++) {
word = Read OCP NVM address 2*i; // 16 bit read, address in byte resolution
checksum += word;
}

checksum = IXGBE_EEPROM_SUM - checksum;
```

Note: In version 1.x, the checksum is calculated over the first 254 bytes of the OCP NVM, to enable further expansion.

38.11.2 Firmware Link Configuration

In the 10 GbE controller all link configurations are performed by firmware. At POR firmware initializes the link and then provides a set of Admin commands, described in Section 38.11.3, to allow for software device drivers' modifications of the link parameters and configurations. All access to the link configurations is performed using the Admin commands and device drivers do not directly access the MAC, PHY or SDP settings. This is required in order to provide the software device driver with an API like interface to the MAC, PHY and SDP registers.

Link initialization by firmware includes reading the initial configuration from the NVM, reading the PHY type connected to the link and programming appropriate values to the MAC and PHY registers to bring up the link. See the detailed flow in Section 38.11.2.0.1 for more information.

The firmware also provides services to the device driver by providing status of the link, modifying the link configuration, and re-initializing the link if requested by the software device driver.

38.11.2.0.1 MAUI Link Setup Flow

The 10 GbE controller MAUI interface is configured at start up by firmware (before the device driver is loaded) in the following manner:

1. Firmware reads default settings from the NVM per port.
2. Firmware reads GPIO, MDC/I^2C settings pre-loaded by hardware from the NVM.
3. If link is setup for backplane connectivity:
   - Firmware waits for link to be established.
   - Based on resulting speed and parameters (FC) firmware configures other blocks in the 10 GbE controller.
4. If link is setup for module connectivity:
   - Firmware verifies that the connected module is a qualified module (when enabled)
   - Based on module type, optical or direct attach, firmware configures the PCS registers.
   - Firmware restarts the link setup.
5. If link is setup for BASE-T PHY connectivity (external PHY):
   - Firmware verifies that the connected PHY is valid.
   - Firmware programs external PHY's registers
   - Firmware restarts the auto-negotiation process on the BASE-T interface.
   - When link is established, firmware programs the internal MAC-PHY interface according to the negotiated speed and parameters (FC) as well as other relevant blocks in the 10 GbE controller.
6. When the internal link is established, firmware generates a LSE if enabled.

Note: If link synchronization is not successful, firmware does not report link-up and continuously polls link indications while waiting for link to be established.

If module or PHY qualification is enabled and connected device is not found in the qualified list, firmware stops the link setup process and reports event to software.

38.11.2.1 Software Link Management

At power up, firmware initializes the link based on the initial configurations loaded from the NVM. Following the initial setup, the software device driver might override any of the link settings; however, even though the software device driver is the owner of the link in SFP mode, it does not directly access the MAC or PHY registers and SDP pins controlling the PHYs instead all access to the link settings is performed through firmware as described in Section 38.11.3.

38.11.3 Link Configuration Admin Commands

The 10 GbE controller supports the following admin commands for configuring and managing the link. Software should use the admin commands to configure the link. This includes configuring the MAC and internal/external PHY devices. The firmware provides link configuration and status services to the device driver based on these admin commands.

Software can use the Get Link Status command to find out the actual status of the link.

Table 38-51. Link Configuration Admin Commands (0x06xx)

Command	Opcode	Description	Detailed Description
Set PHY Config	0x0601	Set various PHY configuration parameters on port.	Section 38.11.3.1.1
Set MAC Config	0x603	Set various MAC configuration parameters on the port.	Section 38.11.3.1.2
Setup Link and Restart[1] AN	0x0605	Sets up the link and restarts link auto-negotiation. This operation could bring down the link. This command needs to be executed for other set link parameters to take effect on the link.	Section 38.11.3.1.3
Get PHY Abilities	0x0600	Get various PHY abilities supported on the port.	Section 38.11.3.1.4
Get Link Status	0x0607	Get link status of the port.	Section 38.11.3.1.5
Link Status Event	0x0607	Firmware sends this asynchronous event notification to software when there is a change in status in any of the event causing conditions (such as link up/down or other link error conditions).	Section 38.11.3.1.6
Set Event Mask	0x0613	Sets event mask. Software can mask some or all of the link status event causing conditions.	Section 38.11.3.1.7

Notes:

1. In SFP mode, the Setup link and restart auto-negotiation command needs to be executed by the device driver in order for any other change in link parameters to take effect on the link. This operation could disrupt the link since the link state may toggle while the link is re-initialized with the new parameters.

38.11.3.1 Link Configuration Commands

This section provides a detailed description of the link configuration Admin commands and its structure.

38.11.3.1.1 Set PHY Config

This command is used by the device driver to set the various PHY configuration parameters supported on the port.

This is a Direct command. The set PHY command parameters data structure is placed in the descriptor.

Note: This command must be followed by the Setup Link And Restart Auto-Negotiation command in order for any changes to the link parameters to actually take place.

Table 38-52. Set PHY Config Command (opcode: 0x0601) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	1:0	0x0	
Opcode	2-3	Opcode	Command opcode.
Datalen	4-5	0x0	Must be 0b, value is ignored.
Return value/VFID	6-7		Return value. Zeroed by device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.

Table 38-52. Set PHY Config Command (opcode: 0x0601) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Set PHY Config	16-31	See Table 38-53.	16-byte data structure that holds the set PHY config command parameters listed in Table 38-53.

Table 38-53 lists the data structure of the set PHY config command parameters such as PHY type, PHY ID, speed ability, pause ability, etc.

Table 38-53. Set PHY Config Command Data Structure (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
PHY Type	0-3	PHY Type	PHY Type Supported On Port One bit per PHY type. The 10 GbE controller might be capable of supporting multiple PHY types. The following parameter indicates the bit number. 00 = Reserved. 01 = 1000BASE-KX. 03 = 10GBASE-KR. 06 - Reserved. 07 = SFI. 11 = 10GBASE-CR1 (used with 4x10 Gb/s QSFP+ direct attach copper). 17 = Reserved. 18 = Reserved. 19 = 10GBASE-T.
			20 = 10GBASE-SR (10 Gb/s SFP+ SR optical module). 21 = 10GBASE-LR (10 Gb/s SFP+ LR optical module). 22 = 10GBASE-SFP + Cu (direct attach copper). 23 = 10GBASE-CR1 (4x10 Gb/s QSFP+ CR over direct attach copper). 30:24 = Reserved. Other bits - Reserved, Must be zero.
			This parameter is used by the device driver to set the various PHY types' configuration parameters to be supported on the port. The port can be configured for a subset of the actual PHY types available on the port. The actual PHY types available are read by the device driver using Set PHY Config command. When auto-negotiation is enabled, the 10 GbE controller negotiates and selects one of the PHY types enabled. When auto-negotiation is disabled, this field should enable only a single value and then the 10 GbE controller is forced to operate in that selected mode.
Link Speed	4	Link Speed	Set link speed on port, only one operational speed is set by the device driver. Bit 4.1 = Reserved. Bit 4.2 = 1 Gb/s. Bit 4.3 = 10 Gb/s. Other bits - Reserved, Must be zero. This parameter is used by the device driver to set the operational link speed of the port. The 10 GbE controller might have the ability to support multiple link speeds on the same port that can be found using the Get PHY ability command (see Section 38.11.3.1.4). One of the link speeds can be enabled due to the result of auto-negotiation. This command can be used by the device driver to manually set the link speed when auto-negotiation is disabled.

Table 38-53. Set PHY Config Command Data Structure (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Pause ability	5.0:5.1	Pause Ability	5.0 set to 1b to enable IEEE 802.3x Tx link pause ability, or set to 0b to disable pause ability. 5.1 set to 1b to enable IEEE 802.3x Rx link pause ability, or set to 0b to disable pause ability. Auto-negotiation might have be restarted for this configuration to take effect over the link. This parameter is used by the device driver to set the IEEE 802.3x pause ability of the port. The 10 GbE controller's pause ability can be read by using the Get Link Status or Get PHY Abilities commands. The device driver might disable the IEEE 802.3x link pause ability using this command. If the link is already up and configured, the device driver needs to restart auto-negotiation, so the updated pause ability could be advertised to the link partner in order for the setting to take effect on the link.
Low power mode	5.2	Low Power Mode	Set to 1b to enable PHY in low power mode or set to 0b for normal operation. This field is valid only if the PHY supports low power mode. Certain external PHYs connected to the port might have the ability to support low power mode. Setting the PHY in low power mode affects normal operation. This bit should be set to 0b and set restart auto-negotiation to bring the link to normal operation from low power mode.
Enable Link	5.3	Enable Link	Set to 1b to enable the link. Set to 0b to disable the link. Device driver should not force link down when port is being used for manageability or WoL.
Enable AN	5.4	Enable AN	Set to 1b to enable auto-negotiation on the link.
Enable Atomic Link Update	5.5	Enable Atomic Link Update	When this field is set to 1b, firmware automatically executes the Setup Link And Restart Auto-Negotiation command following this command. When this field is set to 0b the device driver maintains the responsibility for sending the Setup Link And Restart Auto-Negotiation command. When automatic link update is enabled, the device driver should be aware that a link change event may occur following the Set PHY Config command.
Reserved	5.6:5.7	Reserved	Reserved, must be zero.
Low Power Control	12	$D3_{COLD}$ LPAN	$D3_{COLD}$ LPAN. Bit 12.0 = Set to 0b to disable $D3_{COLD}$ low power auto-negotiation. Bit 12.0 = Set to 1b to enable $D3_{COLD}$ low power auto-negotiation. Other bits = Reserved, must be zero.
Reserved	13-15	Reserved	Must be set to 0x0, value is ignored.

The following structure describes the response by firmware to the Set PHY Config command.

Table 38-54. Set PHY Config Command Response (Opcode: 0x0601)

Name	Bytes.Bits	Value	Remarks
Flags	1:0	0x0	
Opcode	2-3	Opcode	Command opcode.
Datalen	4-5	0x0	Must be 0x0, value is ignored.
Return Value/ VFID	6-7		Return value. 0x0 command success. Returns EPERM code if the operation is not permitted.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Param0	16-19	Reserved	Zeroed by firmware, value is ignored.
Param1	20-23	Reserved	Must be 0x0, value is ignored.
Data Address High	24-27	Reserved	Value 0x0.
Data Address Low	28-31	Reserved	

Note: When *Enable Automatic Link Update* is set to 1b, firmware sends a completion for the Set PHY Config command only after making all the necessary configuration changes and executing the Setup Link And Restart Auto-Negotiation command.

38.11.3.1.2 Set MAC Config

This command is used by the device driver to set the various MAC configuration parameters supported on the port. This status is indicated by the command response.

This is a direct command. The Set MAC command parameters data structure is placed in the command descriptor.

Table 38-55. Set MAC Config Command (Opcode: 0x0603)

Name	Bytes.Bits	Value	Remarks
Flags	1-0	0x0	
Opcode	2-3	Opcode	Command opcode.
Datalen	4-5	0x0	Must be 0x0b, value is ignored.
Return Value/ VFID	6-7		Return value. Zeroed by the device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Set MAC Config	16-31	See Table 38-56	16-byte data structure that holds the Set MAC Config command parameters is listed in Table 38-56.

Table 38-56 lists the data structure of the Set MAC Config command parameters such as max frame size, etc.

Table 38-56. Set MAC Config Command Data Structure (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Max Frame Size	0-1	Max Frame Size	16-bit value used to set the maximum frame size of the Ethernet frame on the port. This parameter is used by the device driver to set the maximum frame size on the port both for Rx and for Tx. This parameter should be set to the maximum expected L2 packet size. It is ~1.5 KB or ~9.5 KB depending if jumbo packets are expected on the link.
Reserved	2.0-2.01	Reserved	Must be 0x0.
CRC Enable	2.2	CRC Enable	Bit 0 = Set to 1b to enable the MAC to append the CRC on transmit. Set to 0b if software appends the CRC. This parameter is used by the device driver to enable the MAC to append the CRC on the link. This is the default configuration. This bit is set to 0b if software needs to disable the MAC to append CRC.
Pacing Config	2.3-2.6	Pacing Config	Bit 3:0 = This is 4 bit field that allows configuring PACE parameter in the MAC to slow down the effective data rate as defined in Table 38-36.
Auto Drop blocking packets	2.7		This bit controls the behavior when a no-drop packet is blocking a TC queue. 0b - The PF driver is notified. 1b - The blocking packet is dropped and then PF driver is notified.
Transmit Timer Priority	3	Transmit Timer Priority	This field selects the priority <n> whose Transmit Time Value field should be updated with the value provided in this command.
Transmit Timer Value	4-5	Transmit Timer Value	This is the priority <n> timer value that is included in the XOFF frames being transmitted. <n> is selected in the <Transmit Timer Priority> field previously described. <n> = 0 is used for Link Level FC.

Table 38-56. Set MAC Config Command Data Structure (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
FC Refresh Threshold	6-7	FC Refresh Threshold	This field represents the number of slot times before XOFF expires and a new XOFF is sent. This value is used to calculate the actual refresh period for sending the next pause frame if conditions for a pause state are still valid.
Reserved	8-15	Reserved	Must be 0x0.

The following data structure describes the response by firmware to the Set MAC Config command.

Table 38-57. Set MAC Config Command Response (Opcode: 0x0603)

Name	Bytes.Bits	Value	Remarks
Flags	1-0	0x0	
Opcode	2-3	Opcode	Command Opcode.
Datalen	4-5	0x0	Must be 0x0, value is ignored.
Return Value/ VFID	6-7		Return Value 0x0 = Command success. Returns EPERM code if the operation is not permitted.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Reserved	16-31	Reserved	Value 0x0.

38.11.3.1.3 Setup Link And Restart Auto-Negotiation

This command is used by the device driver to setup the link and execute previously sent Set PHY Config commands as well as restart the auto-negotiation over the link. This command needs to be executed for any change in link parameters, such as set link speed, etc., to take effect.

Table 38-58. Restart AN Command (Opcode: 0x0605)

Name	Bytes.Bits	Value	Remarks
Flags	1-0	0x0	
Opcode	2-3	Opcode	Command opcode.
Datalen	4-5	0x0	Must be 0x0, value is ignored.
Return value/VFID	6-7		Return value. Zeroed by device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Command Flags	16	Command	Bit 16.1: — Set to 1b to restart the link. Bit 16.2[1]: — Set to 1b to enable link. — Set to 0b to disable link. Other bits = Reserved, must be zero. This command maybe executed automatically by firmware, following a Set PHY Config command. In such a case these bits are assigned by firmware as follows: Bit 16.1 is set to 1b. Bit 16.2 is copied from the Set PHY Config <Enable Link> field.
Reserved	17-31	Reserved	Must be 0x0, value is ignored.

Notes:

1. Used by the device driver to enable/disable the link without modifying the other link settings. This is useful at POR when an application needs to have link powered down until the device driver loads.

The following structure describes the response by firmware to the Restart Auto-negotiation command.

Table 38-59. Restart Auto-negotiation Command Response (Opcode: 0x0605)

Name	Bytes.Bits	Value	Remarks
Flags	1-0	0x0	
Opcode	2-3	Opcode	Command opcode
Datalen	4-5	0x0	Must be 0x0, value is ignored.
Return Value/ VFID	6-7		Return Value 0x0 = Command success
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Reserved	16-31	Reserved	Value 0x0

38.11.3.1.4 Get PHY Abilities

This command is used by the device driver to find out the various PHY abilities supported on the port.

This is a Direct command.

Table 38-60. Get PHY Abilities Command (Opcode: 0x0600)

Name	Bytes.Bits	Value	Remarks
Flags	1-0	0x0	
Opcode	2-3	Opcode	Command opcode
Datalen	4-5	0x0	Must be 0x0, value is ignored.
Return value/VFID	6-7		Return value. Zeroed by the device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Param0	16-19		First Command Parameter 16.0 Report qualified modules. List of qualified modules will be part of the response only when this bit is set to 1b. 16.1 Report Active/Init. 0b = Report the ACTIVE values. For example, the values assigned to each parameter following the last Set PHY Config command. 1b = Report the initial values of the different fields. For example, the values loaded after the last reset event. All other bits are reserved.
Param1	20-23		Second command parameter
Data Address High	24-27	Buff Address	High bits of buffer address
Data Address low	28-31	Buff Address	Low bits of buffer address

Table 38-61 lists the get PHY abilities response structure returned by firmware to the Get PHY Abilities command.

Direct response, opcode and get PHY abilities response data structure are placed in the descriptor.

Table 38-61. Get PHY Abilities Command Response (Opcode: 0x0600)

Name	Bytes.Bits	Value	Remarks
Flags	1-0	0x0	
Opcode	2-3	Opcode	Command opcode
Datalen	4-5	0x0	Must be 0x0, value is ignored.
Return value/VFID	6-7		Return value. Zeroed by the device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Param0	16-19		Reserved. Must be set to 0x0.
Param1	20-23		Reserved. Must be set to 0x0.
Data Address High	24-27	Buff Address	Buffer address Buffer content is listed in Table 38-62.
Data Address Low	28-31	Buff Address	

Table 38-62 lists the data structure of the Get PHY Abilities command response. The command response returns various PHY parameters such as PHY type, PHY ID, speed ability, pause ability, etc. The following table lists the format of the buffer content for the Get PHY Abilities reply.

Table 38-62. Get PHY Abilities Command Response Data Structure (Sheet 1 of 3)

Name	Bytes.Bits	Value	Remarks
PHY Type	0-3	PHY Type	PHY Type Supported On The Port One bit per PHY type. The 10 GbE controller might be capable of supporting multiple PHY types. The following parameter indicates the bit number. 00 = Reserved 03 = 10GBASE-KR. 06 = Reserved. 07 = SFI. 11 = 10GBASE-CR1 (used with 4x10 Gb/s QSFP+ direct attach copper). 17 = Reserved 18 = 1000BASE-T. 19 = 10GBASE-T. 20 = 10GBASE-SR (10 Gb/s SFP+ SR optical module). 21 = 10GBASE-LR (10 Gb/s SFP+ LR optical module). 22 = 10GBASE-SFP+ Cu (direct attach copper). 23 = 10GBASE-CR1 (4x10 Gb/s QSFP+ CR over direct attach copper). 24 = Reserved Other bits - Reserved, must be zero. This parameter is used by the device driver to find out the various PHY types supported on the port. The 10 GbE controller might support multiple PHY types on the same port. One of the PHY types might be enabled due to the result of auto-negotiation or manually set by the firmware when auto-negotiation is disabled.

Table 38-62. Get PHY Abilities Command Response Data Structure (Sheet 2 of 3)

Name	Bytes.Bits	Value	Remarks
Link Speed Ability	4	Link Speed	Link rate supported on the port. Bit 4.1 = Reserved. Bit 4.2 = 1 Gb/s. Bit 4.3 = 10 Gb/s. Other bits - Reserved, must be zero. This parameter is used by the device driver to find out the various link speeds supported on the port. The 10 GbE controller might have the ability to support multiple link speeds on the same port. One of the link speeds might be enabled due to the result of auto-negotiation or manually set by the device driver when auto-negotiation is disabled.
Pause Ability	5.0:5.1	Pause Ability	Zero returns 1b if the port supports IEEE 802.3x Tx link pause or returns 0b otherwise. One returns 1b if the port supports IEEE 802.3x Rx link pause or returns 0b otherwise. This parameter is used by the device driver to find out the IEEE 802.3x pause ability of the port.
Low Power Ability	5.2	Low Power Ability	Returns 1b if the PHY has low power ability or returns 0b otherwise. Certain external PHYs connected to the port might have the ability to support low power mode. Setting the PHY to low power mode affects normal operation.
Link Mode	5.3	Link Mode	0b = Link is disabled. 1b = Link is enabled.
AN Mode	5.4	AN Mode	1b = AN is enabled. 0b = AN is disabled.
Enable Module Qualification	5.5	Enable Module Qualification	Returns 1b if a external module or PHY qualification check is enabled.
Reserved	5.6:5.7	Reserved	Reserved, must be 0b.
Low Power Control	12	D3ColdLPAN	$D3_{COLD}$ LPAN. Bit 0: 0b = $D3_{COLD}$ low power auto-negotiation disabled. 1b = $D3_{COLD}$ low power auto-negotiation enabled. Other bits = Reserved, must be zero.
Reserved	13-15	Reserved	Reserved, must be zero.
Current PHY ID/ Vendor OUI	16-19	PHY ID/OUI	This parameter is used by the device driver to find out the PHY/module ID connected on the port. If the 10 GbE controller is connected to an external BASE-T PHY: This four-byte field returns the {OUI, Manufacturer Model#, Revision ID} as defined in IEEE 802.3, 22.2.4.3.1 PHY Identifier (Registers 2 and 3). Bytes 17:16 = Register3. Bytes 19:18 = Register2. If the 10 GbE controller is connected to an external module: This field returns the three-byte vendor OUI of the module (MSB is padded with zeros).

Table 38-62. Get PHY Abilities Command Response Data Structure (Sheet 3 of 3)

Name	Bytes.Bits	Value	Remarks
Current Module Type	20-22	Module Type	Returns the three-byte module ID. First byte: Module identifier. Defined by SFP+ (Addr 0xA0, Byte 0) or QSFP+ (Addr 128, page 0) specifications. Second byte: The following bits might be set to indicate the supported technologies: 0 = SFP+ Cu passive. 1 = SFP+ Cu active. 4 = 10G Base-SR. 5 = 10G Base-LR. 7:6 = Reserved. Remaining bits are reserved. Third byte: GbE compliance code. Defined by SFP+ (Addr 0xA0, Byte 6) or QSFP+ (Addr 134, page0) specifications. This parameter is used by the device driver to find out the module type on the port when connected to external modules. For example, the 10 GbE controller might be connected to an SFP+ or QSFP+ optical or direct attached copper modules. The format of the module type returns the ID and Ethernet compliance code fields as defined in the SFP+ or QSFP+ specifications. There is no separate Ethernet compliance code for SFP+ copper modules. It is reported in a separate byte in SFP+ module. However, the 10 GbE controller uses the unused bits in second byte to report SFP+ direct attach cables.
Qualified Module Count	23		Number of qualified modules to be listed in the following bytes.
Qualified Module ID-n	24+n*24 - 47+n*24		This is a list of qualified modules that are supported by the 10 GbE controller and might be connected. The list contains a 24-byte field per module, based on IEEE Std 802.3 definition of device ID, containing: Vendor OUI (3 bytes). Reserved (1 bytes). Vendor Part# (16 bytes). Vendor Rev# (4 bytes).

38.11.3.1.5 Get Link Status

This command is used by the device driver to find out the link status of the port. Firmware returns link status = up when the link is available for transmission/reception. This command also returns other operating parameters of the link such as negotiated speed, PHY type, etc.

This is a Direct command.

Table 38-63. Get Link Status Command (Opcode: 0x0607) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	1-0	0	
Opcode	2-3	Opcode	Command opcode.
Datalen	4-5	0	Must be 0x0, value is ignored.
Return value/VFID	6-7		Return value. Zeroed by the device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.

Table 38-63. Get Link Status Command (Opcode: 0x0607) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Command Flags	16-17	Reserved	16.1:0. 0 = NOP: LSE notification value is not modified and Get Link Status response returns the most updated value of enable/disable. 1 = Reserved. 2 = Disable link status event notification to software. 3 = Enable link status event notification to software. See Section 38.11.3.1.6 for details on LSE and enabling/disabling LSE events. All other bits are reserved. Must be 0x0, value is ignored.
Reserved	18-31	Reserved	Must be 0x0, value is ignored.

The following structure describes the response by firmware to the Get Link Status command.

Table 38-64. Get Link Status Response (Opcode: 0x0607)

Name	Bytes.Bits	Value	Remarks
Flags	1-0	0x0	
Opcode	2-3	Opcode	Command opcode.
Datalen	4-5	0x0	Must be 0x0, value is ignored.
Return value/VFID	6-7		Return value. Zeroed by the device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Command Flags	16-17	Command Flags	Bit 0 = LSE enable: Enable link status event notification to software. Firmware sets this bit to 1b to indicate that LSE is enabled or sets to 0b if LSE is disabled. See Section 38.11.3.1.6 for further details on LSE and enabling/disabling LSE events. All other bits are Reserved. Must be 0, value is ignored.
Get Link Status	18-31	See Table 38-65	14-byte data structure that holds the Get Link Status command response parameters listed in Table 38-65.

Table 38-65 lists the data structure of the Get Link Status command parameters such as link up/down, negotiated/operating speed, fault conditions, etc.

Table 38-65. Get Link Status Response Data Structure (Sheet 1 of 3)

Name	Bytes.Bits	Value	Remarks
PHY Type	0	PHY Type	Returns operating PHY type or PHY type negotiated if auto-negotiation is enabled. The 10 GbE controller might be capable of many different PHY types but only one PHY type is enabled as result of configuration or auto-negotiation. This parameter is an 8-bit integer, each value corresponds to a PHY type as follows. 0x0 = Reserved. 0x1 = 1000BASE-KX. 0x3 = 10GBASE-KR. 0x6 = Reserved 0x7 = SFI. 0xB = 10GBASE-CR1 (used with 4x10 Gb/s QSFP+ direct attach copper). 0xC = SFP+ active direct attach. 0xD = QSFP+ active direct attach. 0x12 = 1000BASE-T. 0x13 = 10GBASE-T. 0x14 = 10GBASE-SR (10 Gb/s SFP+ SR optical module). 0x15 = 10GBASE-LR (10 Gb/s SFP+ LR optical module). 0x16 = 10GBASE-SFP+ Cu (direct attach copper). 0x17 = 10GBASE-CR1 (4x10G QSFP+ CR over direct attach copper). 0x1D:0x1B = Reserved. 1E = Other values are not used. This parameter is used by the device driver to find out the operating PHY type on the port. One of the PHY types might be enabled due to the result of auto-negotiation/parallel detection or manually configured by firmware or software when auto-negotiation is disabled.
Link Speed	1	Link Speed	Returns operating link speed of the port. The PHY might be capable of many speeds but only one speed is enabled as result of configuration or auto-negotiation. This parameter is an 8-bit field, each bit corresponds to a link speed as follows. Only one bit is set at any given time. 1.1 = Reserved. 1.2 = 1000 Mb/s. 1.3 = 10 Gb/s. Other = Reserved, must be zero. This parameter is used by the device driver to find out the operating Link speed on the port. The link might be enabled at one of the link speeds due to the result of auto-negotiation/parallel detection or manually configured by firmware or software when auto-negotiation is disabled.
Link Status	2.0	Function Link Status	Returns 1b if link status = up or returns 0b if the link status = down. In SFP setup it is the port link status. This parameter indicates if the Link is up and ready for data communication.
Link Fault	2.1:2.4	Link Fault	Bit 2.1 = Returns 1b if PHY has detected a link fault condition. The fault could be anywhere in the PHY layer and either on transmit or receive local or remote fault. The following bits provide additional information about a link fault condition. Bit 2.2 = Returns 1b if a transmit link fault condition is detected, 0b otherwise. Bit 2.3 = Returns 1b if a receive link fault condition is detected, 0b otherwise. Bit 2.4 = Returns 1b if a remote fault condition detected, 0b otherwise.
External Port Link Status	2.5	Port's Link Status	Returns 1b if link status = up or returns 0b if the link status = down. This bit returns always the port's link status in SFP mode.

Table 38-65. Get Link Status Response Data Structure (Sheet 2 of 3)

Name	Bytes.Bits	Value	Remarks
Media Available	2.6	Media Available	Returns 1b if media is available for normal link communication or returns 0b otherwise. This parameter is used by the device driver to find out if the media is available on the port. When connected to an external module, this command returns if the media is plugged in and is available for normal communication. ***Note:*** This field is not relevant when connecting to an external 10GBASE-T PHY.
Signal Detect	2.7	Signal Detect	Returns 1b if a receive signal is detected by the PHY or module, or returns 0b otherwise. In the case of external PHYs, for example, this maps to the global signal detect function in the PHY and in some of the PHYs this maps to energy detect function on the link. In the case of external modules, this maps to the inverse of loss of signal function.
AN Completed	3.0	AN Completed	Returns 1b if auto-negotiation completed successfully or returns 0b otherwise. This bit is valid only if the PHY supports auto-negotiation and auto-negotiation is enabled. Typically auto-negotiation is enabled for backplane and BASE-T PHYs.
LP AN Ability	3.1	LP AN Ability	Returns 1b if the link partner is able to perform auto-negotiation or returns 0b otherwise. This bit is valid only if the PHY supports auto-negotiation and auto-negotiation is enabled.
Parallel detection Fault	3.2	Parallel Detection Fault	Returns 1b if the PHY detects parallel detection fault or returns 0b otherwise. This bit is valid only if the PHY supports auto-negotiation with parallel detection enabled.
FEC Enabled	3.3	FEC Enabled	Returns 1b if FEC is enabled on the link or returns 0b otherwise. This bit is valid only for backplane KR. FEC might be enabled on the link during auto-negotiation.
Low Power State	3.4	Low Power State	Returns 1b if the PHY is in a low power state or returns 0b otherwise.
Link Pause Status	3.5:3.6	Link Pause Status	Bit 3.5 - Returns 1b if Tx link pause is enabled on the link during auto-negotiation or returns 0b otherwise. Bit 3.6 = Returns 1b if Rx link pause is enabled on the link during auto-negotiation or returns 0b otherwise. Link pause should be disabled if PFC is enabled on the link. Simultaneous operation of link pause and PFC is not supported.
Qualified Module	3.7	Qualified Module	When the 10 GbE controller is connected to an external SFP+/QSFP+ module, this field indicates if the module is a qualified module whose OUI matches one of the pre-defined qualified modules. 0b = Module was not found in pre-configured list of qualified modules. 1b = Module is qualified.
PHY Temp Alarm	4.0	PHY Temp Alarm	Returns 1b if a temperature alarm condition is reported by the PHY or returns 0b otherwise. Typically an external PHY generates a temperature alarm condition by signaling a PHY interrupt to firmware. The temperature alarm feature should be enabled in the PHY to generate this condition.
Excessive Link Errors	4.1	Excessive Link Errors	Returns 1b if an excessive errors over the link condition is reported by the PHY or returns 0b otherwise.
Port TX Suspended	4.2-4.3	Port TX Suspended	0 = Port's Tx active. 1 = Port's Tx suspended and drained. 2 = Reserved. 3 = Port's Tx suspended and drained. Blocked TC pipe flushed.
Reserved	4.45-4.7	Reserved	Reserved.
Reserved	5	Reserved	
Max Frame Size	6-7	Max Frame Size	Maximum frame size set on this port.
Reserved	8.0-8.1		Must be 0b.
CRC Enable	8.2	CRC Enable	1b = CRC append is enabled on this port. 0b = CRC append is disabled on this port.

Table 38-65. Get Link Status Response Data Structure (Sheet 3 of 3)

Name	Bytes.Bits	Value	Remarks
Pacing Config	8.3-8.6	Pacing Config	This is 4-bit field that enables configuring a pace parameter in the MAC to slow down the effective data rate as listed in Table 38-36.
Reserved	8.7		Must be 0b.
Reserved	9-13	Reserved	Reserved.

38.11.3.1.6 Link Status Event (Opcode: 0x0607)

The Link Status Event (LSE) is generated by firmware to the device driver when there is a change in status in any of the event causing conditions. Event causing conditions listed in Table 38-66 can be individually masked from generating LSE by using the Set Event Mask Command (See Section 38-67). The LSE uses the same link status response data structure listed in Table 38-64 and Table 38-65. Firmware posts this data structure to the admin receive queue with the Flags.*CMP* bit cleared indicating that this is an asynchronous event generated by firmware (such as the message is not in response to an AQ command from software).

The LSE is disabled by default, unless explicitly enabled by software. Software enables an LSE by setting the *LSE Enable* bit when issuing Get Link Status command (See Table 38-63). Firmware disables the LSE immediately after generating an LSE and does not queue further events until LSE is explicitly enabled by software by the Get Link Status command. Firmware also indicates the LSE enabled status through the *LSE Enable* bit in the Get Link Status command response data structure (See Table 38-64).

The LSE is only generated by firmware to respective PF drivers and it is software's responsibility to communicate relevant link status change events to the VF through appropriate PF to VF communication mechanisms. Software is not expected to use any hardware link status interrupt mechanisms. Hardware link status change interrupts are provided only for diagnostic use. Hence, hardware link status interrupts to PFs and VFs should be disabled for normal operation. Software should use the AQ mechanism to get the link status change notifications using the Get Link Status command and LSE.

Table 38-66. Reported Link Events

Event	Description
Link Change	Link state change. For example, the link state changes from link up to link down.
Media Not Available	Event is reported when an external module is pulled out of its cage.
Link Fault	
PHY Temperature Alarm	Event is generated when an external PHY or module generates a temperature alarm interrupt.
Excessive Errors	
Signal Detect Condition	Signal detect of Link Partner's laser signal indication has been asserted or de-asserted.
Auto-Negotiation Completed	
Module Qualification Failure	When working with external modules, firmware might be enabled to perform a validation process where the module ID parameters are compared with a per-configured, NVM loaded, list of qualified modules. If, qualification check is enabled and connected module is not found in the list, then firmware terminates the link initialization process and then generates this event.
Port Tx Suspend	Indicates that the port's Tx data path is temporarily suspended for configuration purposes.

38.11.3.1.7 Set Event Mask

This command is used by the device driver to mask the event causing conditions of the link status event from firmware. The link status event is generated by firmware to the PF as described in Section 38.11.3.1.6.

This is a Direct command.

Table 38-67. Set Event Mask Command (Opcode: 0x0613)

Name	Bytes.Bits	Value	Remarks
Flags	1-0	0x0	
Opcode	2-3	Opcode	Command opcode.
Datalen	4-5	0x0	Must be 0x0 value is ignored.
Return Value/ VFID	6-7		Return value. Zeroed by the device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Data Address High	16-19	Reserved	Value 0x0.
Data Address Low	20-23	Reserved	
Event Mask	24-25	Event Mask	
Reserved	26-31	Reserved	

The following structure describes the response by firmware to the Set Event Mask command.

Table 38-68. Set Event Mask Command Response (Opcode: 0x0613)

Name	Bytes.Bits	Value	Remarks
Flags	1-0	0x0	
Opcode	2-3	Opcode	
Datalen	4-5	0x0	
Return Value/ VFID	6-7		
Cookie High	8-11	Cookie	
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Reserved	16-31	Reserved	Must be 0x0, value is ignored.

38.11.3.2 PF-to-physical Lane Mappings

Table 38-69. PF-to-physical Lane Mappings (Supported Configurations) (Sheet 1 of 2)

Cfg ID	Application	10 GbE Controller Interface Protocol	PF Assignment to Port
3.0 3.01	Quad Port Backplane 10G/1G	KR	
		KX	
		AN	
3.6 3.61	Quad Port Backplane 10 GbE/1 GbE	KR	PF0 to P3 PF1 to P2 PF2 to P1 PF3 to P0
		KX	
		AN	
6.00	Quad Port 10 GbE SFP+	SFI	
6.01	Quad Port 1 GbE SFP+	KX	
6.30	Quad Port QSFP+ 10 GbE/1 GbE plus re-driver	KR	
6.31		KX	
7.00	Quad Port 10 GbE SFP+	SFI	PF0 to P3 PF1 to P2 PF2 to P1 PF3 to P0
7.01	Quad Port 1 GbE SFP+	KX	
7.30	Quad Port SFP+ 10 GbE/1 GbE plus re-driver (Inphi)	KR	
7.31		KX	
8.30	Quad Port QSFP+ 10 GbE/1 GbE	SFI	
8.31		KX	
9.00	Quad OCP	SFI/KR/KX	
9.3	Dual OCP plus Dual X550	2xKR plus 2xSFI/KR/KX	
9.6	Dual OCP		
10.0	Quad Port BASE-T 10 GbE X550	KR	
10.01	Quad Port BASE-T 1 GbE X550	KX	
		AN	
10.10 10.11	Quad Port BASE-T 1 GbE Marvell	KX	
11.0	Dual Port Backplane 10 GbE/1 GbE	KR	
		KX	
		AN	
	Dual Port BASE-T 10 GbE/1 GbE X550	KR	
		KX	
		AN	

Table 38-69. PF-to-physical Lane Mappings (Supported Configurations) (Sheet 2 of 2)

Cfg ID	Application	10 GbE Controller Interface Protocol	PF Assignment to Port
12.0	Dual Port Backplane 10 GbE/1 GbE	KR	
		KX	
		AN	
	Dual Port 10 GbE SFP+ Dual Port 1 GbE SFP+	SFI/CR1	
		KX	
12.01	Dual Port Backplane 10 GbE/1 GbE	KR	
		KX	
		AN	
	Dual Port 10 GbE SFP+ Dual Port 1 GbE SFP+ plus re-driver	SFI/CR1	
		KX	
13.0	Dual Port BASE-T 10 GbE/1 GbE X550	KR	
		KX	
		AN	
	Dual Port 10 GbE SFP+ Dual Port 1 GbE SFP+	SFI/CR1	
		KX	
13.01	Dual Port BASE-T 10 GbE/1 GbE X550	KR	
		KX	
		AN	
	Dual Port 10 GbE SFP+ Dual Port 1 GbE SFP+ plus re-driver	SFI/CR1	
		KX	

38.11.4 Non-Volatile Memory (NVM)

38.11.5 General Overview

There are new conditions specific to the 10 GbE controller that induced a new approach concerning NVM access:

- LAN and RDMA traffic can be handled only if the EMP code runs.
- For flexibility reasons, the main EMP code is retrieved from the NVM and not from ROM.

38.11.5.1 Requirements on NVM Access

The basic requirements from NVM access in the 10 GbE controller include the following:

1. Guarantee that only Intel provided firmware code (EMP, PE) is run by the 10 GbE controller.
   a. Intel prevents older NVM images from replacing newer NVM images with an incremented rollback revision number. Note the rollback revision number is separate from any other version fields, that might be used to distinguish versions of NVMs that modify functionality or device settings.

2. Protect NVM update flow from power failure before completion. This implies the image-update procedure of modules uses a double-bank policy.
3. Meet the boot time requirements.
4. Guarantee that NVM settings that are critical for the 10 GbE controller's accessibility from the host be provided only by Intel, but can still be replaced in the field with another Intel provided setting if necessary. It relates to the following NVM module:
   a. RO PCIR registers auto-load

Requirements 1 and 4 imply that the contents of several modules be protected via authentication while others be made RO. Refer to Section 38.11.13 for details about NVM authentication.

38.11.5.2 Operational Limitations

The NVM protection method selected in the 10 GbE controller relies on an authenticate on update concept. It means that the protected modules are not authenticated after initialization, but only before committing to a module update operation. NVM protection is guaranteed by an induction authentication chain, starting from a first secured NVM image and requiring that any step taken later on for modifying the image be secured.

This method induces several limitations and restricted working assumptions:

1. A first good EMP image is loaded into the Flash at the manufacturing site that is assumed to be safe. For example, it can be done as part of the full chipset NVM programming.
   a. It assumes customers (OEM and end-user) know the source of the installed components, the supply chain producing these components is not compromised during manufacturing, and after being deployed the NIC/LOM is *physically* protected from modification.
   b. The possibility exists that unauthorized firmware be loaded into the 10 GbE controller via physical modification post manufacturing, as well as supply chain vulnerabilities. However, firmware updates via programmatic (software) methods are enhanced to require authentication prior to updating NVM settings. Furthermore, host software can independently detect whether the firmware image has an invalid digital signature.
2. In normal operating mode, NVM write accesses are controlled by the 10 GbE controller (by EMP) and cannot be performed via the memory mapped accesses. Memory mapped NVM access remains available for NVM read accesses only. For simplicity and flexibility reasons, NVM write accesses (except for VPD) can be initiated only via an admin command or following to a BMC command, which are both handled by the EMP.
3. All supported Flash parts share the same set of opcodes used by the 10 GbE controller.
4. The blank Flash programming mode is not safe, and must therefore be used only as a last resort to recover from initial mis-configuration. This programming mode is entered either:
   a. When a blank Flash is detected.
   b. When the EMP image loaded (even if authenticated by Intel) does not belong to the 10 GbE controller.

When the host debug mode is entered via DEBUG_EN pin, the host can remove address protection of the PF space and write into the register that controls the blank Flash programming mode.

38.11.6 Shared External Flash

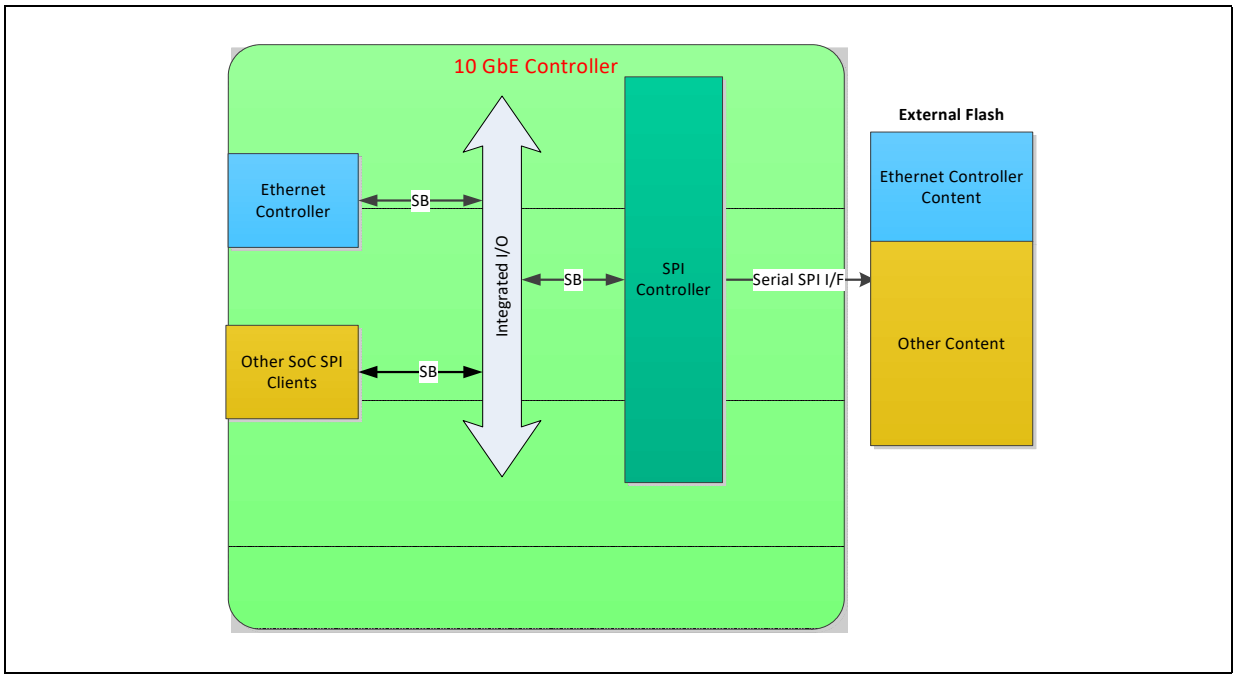
In the 10 GbE controller, the external Flash is shared between several soft IPs. In order to manage the arbitration between these different clients as well as for simplicity and centralization of the physical access logic an additional soft-IP is added: the SPI controller.

The SPI controller is responsible for the following tasks:

- Perform arbitration between the different clients.
- Manage the access control to the different regions.
- Reflect a zero-base address to each of the clients such as each client's first address is address zero.
- Enable access for a software tool for global blank Flash programming.

As shown in Figure 38-18, NVM sharing enables sharing of an external Flash device for the 10 GbE controller and other 10 GbE controller clients.

Figure 38-18.Sample Shared External Flash Illustration



38.11.6.1 Shared External Flash - Blank Flash Programming

In devices supporting a shared external Flash, there are two levels of blank Flash programming, global and local.

Global blank Flash programming is a scenario where the entire Flash content is blank. This condition is identified by the SPI controller when performing its first access to the Flash in order to read its own header. In such a scenario, it is the responsibility of the SPI controller to enable access for a tool to program the full Flash content and it is up to the global Power Control Unit (PCU) to make sure that the 10 GbE controller is not taken out of reset before the Flash has been programmed.

Local blank Flash programming is a scenario where due to some local data error only the 10 GbE controller region has been corrupted and only that single region needs to be re-programmed. In this scenario, it is up to the 10 GbE controller to signal the software device driver of such a condition and to enable host access for re-writing the specific corrupted region. In this case, the host access is done through the 10 GbE controller's sideband direct access. See Section 38.11.8.2 for more details.

38.11.7 Shadow RAM

NVM modules that meet one of these criteria must also be mirrored internally into a shadow RAM that is loaded once after POR:

1. Software must be able to partially update the module without being forced to rewrite the entire module (like SMBus address, VPD, etc.).
   Unlike an EEPROM, Flash devices require rewriting an entire sector even if it comes to updating a single byte. The partial update is first performed against the shadow RAM. Later on, the 10 GbE controller commits the entire updated shadow RAM into the Flash.
2. On device-level resets (in contrast to function-level resets), the module (or parts of it) is auto-loaded by the 10 GbE controller into registers that are mapped to the host memory BAR.
   Auto-load done after PCIe fundamental resets (POR and PERST# must be completed within a bounded time, and cannot wait for the delays involved by a sector erase operation (hundreds of ms) that could have been initiated just before. Flash read accesses are suspended until a Flash sector erase operation completes. NVM auto-load performed further to device-level resets are done from the internal shadow RAM into the registers, without involving Flash read cycles.

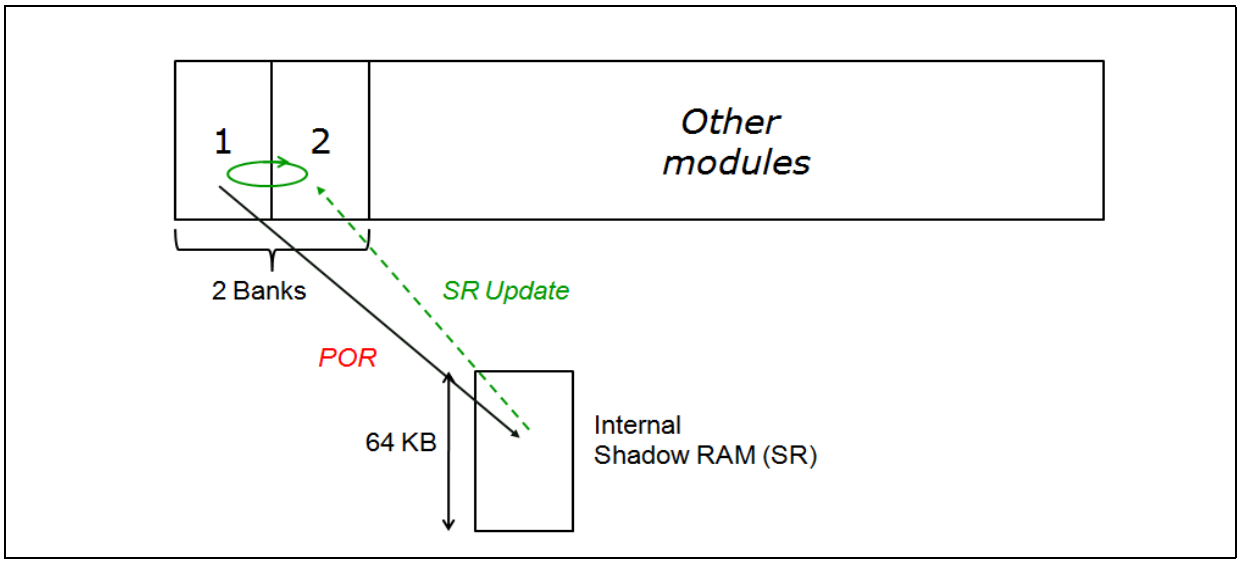
The 10 GbE controller maintains the first 32 x 4 KB sectors of the Flash area allocated to it for the configuration content that must be mirrored into the shadow RAM. These sectors are organized in two equally sized banks, each one capable of containing the entire shadow RAM contents. These banks are referred to as the basic NVM banks. At any time one of these two banks must be valid or else the 10 GbE controller is set by hardware default and enters into blank Flash programming mode (refer to Section 38.11.8.2).

Following a Power On Reset, the 10 GbE controller copies the valid bank of the Flash device into the internal shadow RAM, which is made resilient to device-level resets that might occur later on. The valid bank is the lowest indexed bank with the validity field content read as 01b. The NVM *Validity* field is located at NVM Control Word 1. At any time, the valid bank is referred to as the current basic bank, while the other is referred as the next basic bank. Any further accesses of software to this section of the NVM are directed to the internal shadow RAM. Modifications made to the shadow RAM content are then copied by the 10 GbE controller into the next bank of the NVM, flipping circularly the valid bank between bank 0 and bank 1 of the Flash.

This mechanism also provides a way for software to protect an image-update procedure from power down events by establishing a dual-bank policy even when performing a module partial update.

Figure 38-19 shows the shadow RAM mapping and interface.

Figure 38-19.NVM Shadow RAM



38.11.8 NVM Access Modes

38.11.8.1 Normal Mode

For BIOS read accesses and VPD accesses, any read or write access to the NVM by the host must be preceded by taking ownership of the NVM resource via the Request Resource Ownership admin command. This prevents the following situations:

1. Reading a module that is currently being modified by another entity.
2. Concurrent modifying a module contents.

38.11.8.1.1 Normal Read Access

Read accesses to the NVM do not require the EMP being involved (except if performed via the NVM Read admin command). Available read accesses are as follows:

1. An NVM Read admin command from the PF.
2. VPD register set. The EMP asserts the *Done* bit.
3. Memory mapped read via the memory/expansion ROM BAR. To save host memory addresses, memory BAR access to the NVM is not always available. It is enabled/disabled by setting the Flash_Expose bit in the NVM (or setting the GLPCI_LBARCTRL.FLASH_EXPOSE CSR bit).

38.11.8.1.2 Normal Write Access

Write accesses to the NVM are controlled by the EMP. Two accesses are provided:

1. An NVM Update admin command from the PF.
2. VPD register set. The EMP asserts the *Done* bit.

NVM write access attempts performed via the memory/expansion ROM BARs are not performed by the 10 GbE controller, although PCIe transactions are (apparently) normally completed.

38.11.8.2 Blank Flash Programming Mode

The 10 GbE controller enters blank Flash programming mode based on the following:

1. When a blank Flash is detected. It means that the NVM *Validity* field (NVM Control Word 1) read from the two basic banks is not equal to 01b.
2. When the EMP image read at initialization time does not belong to the 10 GbE controller.
3. When host debug mode is entered via a dedicated JTAG code (0x3E) or via setting of the DEBUG_EN pin, the host can remove address protection of the PF space and then clear the *GLNVM_FLA.LOCKED* bit. It is recommended that Flash programming platforms at manufacturing sites be provided with the JTAG and/or a strapping option as a back-up means.

This mode is not safe and must be used only at manufacturing time and/or as a last resort to recover from initial mis-configurations. Only host access to the Flash and shadow RAM is guaranteed when in this mode.

It is not recommended to enter this mode at run time because no resource ownership taking is required prior to accessing the NVM. Also, taking resource ownership requires an operational EMP, which is not the case when in this mode.

When the 10 GbE controller is in this mode (see steps 1 and 2 previously described), the EMP/PE codes are not loaded from the NVM and the EMP/PE remains disabled. The 10 GbE controller is not able to exchange any kind of traffic over the lines and no Admin command can be posted. The 10 GbE controller is in an unknown operational state where only the Flash programming flow is operational.

38.11.9 NVM Update Flows

The flows described in this section only affects normal programming mode. When in the blank Flash programming mode, refer to NVM access procedures described in Section 38.11.12.

It is assumed that an NVM update sequence does not last beyond the NVM ownership timeout for a write of three minutes. Otherwise, the sequence must be fragmented in several shorter sequences, releasing NVM ownership in between.

The software tool is responsible to re-compute and update the software checksum (word 0x3F) each time a change was made to words 0x00-0x3E further to NVM Update commands issued.

38.11.9.1 Flash High Level Map

Figure 38-20.Flash High Level Map Diagram

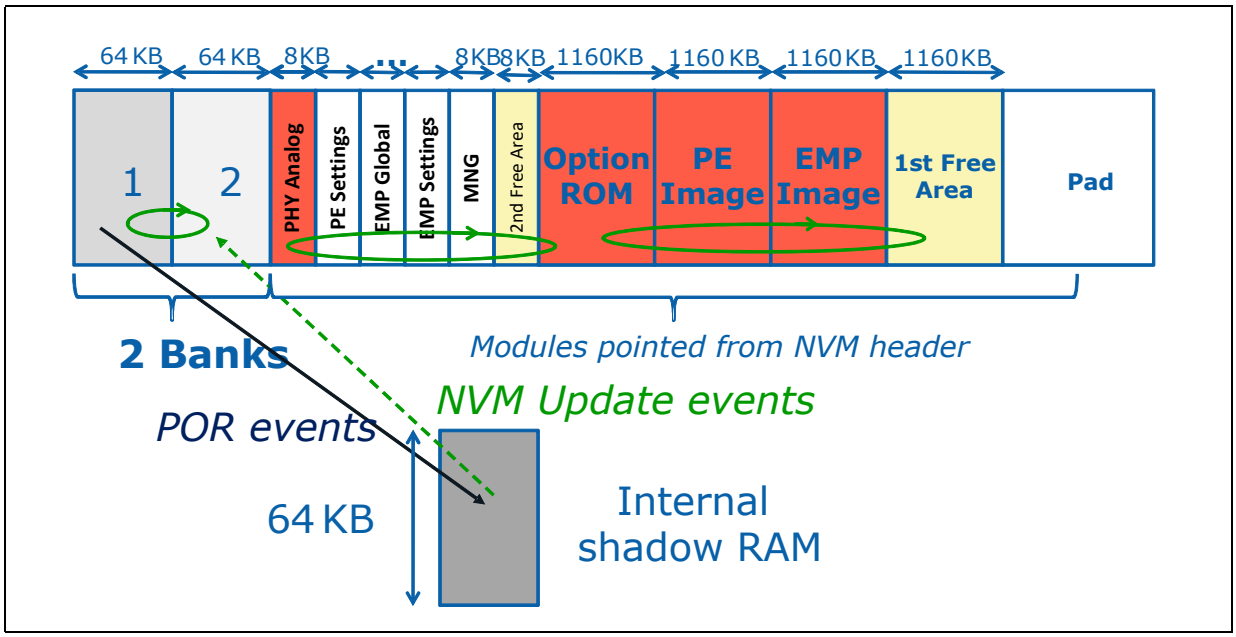


Figure 38-20 shows the high level mapping of the NVM in the 10 GbE controller. It is made of the following areas:

- **Two Basic Banks a, b** — Among other modules, it contains the VPD area. The current valid bank is mirrored into the internal shadow RAM at POR events. Changes that are made in the shadow RAM are finally dumped into the next bank, which becomes the current valid, and so forth cyclically. Refer to Section 38.11.9.2 and Section 38.11.9.3 for the update flows.
- **PE and EMP Image Areas** — Contains the firmware code to be run by PE and the EMP embedded processors. These are the only two modules which require authentication before an update. Refer to Section 38.11.9.5 for the update flow. The last 4 KB sector of the EMP image is covered by authentication but is not part of the compiled EMP code version. This sector contains commands to EMP for updating NVM RO words and modules (refer to Section 38.11.9.6). This last EMP image sector is referred as the RO commands section.
- **Expansion/Option ROM Area (OROM area)** — Contains pre-boot code and settings read by BIOS. Refer to Section 38.11.9.4 for the update flow. Pre-boot code is authenticated by BIOS at initialization time before being executed. Pre-boot code is also internally authenticated on update.
- **First free area** — The free provisioning area used for modifying one of the 1160 KB long areas such as PE or the EMP image areas. Once the new module content has been written into the first free area, the old module area is used as the new first free area, and so forth cyclically.
- **EMP Global, MNG, and EMP Settings Modules** — They contain information to be handled by the EMP. Refer to Section 38.11.9.4 for the update flow.
- **PHY Analog module**. This module is loaded by 10 GbE controller at power-up events only. Refer to Section 38.11.9.5 for the update flow.
- **Second Free Area** — The free provisioning area used for modifying one of the 8 KB long areas such as EMP settings or MNG areas. Once the new module content

has been written into the second free area, the old module area is used as the new second free area, and so forth cyclically.

Refer to Section 38-20 for the detailed NVM map.

38.11.9.2 VPD Update

38.11.9.2.1 First VPD Area Programming

The VPD capability is exposed on the PCIe interface only if the GLPCI_CAPCTRL.VPD_EN bit is set to 1b, regardless to any other sanity check that is performed on the VPD area contents.

The VPD area and VPD pointer must be written on a blank Flash and must contain a valid contents from this first programming. If the VPD *Write Enable* bit is set to 1b in NVM Security Control word (offset 0x02), the entire VPD area can be modified later on by a host via the NVM Update AQ command. If VPD tags were modified, it is required to issue a PCIe reset before write accessing the VPD area from PCIe configuration space.

VPD Area Update from PCIe Configuration Space

The flow described on this section is used once the VPD area contents and pointer have been already programmed in the NVM and loaded into the 10 GbE controller.

1. **A PF VPD Software Performs a VPD Write** — It sets write offset/data into VPD register set of the relevant PF configuration space, setting the VPD *Flag* (bit 15 in VPD Address register 0x0E2).
2. **Hardware Notifies the EMP** — It issues a internal VPD access interrupt to the EMP to notify it of the VPD access and of the PF affected.
3. **The EMP Checks that the VPD Write is Allowed** — It checks that the write offset points to the VPD-RW area (and not to a VPD RO area).
   a. If not, the EMP clears the VPD *Flag* in the PF configuration space to notify PF VPD software that the transaction completed and then exits the flow.
4. **EMP Writes the Change into Shadow RAM**.
5. **The EMP Completes the VPD Access to Software —** The EMP clears the VPD *Flag* in the PF configuration space to notify PF VPD software that the access completed.
6. **The EMP Takes Ownership Over the NVM Resources for a Write** — The EMP checks that the NVM is not busy and not owned by software. It then marks it internally to be owned by the EMP for a write operation, starting the 3-minute timer.
   - If the Flash is busy by a previous sector erase operation or if NVM ownership is held by software, it might that the flow needs to be restarted from step 1. at this stage by successive VPD write accesses initiated by VPD software. That way, successive VPD writes might generate only two next bank erase operations, one for the first and one for the last VPD write in a sequence, thus increasing Flash longevity.
7. **The EMP Erases the Next Bank** — It erases the contents of the next basic bank sectors, via the procedure described in Section 38.11.12.2.
8. **The EMP Copies Shadow RAM into the Next Bank** — It copies the shadow RAM into the next basic bank sectors with the exception of the *Validity* field, which is left as all ones.

9. **The EMP Checks the Flash Write** — The new bank content is read and checked to be identical to the shadow RAM contents. This can be done in the course of writing to the Flash using previous step.
   a. **If not (such as Flash defect), Exit the Flow**. The EMP releases the NVM ownership (internally).
   b. **If the Check was Successful, the EMP Validates the New Bank and Invalidates the Old Bank**.
      - The *Validity* field of the new bank is set to 01b. The EMP checks that the *Validity* field is read as written in the Flash. If not, then go to the previous sub-step.
      - The EMP toggles the state of the BANK1VAL bit in the GLNVM_GENS register to indicate that the non-valid bank became the valid one and vice versa.
      - The current (old) bank is invalidated by setting its *Validity* field to 00b.
      - The EMP releases the NVM ownership (internally).

Note: Users must be made aware that dumping the VPD change into the Flash might take a few hundreds of a ms after the VPD transaction completes to software (by clearing the VPD *Flag*). As a result, they must wait few seconds before they can shut down the system.

If the VPD write access is attempted by the host when the 10 GbE controller has just started a shadow RAM dump (step 8), then it might be that the write request times out.

38.11.9.3 Updating a RW Module or RW Words Mapped Inside Shadow RAM

Besides any RW word in the NVM header, the following NVM modules are affected by this flow:

- PCIR Registers Auto-load (pointed by NVM word 0x08).
- PBA Block (pointed by NVM word 0x16).
- Boot Configuration (pointed by NVM word 0x17).
- VPD Area (pointed by NVM word 0x2F) — if the VPD *Write Enable* bit is set to 1b.
- PXE Setup Options (pointed by NVM word 0x30).
- PXE Configuration Customization Options (pointed by NVM word 0x31).
- Software Alternate MAC Addresses (pointed by NVM word 0x37).
- POR Registers Auto-load (pointed by NVM word 0x38).
- Software RDMA MAC Address (pointed by NVM word 0x39)
- GLOBR Registers Auto-load (pointed by NVM word 0x3B).
- CORER Registers Auto-load (pointed by NVM word 0x3C).

The flow is as follow:

1. Software **takes ownership over the NVM resource for a write**.
2. **Software issues one or several NVM Update commands** — It posts the NVM Update admin command to the EMP with the *Authenticated* bit cleared, providing the following parameters (see Section 38.11.14.3):
   - Address in the NVM header of the pointer to the module or 0x0000 for updating a RW word of the NVM header
   - Offset inside the module
   - Buffer in host memory

- Buffer length (maximum supported is 4 KB)

If the *Last Command* bit is cleared in the command, it means that the command belongs to a complex NVM update operation made of several elementary NVM update commands that are posted in the admin queue. In between completions of elementary commands in a chain, other commands can be posted by a PF, besides other NVM-related commands. The entire NVM change is committed to the Flash part only once the last NVM command of the sequence is processed.

It is assumed that each command in the sequence leaves the shadow RAM with a usable/consistent contents as needed if a device-level reset occurs in the middle of the NVM update sequence.

The Flush on Error (FE) bit must be set for all the commands of a sequence, with the exception of the last one.

3. **The EMP checks that the command is valid and posts a response to software** — It performs the following command validity checks and posts a response (ACK/NACK) to software. If one check fails then the EMP flushes the remaining NVM update commands (if any) of the sequence and exits the flow. Otherwise, if no error is encountered, the EMP schedules the NVM command to run in a separate thread, resuming from the next step.
   a. The pointer location is not a RO module or word. The size of a RO module mapped to the shadow RAM is given at the first word of the module.
   b. The start/end offsets, once applied to the module's location in the basic bank, do not lead to addresses beyond the shadow RAM size. The start/end offsets, once applied to the module's location in the basic bank, do not overwrite contents of a RO module.
   c. If the pointer location is 0x0000, it means that the NVM update operation might affect fields in the NVM header. In that case, it must be checked that the start/end offsets does not lead to write over RO words.
4. **The EMP writes the change into shadow RAM**.
5. If the pointer is to the PFA (0x40), the EMP parses the modified PFA contents and performs the required configuration changes throughout the internal SR according to the meta-data section; otherwise, only the specific section within the PFA is updated.
   a. If items loaded at EMPR only (such as MNG filter module) were modified, firmware updates them into the device RAM or registers.
6. **If the *Last Command* bit is set in the command** — The EMP commits the change into the next basic shadow RAM bank (a or b) by performing the following steps. Otherwise, it completes the command to software, exits the flow and goes back to step 2. when the next command is posted.
7. **The EMP erases next bank** — It erases the contents of the next basic bank sectors, via the procedure described in Section 38.11.12.2.
8. **The EMP copies shadow RAM into next bank** — It copies the shadow RAM into the next basic bank sectors with the exception of the *Validity* field, which is left as all ones.
9. **The EMP checks the Flash write** — The new bank content is read and checked to be identical to the shadow RAM contents. This can be done in the course of writing to the Flash using a previous step.
   a. **If not (such as Flash defect), return an error status —** The command completes to software (in the next step) with the Flash defect error bit set.
   b. If the check was successful, the EMP validates the new bank and invalidates the old bank.

— The *Validity* field of the new bank is set to 01b. The EMP checks that the *Validity* field is read as written in the Flash. If not, then go to the previous sub-step.
— The EMP toggles the state of the BANK1VAL bit in the GLNVM_GENS register to indicate that the non-valid bank became the valid one and vice versa.
— The current (old) bank is invalidated by setting its *Validity* field to 00b.

10. **The EMP posts an event completion on ARQ to software**.
    a. **If the NVM ownership timeout for write ends before reaching this step**, the EMP flushes the remaining NVM update commands (if any) of the sequence, reporting a timeout error status.
11. **Software releases NVM ownership**.
12. **Software initiates a system reset (PERST)** for loading the modifications into the 10 GbE controller.

Note: In order to allow adding a new shadow RAM RW module in the future without requiring a new EMP image be loaded before, the list of concerned modules is not exhaustive, and therefore, the EMP must not check that the module(s) to be updated belongs to the list.

The flow used is the same as that in the previous section, with the additional step that after step 5., firmware updates the immediate fields in the PFA to reflect the change.

38.11.9.4 Reprogramming a Non-Authenticated Module Mapped Outside Shadow RAM

The following flow only supports the full module replacement via a double-bank policy. The following NVM modules are affected by this flow:

1. EMP Global (pointed by NVM word 0x09)
2. Manageability (pointed by NVM word 0x0E)
3. PE Settings (pointed by NVM word 0x3D)
4. EMP Settings (pointed by NVM word 0x0F)
5. Next bank of extension TLVs (pointed by an extension TLV in the PFA module)

Note: It is guaranteed that the contents of the last three modules in the list cannot alter the programming path from the host to the NVM. This means that the EMP cannot break further to a mis-configuration of these modules.

Modules items (steps 1. through 4.) use the second free area pointer (NVM word 0x46) for their bank swap. This free area is made up of two consecutive
4 KB sectors and it is used for the update of one module at a time.

1. Software takes ownership over the NVM resource for a write.
2. Software issues one or several NVM Erase and/or Update commands — It posts an NVM Erase/Update Admin command to the EMP, providing the following parameters (see Section 38.11.14.3):
    - Address in the NVM of the pointer to one of the modules previously listed

    It is software's responsibility to erase a Flash sector prior to writing it.

    If the *Last* command bit is cleared in the command, it means that the command belongs to a complex NVM update operation made up of several elementary NVM Update and/or Erase commands that are posted in the admin queue. In between completions of elementary commands in a chain, other commands can be posted by a PF, besides other NVM-related commands. The pointers are updated in the Flash part only once the last NVM command of the sequence is processed.

The *Flush on Error* (FE) bit must be set for all the commands of a sequence with the exception of the last one.

3. The EMP checks that the command is valid and posts a response to software — It performs the following command validity checks and posts a response (ACK/NACK) to software. If one check fails then the EMP flushes the remaining NVM update commands (if any) of the sequence and exits the flow. Otherwise, if no error is encountered, the EMP schedules the NVM command to run in a separate thread, resuming from the next step.
   a. The pointer location belongs to one of the modules in the list.
   b. The start/end offsets, once applied to the relevant free area, do not lead to addresses beyond the free area devices NVM area size.
   c. The start/end offsets, once applied to the relevant free area, do not spread over two consecutive 4 KB sectors.
4. The EMP erases/writes the free provisioning area — According to the command, the EMP erases the 4 KB sector(s) in the free provisioning area via the procedure previously described in Section 38.11.12.2 or writes the change required by the (elementary) command into it.
5. If the update sequence did not end, the EMP posts an event completion on ARQ to software — If the *Last Command* bit is cleared in the (elementary) command, the EMP completes the command and then goes to back to step 3. of this flow for the processing of the next (elementary) command of the sequence.
6. If the update sequence ended, the EMP swaps pointers — If the *Last Command* bit is set in the command, the EMP swaps between the module's pointer and the free provisioning area pointers in the shadow RAM NVM header.
7. The EMP erases the next bank — It erases the contents of the next basic bank sectors, via the procedure described in Section 38.11.12.2.
8. The EMP copies shadow RAM into the next bank — It copies the shadow RAM into the next basic bank sectors with the exception of the *Validity* field, which is left to all ones.
9. The EMP checks the Flash write — New bank content is read and checked to be identical to the shadow RAM contents. This can be done in the course of writing to the Flash using a previous step.
   a. If it is not (like Flash defect), return an error status — The EMP swaps back the two pointers in shadow RAM and the command completes to software (in the next step) with the Flash defect error bit set.
      - If the check was successful, The EMP validates the new bank and invalidates the old bank. The *Validity* field of the new bank is set to 01b. The EMP checks the *Validity* field is read as written in the Flash. If not, it goes to the previous sub-step.
      - The EMP toggles the state of the BANK1VAL bit in the GLNVM_GENS register to indicate that the non-valid bank became the valid one and vice versa.
      - The current (old) bank is in-validated by setting its *Validity* field to 00b.
10. The EMP posts an event completion on ARQ to software.
    a. If the NVM ownership timeout for write ends before reaching this step, the EMP flushes the remaining NVM update commands (if any) of the sequence, reporting a time out error status.
11. Software releases NVM ownership.

12. Software initiates a device reset (PCIR, CORER, or GLOBR) for loading the modifications into the 10 GbE controller.
13. If the module modified is the manageability or the EMP settings module, then the EMP resets itself to load the new settings into the 10 GbE controller.

Note: The contents of the old module cannot be erased by a software command prior to completing this flow.

Figure 38-21.Initial State Before the NVM Update Command

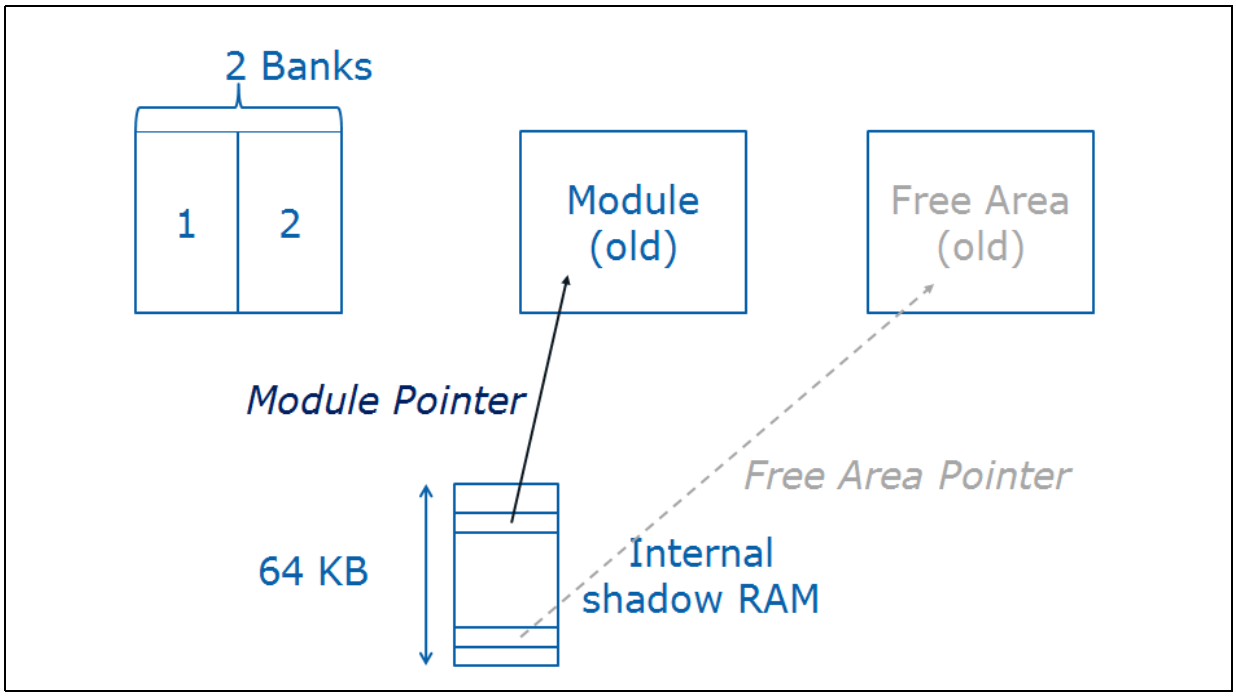
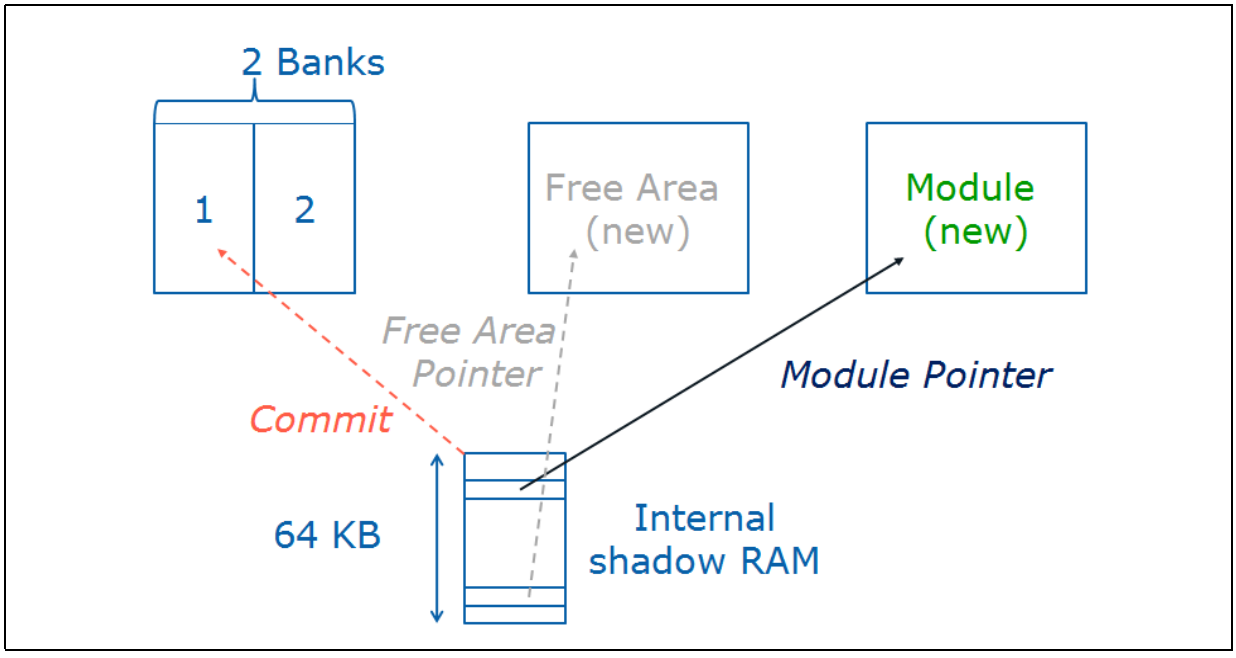


Figure 38-22.Final State After Completion of the NVM Update Command



38.11.9.5 Reprogramming An Authenticated Module Mapped Outside Shadow RAM

The following flow only supports the full module replacement via a double-bank policy. The following NVM modules are affected by this flow:

1. Option ROM (pointed by NVM word 0x05)
2. EMP Image (pointed by NVM word 0x0B)
3. The first free area pointer (NVM word 0x40) is used for the bank swap modules. This free area must be sized exactly to 1160 KB. Software then takes ownership over the NVM resource for a write.
4. Software issues one or several NVM Erase and/or Update commands — It posts the NVM Erase/Update admin command to the EMP, providing the following parameters (refer to Section 38.11.14.3):
   - Address in the NVM of the pointer to one of the modules previously listed
   - Offset inside the module
   - Buffer in host memory
   - Buffer length (maximum supported is 4 KB)

   It is software's responsibility to erase a Flash sector prior to writing it.

   If the *Last Command* bit is cleared in the command, it means that the command belongs to a complex NVM update operation made up of several elementary NVM update and/or erase commands that are posted in the admin queue. In between completions of elementary commands in a chain, other commands can be posted by a PF, besides other NVM-related commands. The entire 1160 KB free provisioning area must always be written.

   It is software's responsibility to issue the required NVM Erase commands prior to issuing NVM Update commands.

   The *Flush on Error* (FE) bit must be set for all commands of a sequence with the exception of the last one.
5. The EMP checks that the command is valid and posts a response to software — It performs the following command validity checks and posts a response (ACK/NACK) to software. If one check fails then the EMP flushes the remaining NVM Update commands (if any) of the sequence and then exits the flow. Otherwise, if no error is encountered, the EMP schedules the NVM command to run in a separate thread, resuming from the next step.
   a. The pointer location belongs to one of the modules in the list.
   b. The start/end offsets, once applied to the relevant free area, do not lead to addresses beyond the free area size.
   c. The start/end offsets, once applied to the relevant free area, do not spread over two consecutive 4 KB sectors.
6. The EMP erases/writes the free provisioning area — According to the command, the EMP erases the 4 KB sector in the free provisioning area via the procedure described in Section 38.11.12.2 or writes the change required by the (elementary) command into it.
7. If the update sequence did not end, the EMP posts an event completion on ARQ to software — If the *Last Command* bit is cleared in the (elementary) command, the EMP completes the command and then goes back to step 3. of this flow for the processing of the next (elementary) command of the sequence.
8. The EMP checks the rollback revision — It checks that the rollback revision (lad_srev field in CSS header) of the new module is greater or equal to the rollback revision of the current module.

a. If the check fails, the command is completed to software with the Rollback Revision Check Fails bit set. The EMP goes to step 15.

9. The EMP checks the 10GbE controller *Blank NVM Device ID* and *Module ID* fields — it checks that the the 10 GbE controller *Blank NVM Device ID* field written in the new module is 0x154B. The EMP checks that the 30 LS bits of the *lad_module_id* field in the CSS header correspond to the module currently being updated.
   a. If one of the checks fails, the command is completed to software with the EACCES bit set. The EMP goes to step 15.
   b. Note that before authenticating the PCIe analog, PHY analog module, or option ROM module, the module must be appended with the last 330 words of the module's area, as the CSS header has been moved to the module's trailer.
10. If an update sequence ended, the EMP checks CRC8 and authenticates the module — It reads the new module from the Flash, makes sure its CRC8 is valid and then authenticates its signature according to the procedure described in Section 38.11.13. This can be done in the course of writing to the Flash using the previous steps.
    a. If one of the checks fail, the command is completed to software with the relevant error bit set, either the Public Key Check Fails bit or the Module Signature Check Fails bit. The EMP goes to step 15.
11. The EMP swaps pointers — The EMP swaps between the module's pointer and the free provisioning area pointers in the NVM header of the shadow RAM to point to the next module's bank.
12. The EMP erases next bank — It erases the contents of the next basic bank sectors, via the procedure described in Section 38.11.12.2.
13. The EMP copies shadow RAM into next bank — It copies the shadow RAM into the next bank sectors with the exception of the *Validity* field, which is left as all ones.
14. The EMP checks the Flash write — The new bank content is read and checked to be identical to the shadow RAM contents. This can be done in the course of writing to the Flash at the previous step.
    a. If not (such as Flash defect), return an error status - The EMP swaps back the two pointers in the shadow RAM and the command is completed to software (in the next step) with the Flash defect error bit set.
    b. If the check completed successfully, the EMP validates the new SR bank and invalidates the old SR bank
       - *Validity* field of the new bank is set to 01b. The EMP checks the *Validity* field is read as written into the Flash. If not, it goes to the previous sub-step.
       - EMP toggles the state of the BANK1VAL bit in the GLNVM_GENS register to indicate that the non-valid bank became the valid one and vice versa.
       - The current (old) bank is invalidated by setting its validity field to 00b.
15. **If the module updated was the PE image**, the next CORER event has the effect of reloading the PE image from the Flash.
16. If the module updated was the EMP image and if the last command in the flow completed with an error, then go to next step to report the error.
17. The EMP posts an event completion on ARQ to software — It posts a completion/response to the last Admin command of the sequence.
18. If the module version is stored VPD, software might change the VPD module, which is mapped into PFA, to match the updated version loaded in the Flash.

Note: Software tools are responsible for updating the checksum field's value accordingly.

19. Software releases NVM ownership.
20. If the module updated was the EMP image, the EMP resets itself and reloads from the *new* EMP image.

Note: Contents of the old EMP/PE image cannot be erased by a software command prior to completing this flow.

38.11.9.6 Update Link Mode

This section describes the steps required to change link mode.

Software should follow the following sequence:

1. Take NVM ownership for write (using the Request Resource Ownership AQC).
2. Issue an NVM Config Read Admin command for feature 0xFFF1 (See the note that follows).
3. Compare the read selection with the desired one. If the two are identical, do nothing and exit the flow (See the note that follows).
4. Issue an NVM Config Write Admin command while using one of the command buffers described as follows (according to the desired link mode transition).
   a. Transition to 4x10 GbE: Feature_ID = 0xFFF1; Feature Options = 0x0; Feature selection = 0x0002.
5. Send NVM_Update AQC to trigger a Shadow RAM dump.
6. Wait for the NVM_Update Completion Event.
7. Release NVM ownership (using the Release Shared Resource AQC).
8. Perform PERST reset to apply the new NVM configuration. PERST triggers GLOBR internally.

Note: Steps 2 and 3 are optional.

38.11.10 NVM Clients and Low Level Interfaces

There are several clients that can access the NVM to different address ranges via different access modes, methods, and low-level interfaces. The various clients to the NVM are hardware, software tools (BIOS, etc.), drivers, EMP, PE, BMC (via EMP), and VPD software.

Table 38-70 lists the different accesses to the NVM.

Table 38-70. Clients And Access Types to the NVM (Sheet 1 of 2)

Client	NVM Access Method	Accessed Performed Against	Logical Byte Address Range	NVM Access Interface (CSRs or Other)
VPD Software	Parallel (32-bits)	Shadow RAM	0x000000 - 0x0003FF from VPD module beginning	VPD Address and Data registers. Any write access is immediately pushed by the 10 GbE controller into the Flash.

Table 38-70. Clients And Access Types to the NVM (Sheet 2 of 2)

Client	NVM Access Method	Accessed Performed Against	Logical Byte Address Range	NVM Access Interface (CSRs or Other)
PF Software	Parallel via Memory (CSR)BAR (32-bits read)	Flash Part	0x000000 - 0xFFFFFF	The address is relative to the beginning of the flash.
	Parallel via Expansion ROM BAR (32 bits read only).	Flash Part	0x020000 - 0xFFFFFF	This logical address range is relative to the beginning of the expansion ROM module.
	Via AQC	Flash part/ Shadow RAM	0x000000 - 0x00FFFF	NVM Read, NVM Erase and NVM Update admin commands as described in Section 38.11.14.

38.11.10.1 Memory-Mapped Host Interface

The Flash is read by the 10 GbE controller each time the host CPU performs a read operation to a on that is within the Flash address mapping or upon boot via accesses in the space indicated by the Expansion ROM Base Address register. Accesses to the Flash are based on a direct decode of CPU accesses to a memory window defined in either:

- Memory CSR + Flash Base Address register (PCIe Control register at offset 0x10). The Flash address space is exposed to the host memory BAR when the Flash *Expose* bit is set in the NVM (or the *GLPCI_LBARCTRL.FLASH_EXPOSE CSR* bit is set) or when the 10 GbE controller is in blank Flash programming mode. The Flash size exposed is retrieved from the *GLPCI_LBARCTRL.FL_SIZE* CSR field, and is 8 MB by default (blank Flash programming mode).
- The Expansion ROM Base Address register (PCIe Control register at offset 0x30). The module address space is always exposed to the expansion ROM BAR unless the Flash is blank. The Flash size exposed is retrieved from the *GLPCI_LBARCTRL.EXROM_SIZE* CSR field, and is 512 KB by default. The 10 GbE controller is responsible to map read accesses via the expansion ROM BAR to the physical NVM. Write attempts to the Flash through this BAR are not performed, they are silently dropped. The offset in the NVM of the expansion ROM module is defined by the PCIe expansion/option ROM pointer (Flash word address 0x05). This pointer is loaded by the 10 GbE controller from the Flash before enabling any access to the expansion ROM memory space.
  - When modifying the PXE driver section pointer in the NVM, it is required to issue a PCIe reset on which the updated offset is sampled by hardware.
  - If there is no valid NVM validity field in the two basic banks, then the expansion ROM BAR is disabled.

The 10 GbE controller controls accesses to the Flash when it decodes a valid access. Out of range memory-mapped read access returns arbitrary data.

Notes:

Flash read accesses are assembled by the 10 GbE controller each time the access is greater than a byte-wide access.

The 10 GbE controller byte reads to the Flash take about 2 to 30 ms. The 10 GbE controller continues to issue retry accesses during this time.

During normal operation, the host should avoid memory-mapped accesses to the first two basic banks of the Flash because it might be non-coherent with the shadow RAM contents.

Prior to initiating an NVM read cycle via memory mapped access, PF software is required to take ownership over the NVM resources.

38.11.11 Flash Access Contention

Flash read accesses initiated through PFs might occur concurrently to the EMP modifying the NVM contents. The 10 GbE controller does not synchronize between the different entities accessing the Flash so contentions caused from one entity reading and the other modifying the same locations is possible.

To avoid such a contention between software and EMP accesses, these entities are required to make use of the NVM ownership taking/release flows for any read or write access to NVM. This is also useful to avoid the timeout of the PCIe transaction made to a memory mapped Flash address while the Flash is currently busy with a long sector erase operation.

However, two software entities cannot use the NVM ownership acquiring/release mechanisms: BIOS and VPD software.

- Since VPD software accesses only the VPD module, which is located in the first valid bank of the NVM, VPD accesses are always performed against the shadow RAM first. In this case, the EMP must take/release ownership over the NVM as if it was the originator of the Flash access. It is then hardware/EMP's responsibility to update the NVM according to the Flash update sequence described in Section 38.11.9.2.
- No contention can occur between BIOS and any other software entity (VPD included) as it accesses the NVM while the operating system is down.

However, since BIOS cannot take ownership over the NVM resource, it might be that the Flash part is not accessible when BIOS attempts reading it. This might occur if a Flash erase operation was performed just before PCIe reset. In such a case, read accesses via the expansion ROM BAR returns 0xDEADBEEF.

- It is assumed that the expansion ROM signature check performed by BIOS fails in this case.
- The EMP must avoid initiating sector erase operations at boot time.
- It is assumed and recommended that users do not attempt to update the NVM contents via the BMC while the system is re-booting.
- The BMC should delay PERST# de-assertion or boot running until after the BMC completed any OOB accesses to Flash memory. It is required to route the wake-up signal from the standby button to the BMC and not to the chipset. The BMC issues a system reboot signal to the chipset only after any NVM write access completes.
- If a system reboot is issued by a local user running on the host, there is no technical way to avoid contention in this case.

Note: It is the user's responsibility when accessing the NVM remotely via the BMC to make sure another user in not currently initiating a local host reboot there.

- The EMP is responsible to take NVM ownership on the BMC account prior to performing any NVM read or write access, which is needed for handling an NC-SI command. The NVM ownership is released by the EMP together with completing the NC-SI command. If NVM ownership is not free when processing the NC-SI command, the command completes with a package not ready status.

38.11.12 NVM Access Procedures

Any software read/write or EMP write flow described in this section (except flows executed by VPD software or by BIOS or to flows executed when in blank Flash programming mode) must be preceded by taking NVM ownership. Anytime software is taking NVM ownership, it must re-read the pointers to the module it plans to access because they might have been modified by the EMP in between two ownership takings.

Refer to section that describes NVM ownership taking/releasing procedures as well for the associated timeouts.

38.11.12.1 Flash Erase Flow By The Host

This flow is available to the software PF device driver only when the 10 GbE controller is in blank Flash programming mode.

1. Poll the *MNGSB_MSGCTL.CMDV* bit and make sure it is cleared.
2. Program 10 GbE controller registers for an Erase Message:
   a. *MNGSB_WHDR0.OPCODE_SEL* = Erase (4).
   b. *MNGSB_WHDR0.DEST_SEL* = SPI (1).
   c. *MNGSB_WHDR0.TAG* = 0.
   d. *MNGSB_WHDR1.ADDR* = Byte address of the erased sector, aligned to 4 KB.
   e. *MNGSB_WHDR2.LENGTH* = 0x10 (4 KB).
3. Trigger the Flash Erase transaction.
   a. *MNGSB_MSGCTL.MSG MODE* = Normal (0b).
   b. *MNGSB_MSGCTL.TOKEN MODE* = Normal (0b).
   c. *MNGSB_MSGCTL.HDR_DWS* = Header DWords should always be always 3 for SPI transactions.
   d. *MNGSB_MSGCTL.EXP_RDW* = 0b (no data is expected in return).
   e. *MNGSB_MSGCTL.BARCLR* = 1b (must be 1b for Erase/Write).
   f. *MNGSB_MSGCTL.CMDV* = 1b.
4. Poll the *MNGSB_MDGCTL.CMDV* until it is 0b.
5. Read *MNGSB_RSPCTL.ERR.*
   a. If ERR = 0b then transaction completed successfully.
   b. If ERR = 1b then transaction failed.
6. Steps 2 through 5 can be repeated if needed.

38.11.12.2 VPD Accesses

The VPD module (VPD area) is mapped into the valid basic bank and it is thus mirrored in shadow RAM. It is accessed by VPD software via the PCIe VPD capability structure. Refer to Section 38.11.9.2 for more details.

38.11.13 NVM Authentication Procedure

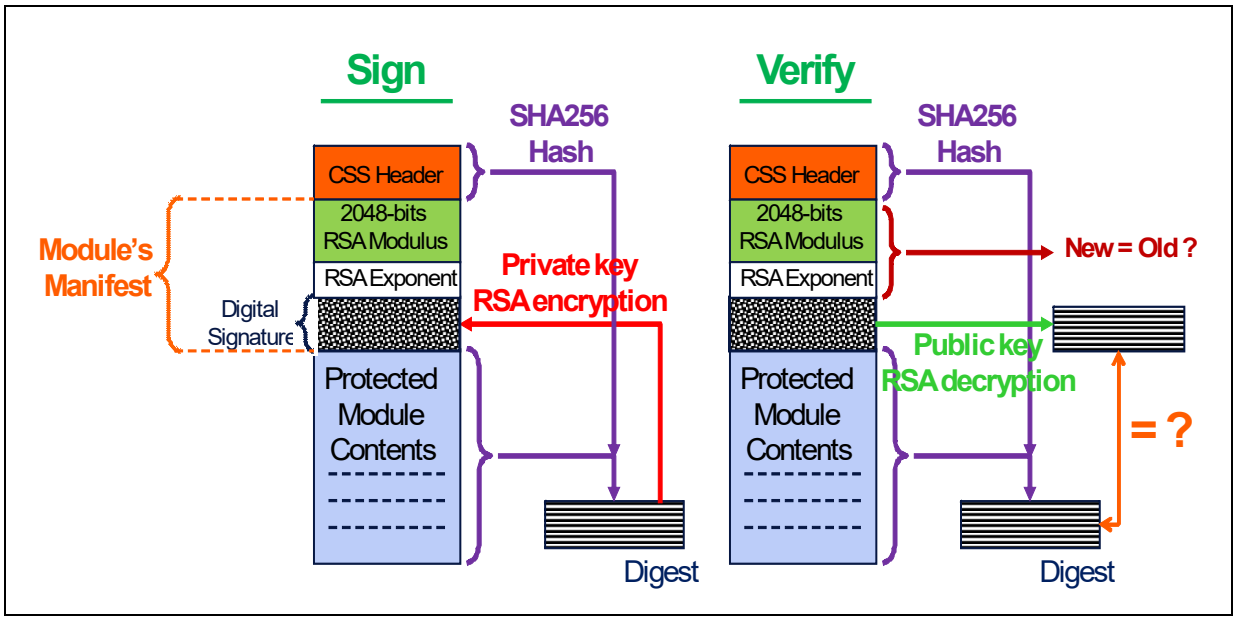
The NVM update integrity feature ensures that only Intel-approved firmware code (or another protected NVM module) is able to be updated on 10 GbE controller devices after manufacturing. This procedure is performed by a RAM-based firmware each time an attempt is made to update one of the protected modules. Refer to NVM update flows in Section 38.11.9 for more details.

Integrity validation of NVM updates is provided by means of a digital signature. The digital signature is a SHA256 hash computed over the protected content (256 bits long) that is then encrypted by a 2048-bits RSA encryption using an Intel private key. This digital signature is stored in what is called the manifest in the NVM module image. Also stored in the manifest is the corresponding RSA modulus (public key) and RSA exponent to be used to decrypt the digital signature.

To verify the authenticity of the digital signature, EMP must first verify that the RSA *Modulus* and RSA *Exponent* fields in the new module loaded are identical to those in the current module. If the RSA *Modulus* and *Exponent* fields are the same, EMP decrypts the digital signature using the 2048-bit RSA *Modulus* and *Exponent* fields stored in the manifest of the old module to extract the expected SHA256 hash of content (stored hash). EMP then performs an independent SHA256 hash over the protected content (computed hash). If the stored hash matches the computed hash, the digital signature is accepted and the NVM module update is applied.

NVM updates are validated prior to invalidating the old NVM configuration, such that the old NVM configuration is still usable if the update fails to validate. After the new NVM is successfully verified, the updated image is committed to the 10 GbE controller's Flash by the EMP.

Figure 38-23.Sign and Verify Procedures For Authenticated NVM Modules



38.11.13.1 Digital Signature Algorithm Details

As previously mentioned, the digital signature generation is a hash computation followed by an RSA encryption. This is performed within Intel as part of the NVM update image generation process and not performed by Intel software in the field, nor by the 10 GbE controller.

The algorithms used are described in the following locations:

- PKCS #1 v2.1: RSA Cryptography Standard, RSA Laboratories, June 14, 2002 - www.rsa.com
- SHA family definition - http://csrc.nist.gov/publications/fips/fips180-3/fips180-3_final.pdf
- SHA usage with digital signatures - http://csrc.nist.gov/publications/nistpubs/800-107/NIST-SP-800-107.pdf
- SHA validation vectors — http://csrc.nist.gov/groups/STM/cavp/documents/shs/SHAVS.pdf

Note: The protected module contents shown in Figure 38-23 starts with the 10 GbE controller's Blank NVM Device ID word of the NVM header and ends with the last word of the 1160 KB long EMP/PE image area, regardless of the size of the EMP/PE code and to the presence and size of a RO commands section at the last sector of the EMP image area.

38.11.13.2 Intel Key Generation And Intel Code Signing System

The integrity of NVM digital signatures requires not only robust private RSA key generation but also continued protection of these private keys into the indefinite future as well as a method to generate new signed images using the existing private keys.

Note: Not all NVM updates increment the Rollback Revision field. Rollback is allowed between supported versions and NVM configuration versions where the rollback version is the same.

38.11.13.3 Protected Modules

Any data that is modified in the field (either by the OEM during manufacturing or by the end user) cannot be included in the signed region of the NVM. The 10 GbE controller cannot generate a signed image by itself because the private key is not available to it to generate the digital signature in the NVM.

Only the following NVM modules require authentication in the 10 GbE controller. Each module is appended by its own digital signature:

- Option ROM
- NVM bank

38.11.13.4 Software Requirements

A software tool must prepare NVM images for the CSS signing step, pre-pending the CSS manifest, applying an Intel rollback revision field. After receiving the signed image, the tool merges the excluded fields back into the NVM image and performs an internal integrity check to verify that the merge was successful (such as a software computation of the digital signature passes).

SVTools (LANconf) MUST implement an NVM update image integrity check option in software prior to applying NVM updates to hardware. SVTools might integrate capabilities to generate self-signed NVM images to assist in the SV and debug process by developers.

CELO (or equivalent manufacturing diagnostics tool) must implement a test to check the GLNVM_FLA *Locked* bit state as part of manufacturing qualification. If after the NVM is programmed, and the 10 GbE controller powers up with the GLNVM_FLA *Locked* Bit not set, an inappropriate NVM image has been loaded during manufacturing (an NVM image with incorrect Flash opcodes). This represents a critical error. With GLNVM_FLA unlocked, unauthorized in-the-field updates can bypass designed firmware integrity checks.

Host software device drivers might implement an interface enabling a network administrator to perform an internal verification check of the signed NVM image. Using Windows drivers, this would take the form of an OID, which reports a SUCCESS or INVALID_PARAMETER. Using Linux, an ethtool command extension is advised to enable command line interrogation of the NVM content using the hash value build into the hardware as well as the saved CSS manifest in the NVM image.

38.11.13.5 Manufacturing Requirement

38.11.13.5.1 End-of-line Verification

OEM's must execute CELO (or an equivalent manufacturing diagnostics tool) to verify the GLNVM_FLA *Locked* bit state (as previously described).

38.11.13.5.2 Post-manufacturing Physical Modification Countermeasures

For add-in NICs and related form factors, applying a conformal epoxy encapsulation coating around the edges of NVM components (Flash and EEPROM) during manufacturing prevents them from being easily removed.

In addition, epoxy is used to coat any traces or vias that could enable direct-probing of the 10 GbE controller. The testing requirement for these coatings must be that they cannot be removed using a standard hobby knife or other hand-held tool.

Note that protection by epoxy encapsulation is just a recommendation, not a security requirement for the 10 GbE controller.

38.11.14 NVM Access Admin Commands and Events

NVM access commands are not supported when the 10 GbE controller is in the blank Flash programming mode. They are available only to the PFs once it has acquired NVM ownership.

Table 38-71. NVM Access Admin Commands And Events

Command/Event	Opcode	Brief Description	Detailed Description
NVM Read	0x0701	Read a segment from the NVM into a host buffer.	Section 38.11.14.1
NVM Erase	0x0702	Erase consecutive 4 KB sectors of the Flash.	Section 38.11.14.2
NVM Update	0x0703	Write a segment of the NVM from a host buffer.	Section 38.11.14.3
NVM Config Read	0x0704	Read Feature Selections.	Section 38.11.14.4
NVM Config Write	0x0705	Program Feature Selections.	Section 38.11.14.5

Note: All parameters in the Admin commands are defined in little endian.

38.11.14.1 NVM Read

This command is useful especially if the host memory-mapped access to the NVM was not enabled with the intent to save memory address space.

Table 38-72. NVM Read Admin Command

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0701	Command opcode.
Datalen	4-5		Length in bytes of command buffer.
Return Value/ VFID	6-7		Must be zeroed by the software device driver.
Cookie High	8-11	Cookie	Opaque value, that is copied by the EMP into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, that is copied by the EMP into the completion of this command.
Command Flags	16		NVM Access Admin Command Parameters bit 0 = Last command bit, used to notify EMP that this is the last admin command of a sequence. bits 1:6 = Reserved, must be zeroed. bit 7 = Flash-only bit. When bit 7 is set, the read is done directly from the flash and not from shadow RAM. Relevant only if access address is below 64 KB. Ignored otherwise.
Module_pointer	17		Module pointer location in words from the NVM beginning. A value of 0x0000 means that the command is performed over the Flash part seen as a flat memory.
Length	18-19		Length of the section to be read, which is expressed in bytes from the offset in the module. A value of 0xFFFF means the last byte to be returned is the last byte of the module (if byte 17 was not set to 0x0000). In any case, a (single) read command is limited up to 4 KB.
Offset	20-23		Byte 23 = Reserved, must be zeroed. Bytes 22:20 = Offset in the module, which is expressed in bytes from the pointed module's beginning. This is the byte offset of the first byte returned in the data buffer.
Data Address High	24-27		Address of command buffer.
Data Address Low	28-31		

Table 38-73. NVM Read Response (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0701	Command opcode.
Datalen	4-5		Length in bytes of command buffer.
Return Value	6-7		Return Value 0x0 = No error (success). EPERM = The module pointer location specified in the command does not permit the required operation. The word contents is not a pointer. EINVAL = Out of range offset/length (beyond the module's size). EIO = Flash defect. EBUSY = The PF is not permitted to post this command because it does not own the NVM resource. This error code is also returned if the PF attempts to post a command while another NVM command is in process.
Cookie High	8-11	Cookie	Opaque value that is copied by the EMP into the completion of this command.

Table 38-73. NVM Read Response (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Cookie Low	12-15	Cookie	Opaque value that is copied by the EMP into the completion of this command.
Reserved	16		Reserved.
Module_pointer	17		Module pointer location and copied from the command.
Length	18-19		Length to be read and copied from the command. If a value of 0xFFFF was set in the admin command (and if byte 17 was not set to 0x0000), this field returns the length from the offset to the module's end.
Offset	20-23		Byte 23= Reserved, must be zeroed. Bytes 22:20 = Offset in the module, copied from the command.
Data Address High	24-27		Address of command buffer.
Data Address Low	28-31		

38.11.14.2 NVM Erase

This command is used to erase the contents of 4 KB Flash sectors.

Table 38-74. NVM Erase Admin Command

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0702	Command opcode.
Datalen	4-5		Must be zeroed by the software device driver.
Return Value/ VFID	6-7		Must be zeroed by the software device driver.
Cookie High	8-11	Cookie	Opaque value that is copied by the EMP into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value that is copied by the EMP into the completion of this command.
Command_flags	16		NVM Access Admin Command Parameters bit 0 = Last command bit used to notify the EMP that this is the last Admin command of a sequence. bits 7:1 = Reserved, must be zeroed.
Module_pointer	17		Module pointer location in words from the NVM beginning. A value of 0x0000 means that the command is performed over the Flash part seen as a flat memory. Attempting to erase the basic banks or RO modules area is not allowed in normal operating mode. Attempting to erase the valid shadow RAM bank (SRa or SRb) or the valid NVM or OROM banks is rejected in normal operating mode. Besides 0x0000, only the address of a free provisioning area module pointer can be listed here. The next NVM bank can be erased by putting word address 0x42 in this field and the next OROM bank by work address 0x44,
Length	18-19		Length of the section to be erased, which is expressed in 4 KB sector units from the offset in the module. The module's beginning must be aligned to a 4 KB sector for the command to be valid. A value of 0xFFFF means that the last 4 KB sector to be erased is the last sector of the free provisioning area (if byte 17 was not set to 0x0000).
Offset	20-23		Byte 23 = Reserved, must be zeroed. Bytes 22: 20 = Offset in the module, which is expressed 4 KB sector index from the pointed module's beginning.
Reserved	24-27		Reserved.
Reserved	28-31		

This command is an asynchronous command. The EMP reads the command from the ATQ and writes back an immediate completion, intended only as an ACK/NACK that the command has been addressed by the EMP. The EMP checks the validity of the command and returns an error (NACK) on the ATQ completion if it is unable to process the command. If successful (ACK), the EMP then schedules the NVM erase operation to be performed by a lower priority thread, which can be preempted by other AQ commands. Once completed, the EMP posts a completion event on the ARQ. Software must hold the NVM resource lock while performing this operation and must release it once NVM operations complete. Software must not post another NVM command while this command is in process. For example, during the time between posting the request on the ATQ and receiving the completion event on the ARQ.

Table 38-75. NVM Erase Response

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0702	Command opcode.
Datalen	4-5		Reserved.
Return Value	6-7		Return Value 0x0 = No error (success). EPERM = The module pointer location specified in the command does not permit the required operation. The word contents is not a pointer to a free provisioning area or attempt to erase sectors from the basic banks. EINVAL = Out of range offset/length beyond the free area module's limits. ENOENT = The module pointed is not aligned with an 4 KB sector beginning. EBUSY = The PF is not permitted to post this command because it does not own the NVM resource. This error code is also returned if the PF attempts to post a command while another NVM command is in process.
Cookie High	8-11	Cookie	Opaque value that is copied by the EMP into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value that is copied by the EMP into the completion of this command.
Reserved	16-31		Reserved.

Table 38-76. NVM Erase Completion (on ARQ) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0702	Command opcode.
Datalen	4-5		Reserved.
Return Value	6-7		Return Value: 0x0 = No error (success). EIO = Flash defect.
Cookie High	8-11	Cookie	Opaque value that is copied by the EMP into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value that is copied by the EMP into the completion of this command.
Reserved	16		Reserved.
Module_pointer	17		Copied from the command.
Length	18-19		Copied from the command. If a value of 0xFFFF was set in the admin command, this field returns the length from the offset to the module's end in 4 KB units.
Offsets	20-23	0x0	Byte 23= Reserved, must be zeroed. Bytes 22:20= Copied from the command.

Table 38-76. NVM Erase Completion (on ARQ) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Reserved	24-27		Reserved.
Reserved	28-31		

38.11.14.3 NVM Update

This command is used to write the data given by the attached buffer into a specified location in the NVM. Erasing the relevant sector(s) by posting NVM erase command(s) (see Section 38.11.14.2) is required prior to posting this command.

Table 38-77. NVM Update Admin Command (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0703	Command opcode.
Datalen	4-5		Length in bytes of command buffer.
Return Value/ VFID	6-7		Must be zeroed by the software device driver.
Cookie High	8-11	Cookie	Opaque value that is copied by the EMP into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value that is copied by the EMP into the completion of this command.
Command_flags	16		NVM Access Admin Command Parameters Bit 0 = Last command bit used to notify EMP that this is the last admin command of a sequence. Bits 2:1 = Preservation mode: • 00b = No preservation. Notify the EMP to avoid any field preservation during a NVM bank update. • 01b = Preserve all (default). Used to notify the EMP to preserve all the necessary customization fields during a NVM bank update. • 11b = Preserve only selected fields. Used to notify EMP to preserve only the immediate fields and section marked as such during a NVM bank update. — PCI serial ID — PF MAC addresses section — Port MAC addresses — Unicast destination address — Software RDMA MAC addresses section — SMBus addresses Bits 6:3 = Reserved, must be zeroed. Bit 7 = Flash-only bit. When bit 7 is set, any flat write (null Module_pointer) into the first 128 KB of the Flash is written directly to the Flash and not in shadow RAM. The bit has no effect when attempting a flat write outside the first 128 KB of the Flash.
Module_pointer	17		Module pointer location in words from the NVM beginning. Attempting to write a RO module is not allowed. A value of 0x00 here means that the command is performed over the Flash part seen as a flat memory. No reset or pointer switch is initiated by the 10 GbE controller in such a case. Any flat write attempt to the first 64 KB of the Flash is performed against the shadow RAM first, and then dumped to the next basic bank in the Flash (see Section 38.11.9.3). Any flat write attempt to the second 64 KB of the Flash is rejected as for write attempts to a RO module. A flat write attempt to the area of a RO module or to a RO word is rejected as well.

Table 38-77. NVM Update Admin Command (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Length	18-19		Length of the section to be written, which is expressed in bytes from the offset in the module. A (single) write command is limited up to 4 KB and must not spread over two (consecutive) 4 KB sectors. Also, attempting to write a RO word invalidates the entire command. These values must be even (word alignment) when writing to CSR auto-load sections of the shadow RAM area.
Offset	20-23		Byte 23 = Reserved, must be zeroed. Bytes 22:20 = Offset, which is expressed in bytes from the pointed module's beginning. This is the byte offset of the first byte to be written. These values must be even (word alignment) when writing to CSR auto-load sections of the shadow RAM area.
Data Address High	24-27		Address of command buffer.
Data Address Low	28-31		

This command is an asynchronous command. EMP reads the command from the ATQ and writes back an immediate completion, intended only as an ACK/NACK that the command has been addressed by the EMP. The EMP checks the validity of the command and returns an error (NACK) on the ATQ completion if it is unable to process the command. If successful (ACK), the EMP then schedules the NVM erase operation to be performed by a lower priority thread, which can be preempted by other AQ commands. Once completed, the EMP posts a completion event on the ARQ. Software must hold the NVM resource lock while performing this operation and must release it once NVM operations complete. Software must not post another NVM command while this command is in process.For example, during the time between posting the request on the ATQ and receiving the completion event on the ARQ.

Table 38-78. NVM Update Response

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0703	Command opcode.
Datalen	4-5		Length in bytes of command buffer.
Return Value	6-7		Return Value 0x0 = No error (success). EPERM = The module pointer location specified in the command does not permit the required operation. The word contents is not a pointer, or an attempt to write a RO module or word. EINVAL = Out of range offset/length (beyond the relative free area module's limits), or write spread over two (consecutive) sectors. EBUSY = The PF is not permitted to post this command because it does not own the NVM resource. This error code is also returned if the PF attempts to post a command while another NVM command is in process.
Cookie High	8-11	Cookie	Opaque value that is copied by the EMP into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value that is copied by the EMP into the completion of this command.
Reserved	16-31		Reserved.

Table 38-79. NVM Update Completion (on ARQ)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0703	Command opcode.
Datalen	4-5		Length in bytes of command buffer.
Return Value	6-7		Return Value 0x0 = No error (success). EIO = Flash defect. EACCES = Security check failed: • Public key check failed. • Module digest check failed. • Module rollback revision check failed. • Device ID check failed. • Module ID check failed.
Cookie High	8-11	Cookie	Opaque value that is copied by the EMP into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value that is copied by the EMP into the completion of this command.
Reserved	16		Reserved.
Provisioning_pointer	17		Location in words from the NVM beginning with the free provisioning pointer used for the command. A value of 0x0000 is returned if the command was performed against the shadow RAM.
Length	18-19		Copied from the command.
Offsets	20-23		Byte 23 = Reserved, must be zeroed. Bytes 22:20 = Copied from the command.
Data Address High	24-27		Address of command buffer.
Data Address Low	28-31		

38.11.14.4 NVM Config Read Admin Command

This Admin command reads currently configured feature selections and immediate field values. The features/fields to be read are specified in the command's buffer. It can also be used for reading all features or fields. Feature or field iteration is used for that. The next Feature_ID / Field_ID to read are returned in the command response in this case.

Table 38-80. NVM Config Read Admin Command

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0704	Command opcode.
Datalen	4-5		Length in bytes of command buffer.
Return Value/ VFID	6-7		Must be zeroed by the software device driver.
Cookie High	8-11	Cookie	Opaque value that is copied by the EMP into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value that is copied by the EMP into the completion of this command.
Command_flags	16		NVM Access Admin Command Parameters. Bit 0 = Single/ Multiple elements. 0b = Only a single Feature_ID, Field_ID is read. 1b = Feature_ID/Field_ID iteration is used. Bit 1 = Feature/field. 0b = Feature selections are read. 1b = Immediate fields are read. Bits 2:7 = Reserved (must be set to zero).
Reserved	17	0x0	Reserved (must be set to zero).
Element Count	18-19	0x0	The number of features/fields returned (zeroed by the software device driver and written by firmware).
Feature_ID/ Field_ID	20-21	See description	Single Feature_ID/Field_ID when bit 16.0 is 0. Feature_ID when bit 16.1 is 0. Field_ID (LS word) when bit 16.1 is 1. Feature_ID/Field_ID to start reading from (iterator) when bit 16.0 is 1. Note Feature_ID/Field_ID = 0 is not valid and indicates that the command should read the data starting from the first Feature_ID/Field_ID in the array. This is only relevant when bit 16.0 is 1.
Reserved	22-23		Reserved
Data Address High	24-27		Address of command buffer.
Data Address Low	28-31		

Table 38-81. NVM Config Read Response

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0704	Command opcode.
Datalen	4-5		Length in bytes of command buffer.
Return Value	6-7		Return Value. 0x0 = No error (success). Others - Error detected in the command.
Cookie High	8-11	Cookie	Opaque value that is copied by the EMP into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value that is copied by the EMP into the completion of this command.
Command_flags	16		NVM Access Admin Command Parameters Bit 0 = Single/ Multiple elements. 0b = Only a single Feature_ID, Field_ID is read. 1b = Feature_ID/Field_ID iteration is used. Bit 1 = Feature/field. 0b = Feature selections are read. 1b = Immediate fields are read. Bits 2:7 = Reserved (must be set to zero).
Reserved	17	0x0	Reserved (must be set to zero).
Element Count	18-19	See description	The number of features/fields returned.
Feature_ID/ Field_ID	20-21	See description	Single Feature_ID/Field_ID when bit 16.0 is 0. Feature_ID when bit 16.1 is 0. Field_ID (LS word) when bit 16.1 is 1. Feature_ID/Field_ID to start reading from (iterator) when bit 16.0 is 1. Note Feature_ID/Field_ID = 0 is not valid and indicates that the command should read the data starting from the first Feature_ID/Field_ID in the array. This is only relevant when bit 16.0 is 1.
Reserved	22-23		Reserved.
Data Address High	24-27		Address of command buffer.
Data Address Low	28-31		

38.11.14.5 NVM Config Write Admin Command

This admin command writes the feature selections and the values of the immediate fields provided in the attached command buffer to the NVM.

Table 38-82. NVM Config Write Admin Command (0x0705) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0705	Command opcode.
Datalen	4-5		Length in bytes of command buffer.
Return Value/ VFID	6-7	0x0	Must be zeroed by the software device driver.
Cookie High	8-11	Cookie	Opaque value that is copied by the EMP into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value that is copied by the EMP into the completion of this command.

Table 38-82. NVM Config Write Admin Command (0x0705) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Command_flags	16		NVM Access Admin Command Parameters Bit 0 = Reserved zero. Bit 1 = Feature/field. 0b = Feature selections are written. 1b = Immediate fields are written. Bit 2 = Add new configuration. 0b = Regular command. 1b = New configuration added. Bits 3:7 = Reserved, must be zero.
Reserved	17	0x0	Reserved, must be zero.
Element Count	18-19	See description	The number of features/fields in the command buffer.
Reserved	20-23	0x0	Reserved, must be zero.
Data Address High	24-27		Address of command buffer.
Data Address Low	28-31		

Table 38-83. NVM Config Write Response (0x0705)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0704	Command opcode.
Datalen	4-5		Length in bytes of command buffer.
Return Value	6-7		Return Value. 0x0 = No error (success). Others - Error detected in the command.
Cookie High	8-11	Cookie	Opaque value that is copied by the EMP into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value that is copied by the EMP into the completion of this command.
Reserved	16-31		Reserved.

38.11.14.6 NVM Config Read / Write Command Buffer

Following is the format of the command buffer attached to the NVM Config Read response and to the NVM Config Write command. The buffer can either be filled with Feature selection fields or Immediate Field fields, depending on bit 16.1 in the command.

Table 38-84. Feature Buffer For NVM Config Read /Write

Parameter	Bytes.Bits	Description
Feature_ID	0-1	Feature_ID
Feature Options	2-3	Feature options for NVM_Config_Read only. Reserved for NVM_Config_Write Bit 0 = OEM only (should be set). Bit 2:1 = Reserved. Bit 3 = If set the Feature fields are mapped in Dword-wise, otherwise are word. Bit 4 = Should be set only in the 10 GbE controller when using a POR CSR. Bit 15:5 = Reserved.
Feature Selection/ Field Value	4-5	Configured feature selection for NVM_Config_Read. Requested feature selection for NVM_Config_Write.

Table 38-85. Immediate Buffer for NVM Config Read /Write

Parameter	Bytes.Bits	Description
Field_ID	0-3	Field_ID
Field Options	4-5	Field options for NVM_Config_Read Reserved for NVM_Config_Write
Field Value	6-9	Field Value

38.11.15 VPD Support

The Flash image can contain an area for VPD. This area is managed by the OEM vendor and does not influence the behavior of hardware. Word 0x2F of the Flash image contains a pointer to the VPD area in the Flash. A value of 0b in the GLPCI_CAPCTRL.VPD_EN register bit means VPD is not supported and the VPD capability does not appear in the configuration space. The register bit must be set to 1b in NVM only once the VPD area has been programmed. Refer to Section 38.11.9.2.1.

The maximum VPD area size provisioned in shadow RAM is 1 KB but it can be smaller. The VPD block is built from a list of resources. A resource can be either large or small. The structure of these resources are listed in the following tables.

Table 38-86. Small Resource Structure

Offset	0	1 — n
Content	Tag = 0xxx,xyyyb (Type = Small(0), Item Name = xxxx, length = yy bytes)	Data

Table 38-87. Large Resource Structure

Offset	0	1 — 2	3 — n
Content	Tag = 1xxx,xxxxb (Type = Large(1), Item Name = xxxxxxxx)	Length	Data

The 10 GbE controller parses the VPD structure during the auto-load process following PCIe reset in order to detect the read only and read/write area boundaries. The 10 GbE controller assumes the following VPD fields with the limitations listed.

Table 38-88. VPD Structure

Tag	Length (bytes)	Data	Resource Description
0x82	Length of Identifier String	Identifier	Identifier string.
0x90	Length of RO Area	RO Data	VPD-R list containing one or more VPD keywords.
0x91	Length of RW Area	RW Data	VPD-W list containing one or more VPD keywords. This part is optional.
0x78	n/a	n/a	End tag.

VPD structure limitations:

- The structure must start with a tag = 0x82.
- The structure must end with a tag = 0x78 before the shadow RAM's end. The tag must also be word aligned.

- If the 10 GbE controller does not detect a value of 0x82 in the first byte of the VPD area, or if no end tag is detected, or if the structure does not follow the information listed in Table 38-88, it assumes the area is not programmed:
  - Any read/write access through the VPD registers set are ignored.
  - EMP considers the VPD area as an empty module and thus allows NVM Update commands to be performed over the area pointed by the VPD area pointer, up to the start of another RO module.
  - The VPD pointer itself remains RO.
- The RO area and RW area are both optional and can appear in any order. A single area is supported by per-tag type. Refer to Appendix I in the PCI 3.0 specification for details of the different tags.
- If a VPD-W tag is found, the area defined by its size is writable via the VPD structure.
- The VPD area can be accessed through the PCIe configuration space VPD capability structure listed in Table 38-88. Write accesses to a RO area or any accesses outside of the VPD area via this structure are ignored. If the VPD *Write Enable* field is set to 1b in the NVM Security Control word, the entire VPD area can be modified via the NVM Update AQ command. Otherwise, the command is completed with an error status.
- VPD area must be mapped into the first valid basic bank of the Flash.
- VPD software does not check the NVM ownership before attempting to access the Flash via dedicated VPD registers. VPD software write access is recorded in the Flash immediately once the Flash part is available (such as not busy by a previous sector erase operation). Refer to Section 38.11.9.2 for more details.

38.12 General Purpose I/O (GPIO)

The 10 GbE controller has a total of 16 GPIOs pins that can be configured as SDPs, LED drivers or dedicated hardware functions such as for connecting to external PHYs or IEEE 1588 auxiliary devices. The GPIO pins can also be configured to be associated with any of the physical ports. The following sections describe the possible configurations for the GPIO pins. Many of the GPIO pins are reserved for specific use and hence named by default as SDP or LED signals. This offers the flexibility to configure any of the GPIO pins, irrespective of their names, to different modes and associated with different ports.

The GPIO pin functionality and other attributes such as I/O direction, default value, interrupt mode etc., can be configured through Global GPIO Control registers. See the Table 38-89 for the list of GPIO pins and their corresponding configuration control registers. The GLGEN_GPIO_CTL has a PIN_FUNC field to configure the pin functionality such as SDP, LED, RoL or 1588 TimeSync modes. The PRT_NUM field is used to configure the port number associated with the GPIO pin. The INT_MODE field is used to configure the GPIO pin to generate an interrupt on the rising edge, falling edge or on both edges. The PHY_PIN_NAME is used to indicate how the GPIO is connected to an external PHY. See the GLGEN_GPIO_CTL register description for additional configuration fields and options. The default value for this register is loaded from the NVM and is typically dependent on the board layout/configuration, number of ports, and PHYs present on the board.

Table 38-89. GPIO Pin Configuration Registers

GPIOn Index (n= 0..29)	Pin Name	GPIO Register
GPIO0	SDP0_0	GLGEN_GPIO_CTL[0]
GPIO1	SDP0_1	GLGEN_GPIO_CTL[1]
GPIO2	SDP1_0	GLGEN_GPIO_CTL[2]
GPIO3	SDP1_1	GLGEN_GPIO_CTL[3]
GPIO4	SDP2_0	GLGEN_GPIO_CTL[4]
GPIO5	SDP2_1	GLGEN_GPIO_CTL[5]
GPIO6	SDP3_0	GLGEN_GPIO_CTL[6]
GPIO7	SDP3_1	GLGEN_GPIO_CTL[7]
GPIO22	LED0_0	GLGEN_GPIO_CTL[22]
GPIO23	LED0_1	GLGEN_GPIO_CTL[23]
GPIO24	LED1_0	GLGEN_GPIO_CTL[24]
GPIO25	LED1_1	GLGEN_GPIO_CTL[25]
GPIO26	LED2_0	GLGEN_GPIO_CTL[26]
GPIO27	LED2_1	GLGEN_GPIO_CTL[27]
GPIO28	LED3_0	GLGEN_GPIO_CTL[28]
GPIO29	LED3_1	GLGEN_GPIO_CTL[29]

Note: GPIO22 through GPIO29 are only used for LED functionality. As such, the GLEN_GPIO_CTL[29:22].*PIN_FUNC* field must be set to LED (001b).

The following registers are used to read and write to the GPIO pins. The GLGEN_GPIO_STAT register is used to read the status of the GPIO pins. This register returns the actual value (0b = high, 1b = low) on the pin. The GLGEN_GPIO_TRANSIT register is used to latch any transition (low-to-high or high-to-low) on the GPIO pins since the last time the register was cleared. The GLGEN_GPIO_SET register is used to set the value (0b = low, 1b = high) of the GPIO output pins when configured as an SDP. The PF (software) is expected to write to a GPIO pin only if it owns it.

38.12.1 LEDs

The 10 GbE controller designates eight pins named LED0_0 through LED3_1 for driving LEDs for ports 0 through 3. However, depending on the board layout, these might be replaced or complemented with any of the GPIO pins by configuring the GPIO to be a LED driver through the GLGEN_GPIO_CTL register as described in Section 38.12.2.1.

In order to configure a GPIO pin as an LED driver using the GLGEN_GPIO_CTL register:

- Set PIN_FUNC to LED
- Set PIN_DIR to output
- Use the PORT_NUM field to assign a port number to the LED pin

The output polarity of the LED pin (active high or active low) can be configured through the LED_INVRT field and the blink mode (blinking versus steady) is configured through the LED_BLINK field. The LED outputs can be individually configured to indicate a particular event, state, or activity by using the LED_MODE field. See Table 38-90 for more information on configuring the LED source mode field. The hardware default configuration for GPIO pins configured as LED outputs can be specified via NVM fields

thereby supporting LED displays configurable to a particular OEM preference. Note that at run-time there is usually no need to override the LED setting, the only exception would be to force a LED to blink in order to physically identify a port.

Table 38-90. LED Source Modes

LED_MODE Value	Mode	Condition[1]
LED_MODE[4] = 0		
0000b	LED_OFF	Always off.
0001b	LINK	True while link is held.
0010b	Reserved	Reserved
0011b	Reserved	Reserved
0100b	Reserved	Reserved
0101b	Reserved	Reserved
0101b	Reserved	Reserved
0110b	LINK_10G	True while the 10 GbE controller has 10 Gb/s link.
0111b	LINK_1G	True while the 10 GbE controller has 1 Gb/s link.
1000b	LINK_10G/1G	True while the 10 GbE controller has 10 Gb/s or 1 Gb/s link.
1001b	Reserved	Reserved
1010b	Combined Ports Activity	Active when any one of the 10 GbE controller's port has an established link with packets being transmitted or received. In this mode LED_BLINK must be set.
1011b	Reserved	Reserved
1100b	LINK_ACT	Asserted steady when link is established and there is no transmit or receive activity. Blinking when there is link and receive or Transmit activity. In this mode LED_BLINK must be cleared at 0b.
1101b	MAC_ACT	Active when link is established and packets are being transmitted or received. In this mode, the LED_BLINK must be set.
1110b	FILTER_ACT	Active when link is established and packets are being transmitted or received that passed MAC filtering. In this mode, the LED_BLINK must be set.
1111b	LED_ON	Always true. Overrides all other settings.
0000b	AGGREGATE_LINK_10G	Asserted when ALL enabled ports have 10 Gb/s link.
0001b	COMBINED_LINK	Asserted when ANY of the enabled ports has link.
0010b	Reserved	Reserved.
0011b	COMBINED_LINK_10G	Asserted when at least one of the enabled ports has 10 Gb/s link.
0100b	COMBINED_LINK_1G	Asserted when at least one of the enabled ports has 1 Gb/s link.
0101b	COMBINED_LINK_10G_NOT_ALL	Asserted when at least one of the enabled ports has 10 Gb/s link but not ALL. ***Note:*** This LED is never asserted when only one port is enabled.
Other	Reserved	Reserved.

Notes:

1. When the condition is true, the LED might blink or be constantly on, depending on the value of LED_BLINK field.

The *LED_INVRT* bit enables the LED source to be inverted before being output or observed by the blink-control logic. LED outputs are assumed to normally be connected to the negative side (cathode) of an external LED.

The LED_BLINK bit controls whether the LED should blink while the LED source is asserted. The blink control can be especially useful for ensuring that certain events, such as ACTIVITY indication, cause LED transitions, which are sufficiently visible to the human eye. The global blink mode bit in GLGEN_LED_CTL register is used to select between blinking every 200 ms or every 83 ms.

Note: LED_ON is not affected by the LED_BLINK control.

Note: The LINK/ACTIVITY source functions slightly different from the others when BLINK is enabled. The LED is:

Off if there is no LINK

On if there is LINK and no ACTIVITY

Blinks if there is LINK and ACTIVITY

Note: Only one software device driver should override any of the LED settings at any time (host-wide). Since this is an administrator specified override, hardware does not enforce this. Typically, there is no need to override the LED hardware defaults loaded from the NVM in a specific board configuration.

The Activity LED does not monitor link status of a specific port. The Activity LED blinks each time a packet is received or transmitted on a port. It blinks regardless of whether it is a host packet or a management packet. In certain cases, the BMC might send a packet to the MAC and the Activity LED might blink correspondingly. If there is no cable connected to the port, then the activity LED can be ignored in this case.

38.12.2 Software-Definable Pins (SDPs)

The 10 GbE controller enables any of the GPIO pins to be configured as SDPs through the GLGEN_GPIO_CTL register as described in Section 38.12.2.1. The 10 GbE controller has designated 16 GPIO pins, SDP0_0 (GPIO0) through SDP3_1 (GPIO7), as SDPs for use with ports 0 through 3. However, depending on the board layout, any of the GPIO pins, irrespective of their name, could be configured as SDPs. In order to configure a GPIO pin for SDP use, set PIN_FUNC to SDP and PIN_DIR to either input or output in the GLGEN_GPIO_CTL register. Use the PORT_NUM field to assign a port number to be associated with the SDP.

The 10 GbE controller SDP pins can be used for hardware connectivity to low-speed, optical-module interfaces, external PHY control or software controllable purposes. The SDP pins can also be configured for use as external interrupt sources. The SDP pins, port ownership, direction and their functions are bound to specific board layout/ configuration. The default values for GPIO control registers (GLGEN_GPIO_CTL[n]) are loaded from the NVM. Typically, there is no need for software to change the default configuration assigned for a specific board layout. Pins assigned for PHY management are owned by firmware and should not be accessed directly by software.

The 10 GbE controller SDP pins can also be configured for use as general purpose external interrupt sources (GPI). To act as GPI pins, the desired pins must be configured as inputs and enabled by the INT_MODE field in the GLGEN_GPIO_CTL register. The INT_MODE field can be used to configure the pin to generate an interrupt on the rising edge, falling edge or on both edges. Rising or falling edge detection occurs by comparing values sampled at the internal clock rate, as opposed to an edge-detection circuit. When detected, a GPIO interrupt is indicated in the PFINT_ITR0 register. The PFINT_GPIO_ENA register is used to enable interrupts to PFs (software) and the EMPINT_GPIO_ENA register is used to enable an interrupt to the EMP (firmware). GLGEN_GPIO_STAT and/or GLGEN_GPIO_TRANSIT registers can be used to read the status of the GPIO pins that caused the interrupt. Software is expected to enable interrupts only for pins that are owned by the PFs.

Software or firmware can read/write to a GPIO/SDP pin as described in Section 38.12.2.1.

The proposed use of SDPs for controlling external PHY devices or modules is explained in Section 38.11.1. Proposed use of the SDPs for 1588 functionality is explained in later in this document and other use cases.

38.12.2.1 Global GPIO Pins

The 10 GbE controller has designated six general purpose I/O pins as Global GPIO pins named GPIO_0 through GPIO_5. These pins are not assigned for a specific usage or port. The mode and port association for these pins can be configured as described in Section 38.12. These pins can be used as SDP, LED, or 1588 TimeSync modes or as general purpose interrupt sources. System designers have the flexibility to assign these GPIO pins for specific use depending on the board configuration.

38.12.3 BMC Interconnects

The 10 GbE controller supports three sideband interfaces:

- SMBus
- Network Controller-Sideband (NC-SI)
- PCIe (together with MCTP)

The SMBus and NC-SI interfaces are described in the sections that follow.

38.12.4 SMBus

SMBus is an optional interface for pass-through and/or configuration traffic between an external BMC and the 10 GbE controller.

The 10 GbE controller also enables reporting and controlling itself using the MCTP protocol over SMBus. The MCTP interface is used by the BMC to only control the NIC and not for pass through traffic. All network ports are mapped to a single MCTP endpoint on SMBus.

38.12.4.1 Channel Behavior

The SMBus specification defines a maximum frequency of 100 kHz. However, when acting as a slave, the 10 GbE controller can receive transaction with a clock running at up to 1 MHz. When acting as a master, it can toggle the clock at 100 kHz, 400 kHz or 1 MHz. The speed used is set by the *SMBus Connection Speed* field in the *SMBus Notification Timeout and Flags* NVM word.

38.12.5 NC-SI Interface

The NC-SI interface in the 10 GbE controller is a connection to an external BMC defined by the DMTF NC-SI protocol. It operates as a single interface with an external BMC, where all traffic between the 10 GbE controller and the BMC flows through the interface.

38.12.5.1 Electrical Characteristics

The 10 GbE controller complies with the electrical characteristics defined in the NC-SI specification.

The 10 GbE controller's NC-SI behavior is configured at power-up in the following manner:

- The *Multi-Drop NC-SI* NVM bit defines the NC-SI topology (point-to-point or multi-drop; the default is point-to-point).

The 10 GbE controller dynamically drives its NC-SI output signals (NC-SI_DV and NC-SI_RX) as required by the sideband protocol:

- At power-up, the 10 GbE controller floats the NC-SI outputs.
- If the 10 GbE controller operates in point-to-point mode, it starts driving the NC-SI outputs some time following power-up.
- If the 10 GbE controller operates in a multi-drop mode, it drives the NC-SI outputs as configured by the BMC.

38.12.5.2 NC-SI Transactions

The NC-SI link supports both pass-through traffic between the BMC and the 10 GbE controller LAN functions, as well as configuration traffic between the BMC and the 10 GbE controller's internal units as defined in the NC-SI protocol.

38.13 Initialization

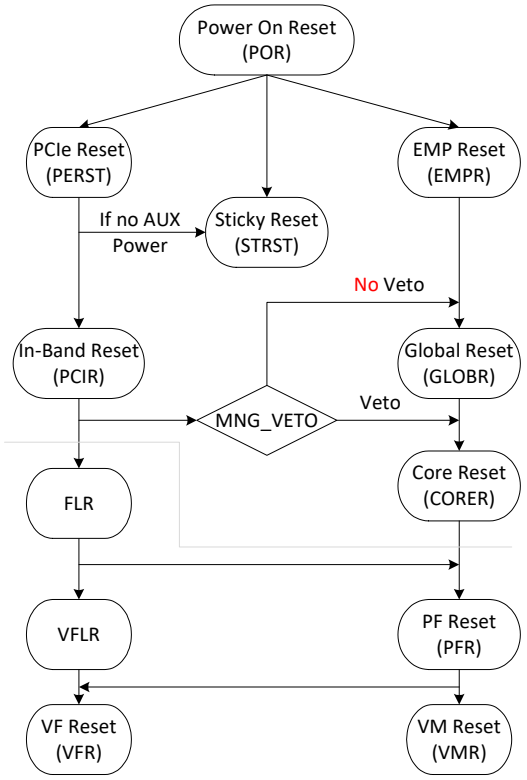
38.13.1 Reset Operation

The following sections lists the hardware and software reset sources that initialize the entire portions of the 10 GbE controller and functions level resets. The reset sources are listed in Section 38.13.1.1 and the reset flows are detailed in Section 38.13.1.2.

38.13.1.1 Reset Sources

This section lists the reset sources supported by the 10 GbE controller while the complete list of initialized logic is listed in Table 38-1. Hierarchical reset tree is shown in the Figure 38-24. Any logic initialized by a specific reset is initialized also by any other reset source that is linked to it and located above it in the reset tree.

Figure 38-24.Hierarchical Reset Tree



Reset Source	Cause List of the Reset Source
Power On Reset (POR)	Power applied to the device
Sticky Reset (STRST)	Internal reset that clears PCIe sticky registers following PCIe reset
PCIe Reset (PERST)	PE_RST_N input signal
In-Band Reset (PCIR)	In-Band PCIe message signalling
EMP Reset (EMPR)	EMP watchdog timer expiration or Force TCO command
Global Reset (GLOBR)	Device driver setting the GLOBR bit in the GLGEN_RTRIG register
Core Reset (CORER)	Device driver setting the CORER bit in the GLGEN_RTRIG register
PF Reset (PFR)	FLR assertion via the PCIe configuration space D3 to D0 transition or Clearing the BME Device driver setting the PFSWR bit in the PFGEN_CTRL register
VM Reset (VMR)	Device driver setting the VMR bit in the VSIGEN_RTRIG register
VF Reset (VFR)	VFLR assertion or setting the VFSWR bit in the GLGEN_VFRTRIG register
PF Level reset (FLR)	Operating sytem sets the FLR bit in the PF PCIe config
VF Level reset (VFLR)	Operating system sets the FLR bit in the VF PCIe config or clearing the VFE bit in the parent s PF PCIe config

Power On Reset (POR) — The 10 GbE controller power on reset signal is translated to a sideband reset and a global reset signal to the 10 GbE controller.

PCIe Reset (PERST) — The PCIe reset signal is kept at the low level when the system is at power down state and at system boot. Transit of PCIe reset to low generates an internal reset signals. If there is no AUX power, the PERST generates an internal STRST signal that clears PCIe sticky bits as listed in Table 39-1. PERST also triggers internally a PCIR, which is detailed later in this section.

Sticky Reset (STRST) — Sticky reset is internal signal triggered by PCIe reset when the 10 GbE controller is not powered by AUX power. This reset clears sticky registers in the PCIe interface (as defined by the PCIe specification). The sticky reset is also initiated by POR.

In-Band Reset (PCIR) — The PCIe supports in-band signaling for PCIe reset (called PCIR). Any cycles on the PCIe bus are gated instantly as well and packet transmission generated by software. The entire data path is initialized other than the EMP cluster. Although the EMP subsystem is not initialized, some packets of the EMP might be lost during a short window of about 1 µs.

Function Level Reset (FLR) — The 10 GbE controller supports the standard FLR interface in the Device Control register of the PCIe capability structure within the PCI configuration space of the PFs. Setting the FLR bit initializes the PCI configuration of the PF (including the VFE flag in the SR-IOV Control/Status register that disables all the VFs of the PF) and initiates an internal PFR described later in this section.

PF Reset (PFR) — PFR initializes the resources and data path of the PF and its VFs with no impact on other PFs, VFs and the EMP subsystem. Any further master cycles of the PF are not initiated while some packets that were already fetched completely might still be sent out. The PFR is generated by one of the following four causes: (1) D0 to $D3_{HOT}$ transition, which is also known as ACPI reset; (2) FLR; (3) PF software sets the PFSWR bit in the PFGEN_CTRL register; (4) De-assertion of the *Bus Master Enable* flag in the PCI configuration space.

VF Level Reset (VFLR) — The 10 GbE controller supports the standard VFLR interface in the Device Control register of the PCIe capability structure within the PCI configuration space of the VFs. Setting the VFLR bit initializes the PCI configuration of the VF and initiates an interrupt to the PF that completes the VFR reset described later in this section. Clearing the VFE flag in the PF configuration space also impacts all its VFs the same as a VFLR.

VF Reset (VFR) — VFR initializes the resources and data path of the VF with no impact on other PFs, VFs and the EMP subsystem. Any further master cycles of the VF are not initiated while some packets that were already fetched completely might still be sent out. The VFR is generated by one of the following two causes: (1) VFLR or clearing the *VFE* bit of the parent PF. Note that after the *VFE* bit is cleared, the PF driver should follow the VFR flow for all the VFs of the PF, including those VFs that are not enabled; (2) PF software sets the *VFSWR* bit in the VPGEN_VFRTRIG register of its VF.

VM Reset (VMR) — There are 384 VSIs where up to 256 of them can be VMDq2 VSIs. Such VSIs are associated with VMs. VMR is a mechanism to reset a VMDq2 VSI. VMR initializes the resources and data path of the VM with no impact on other PFs, VFs, VMs and the EMP subsystem. Any further master cycles of the VM are not initiated while some packets that were already fetched completely might still be sent out. The VMR is generated by the PF software setting the *VMSWR* bit in the VSIGEN_RTRIG register.

Core Reset (CORER) — CORER initializes the shared data path for all functions excluding the EMP subsystem, PCI interface and MAC/PHY logic of all ports. Any further master cycles of all PFs and VFs are not initiated while some packets that were already fetched completely might still be sent out. Even though the EMP subsystem is not cleared, pass-through traffic might be inhibited during the initialization cycle that might take ~20 ms. Also, SMBus accesses are responded with NACK during the initialization cycle. This reset is not expected to be used other than as an escape mechanism in case the 10 GbE controller hangs and the PFR did not resolve the problem. This reset is initiated by the PFs by setting the *CORER* bit in the GLGEN_RTRIG register. The EMP initiates this reset by setting a bit in an internal register.

Global Reset (GLOBR) — GLOBR is a super-set CORER initializing any logic initialized by the CORER plus the MAC/PHY logic of all ports (both internal PHY and external PHY if connected). This reset is not expected to be used other than escape mechanism in case CORER did not resolve the problem. This reset is initiated by the PFs by setting the *GLOBR* bit in the GLGEN_RTRIG register. The EMP initiates this reset by setting a bit in an internal register. Global reset is also initiated following a Force TCO command (if it is not disabled by the F*orce TCO Reset Disable* bit in the NVM).

EMP Reset (EMPR) — EMPR initializes the resources and data path connected to the EMP including its firmware reload. EMPR is triggered internally by the EMP watchdog timer expiration or by EMP setting an internal flag or due to Force TCO command or due to uncorrectable ECC error in one of the EMP memory shells.

Reset Activation	POR	STRST/ PCIR/ PERST	EMPR	GLOBR	CORER	PFR/ FLR	VMR	VFR/ VFLR
Load the NVM to the shadow RAM in the hardware and clear the alternate structure.	+							
Load the EMP and PE firmware from the NVM.	Both+	PE	Both+	PE	PE			
Load device settings from NVM shadow and alternate structure.	Shad	Both	Both	Both	Both			
PF MAC addresses (switch and WoL).[a]	+	+ [a]	+	+	+	+		
MAC and PHY interface.	+	[1]	+	+				
EMP subsystem including firmware reload.	+		+					
Sticky PCIe context.	+	[2]						
PCIe HWInit parameters.	+	PERST only						
PCIe RO registers.	+	+						
PCIe RW/RW1C registers.	+	+				+ [3]		VF by VFLR
Bus master disable.[4]	+	+	+	+	+	PF	VM	VF
All CSRs - refer to the Programming Interface section for the reset source of each register.								
Most of the PF and VF registers (PF registers are named - PFxxx and VF registers named - VFxxx and VPxxx). refer to the Programming Interface section for the reset source of each register.	+	+	+	+	+	PF and its VFs		VF only
Cache contexts (filters, queue context, PE context) and FPMs settings (sector descriptors and cache entries).	+	+	+	+	+	PF and its VFs	VM	VF
Invalidate VF queue mapping tables (VPLAN_QTABLE).	+	+	+	+	+	VFs of the PF		VF
Invalidate VSI context (including the switch, VSILAN_QTABLE and all other VSI registers).	+	+	+	+	+	PF and its VFs	by the PF SW[5]	by the PF SW[5]
Load the PFs MAC address to the switch and the WOL filters from the NVM (not all PFs must have a WOL filter).	+	+ [6]	+	+	+	PF		
Tx and Rx data path + Tx scheduler.	+	+	+	+	+	PF and its VFs	VM	VF
Tx and Rx packet buffers.	+	+	+	+	+			
Tx and Rx queue disable.	+	+	+	+	+	PF and its VFs	by the PF SW	VFs
Admin queue disable (POR and EMP also clear the queue context memories).	+	+	+	+	+	PF and its VFs		VF
Disable interrupts.	+	+	+	+	+	PF and its VFs		VF
Interrupt cause control registers.	+	+	+	+	+	by the PF SW	by the PF SW	by the PF SW
RSS key and table.	+	+	+	+	+	PF		VF
Invalidate FD filters.	+	+	+	+	+	PF		

Notes:

1. PF MAC addresses (switch and WoL) are cleared by hardware at POR and loaded from NVM by the firmware at any of the highlighted reset causes. Following PCIR, the switch MAC addresses are loaded following the assertion of the PCIe reset or in-band reset and the WoL MAC addresses are loaded only following the de-assertion of the PCIe reset.The MAC and PHY are. If the WUC filter is inactive and no port is needed by FW/MNG (PRTPM_GC. EMP_LINK_ON is clear), GLOBR is maintained low by hardware during PCIR de-assertion for power savings.
2. Sticky PCIe context is cleared on PERST if no AUX power.
3. For exception list of registers that are not cleared on PFR.
4. The 10 GbE controller has several flags that control the Bus Master Enable (BME). BME flags on the PCI configuration space (for each PF and VF) and the VMRD flag in the VSIGEN_RSTAT registers for each VM that is not a VF.
   The BME flags of all PFs are cleared by all reset causes indicated by + symbol and a specific PF by FLR.
   The BME flags of all VFs are cleared by all reset causes indicated by + symbol and a specific VF by VFLR.
   The VMRD flags of all VSIs are set by all reset causes indicated by + symbol and a specific VSI by VMR.
   Note that as opposed to BMEs of the function that are cleared by the previous resets (disabling master accesses), the VMRD flags are set (enabling bus master accesses when the BME of their parent PF is set as well).
5. VSIs and its related switch context are cleared by Admin command(s) initiated by the software.
6. Note that as opposed to any logic in The 10 GbE controller that is initialized at the leading edge of PERST#, the WOL filters are initialized at the de-assertion of PERST# (entering the D0u state).

Table 38-91. NVM Section Loaded by Reset Source Covered by GLNVM_ULD Register

NVM Section	POR	PERST	PCIR	EMPR	GLOBR	CORER
POR registers auto-load.	+					
PCIe analog configuration.	+					
PCIR registers auto-load.	+	+	+			
PCIe Transaction Layer (TL) Shared.	+	+	+			
RO PCIe LCB.	+	+				
CORER registers auto-load.	+	+	+	+	+	+
GLOBR registers auto-load.	+	[1]	1	+	+	
PHY analog configuration.	+					
EMPR registers auto-load.	+			+		

Notes:

1. GLOBR registers auto-load only if the MNG_VETO flags in all four PRTPM_GC registers are cleared.

38.13.1.2 Reset Flows

This section describes the reset flows in the 10 GbE controller and software interaction.

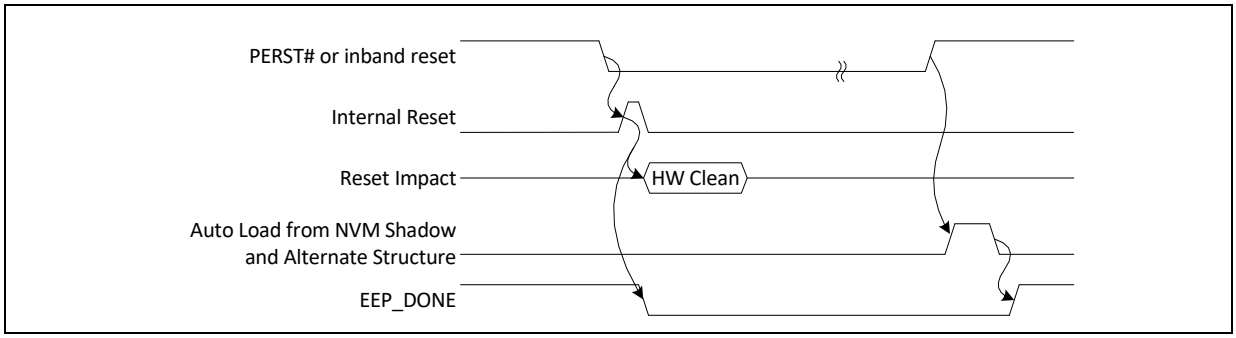
38.13.1.2.1 POR Flow

Refer to the POR flow section for more details.

38.13.1.2.2 PCI Reset and In-Band PCI Reset Flow

The internal reset flow is shown in Figure 38-25 and described by the text that follows.

Figure 38-25.PCI Reset Flow



- Avoid any further master accesses on the PCIe bus and discard any completions for the PF.
- Initialize PCIe registers and core registers as listed in Table 39-1.
- Invalidate the LAN queues mapping tables of the PF and its VFs: VFQTABLEs and VSIQTABLEs.
- Invalidate the FPM tables of the PF and its VFs (function private memory).
- Disable all transmit, receive and admin queues. All packets in the 10 GbE controller are lost. EMP packets might be lost as well for a very short period (in the range of 1 µs).
- Invalidate all filters in the FD table and internal caches.
- Invalidate all internal caches: transmit and receive queue contexts and PE Quad Hash filter.
- Invalidate all associated VSI contexts.
- Clear interrupt settings of all functions.
- The following flags are cleared by the 10 GbE controller at the beginning of the reset flow:
  - The CONF_*_DONE bits in GLNVM_ULD register are cleared.
- The 10 GbE controller repeats the following flow for all previous *_*_DONE flags:
  - If the NVM is not valid:
    - Set the CONF_*_DONE flag in GLNVM_ULD register.
  - If the NVM is valid:
    - Load the matched block from the shadow memory.
    - If EMP_*_REQD is set:

      Load any parameters from the alternate structure into the matched block(s)

      Perform any configuration of the matched block(s).

      Set the EMP_*_DONE flag.
    - Set the CONF_*_DONE flag.
  - At this point, the EMP might start responding the Get Version Admin command, which is blocked until this stage.
    - When checking for hardware configuration to complete, software should either wait for a response to a Get Version Admin command or poll on the CONF_*_DONE bits.

Following these steps, hardware is ready to accept operating system configuration cycles.

38.13.1.2.3 PFR Flow

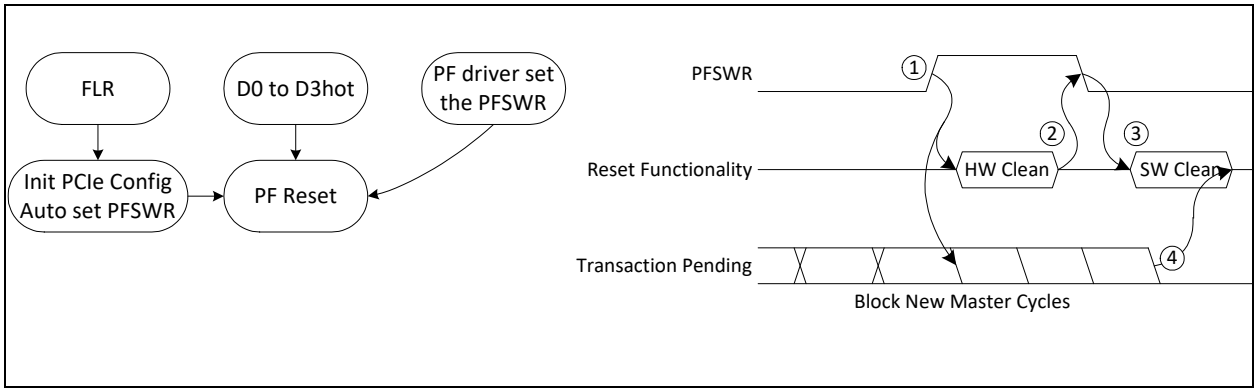
PFR resets a specific PF. For SR-IOV, it also resets the VFs of this PF. The reset flow is shown in Figure 38-26. It is initiated by the PF driver (setting the *PFSWR* bit in the PFGEN_CTRL register) or operating system (setting the *FLR* flag in the Device Control register) or D0 to $D3_{HOT}$ transition.

Before initiating the PFR, software is expected to disable all transmit and receive queues of the PF as described in the following pseudo code:

```
disable_pf_queues() // disable all Tx and Rx queues of the PF

{
```

```
1. Disconnect all Tx and Rx queues from the interrupt link lists as follow

   Set the FIRSTQ_INDX field to 0x7FF in all PFINT_LNKLSTx register of the PF

   Set the FIRSTQ_INDX field to 0x7FF in all VPINT_LNKLSTx register of the VFs of
   the PF

2. last_q = PFLAN_QALLOC.LASTQ - PFLAN_QALLOC.FIRSTQ // "last_q" is a local
   variable used in the steps below. It is the index of the last queue of the PF and
its VFs (in the PF space)

3. Set the SET_QDIS flag in the GLLAN_TXPRE_QDIS registers for all transmit queues
   of the PF and its VFs. The queue indexes are global ones starting by
   PFLAN_QALLOC.FIRSTQ and up to PFLAN_QALLOC.LASTQ.

4. Clear QRX_ENA[register index=0,… last_q]     // Disable all Rx queues of the PF

5. Clear QTX_ENA[register index=0,… last_q]     // Disable all Tx queues of the PF.
   Software must guarantee a gap of at least 400 µs from step 3 to this step

6. Wait for the last Rx queue to be disabled and wait for all Tx queues to be
   disabled or time expires (expiration time is defined by the ENDLESS_XOFF_THRESH
   parameter). Note that if the software is not required to take any special
   actions in case the time is expires and can continue to the next step initiating
   the PFR.

}
```

Figure 38-26.PFR Flow



The PF hardware response to PFR is listed in Table 38.1. Note to the following specific events:

- Emulate internal VFR to all its VFs (the hardware response to VFR is described in Section with the exception that the VF's CSRs are not gated on read).
- Avoid any further master accesses on the PCIe bus and discard any completions for the PF and its VFs.
  - — Once all pending requests of the PF are completed, the *Transaction Pending* bit in the Device Status register in the PCIe configuration space is cleared.
  - — Once all pending requests of each VF of the PF are completed, the *Transaction Pending* bit in the Device Status register the VF PCIe configuration space is cleared (per each VF).
- Disable all transmit receive and admin queues of the PF and its VFs (note that some transmit packets that are already fetched completely might be transmitted).
  - — Disabling the transmit queues might take some time until the 10 GbE controller processes all transmit descriptors that are already fetched. Because many queues are active, this process might take longer. The transmit completion

might be further delayed when the TC is paused by flow control. It can happen if other functions have active transmit queues on the same TC as the function under reset. Hardware includes a timeout mechanism that prevents indefinite latency.

  - Disabling the receive queues might take some time as well until the packets of the function's queues are flushed from the shared receive data path. The completion might be further delayed when there are packets of other functions in the pipe that belongs to no-drop TC with no valid receive descriptors. Hardware includes a timeout mechanism that prevents indefinite latency.

- Clear interrupt settings of the PF and its VFs (excluding the CEQ and LAN transmit and receive cause control registers and any interrupt context in the PCIe configuration space). Write 1b to also clear the VFLRE flags in the *GLGEN_VFLRSTAT* registers of all the VFs owned by the PF.

After all the previous steps are completed, hardware does the following:

- Clears the *PFSWR* bit in the *PFGEN_CTRL* register.
- Releases bus master cycles for the PF and its VFs.

The PF software polls the *PFSWR* bit until it is cleared (indicating that hardware completed its reset flow). On top of it, software should also poll the *Transactions Pending* flag in the PCI configuration space of the PF (indicating that all outstanding requests of the PF were completed).

- Once the *Transactions Pending* flag is cleared, the PF software can release the pinned memory structures
- Once the *PFSWR* bit is cleared as well, the PF software can proceeds to the following steps.

Software Note: If the *PFSWR* bit is not cleared within ~100 ms it can be assumed that the shared data-path inhibits PFR completion. In such a case, software might initiate the CORER to overcome any possible lock.

As part of the initialization flow the PF driver checks, the pending transactions of its VFs (by setting the VF index and the offset of the VF Device Status register in the PF_PCI_CIAA register and then polling the pending flag in the PF_PCI_CIAD register). Once the Transactions Pending is found cleared, the PF software does the following:

- Clears the *VFSWR* flag in the *VPGEN_VFRTRIG* registers of its VFs.
- Sets the VFR_STATE in the *VFGEN_RSTAT* registers of its VFs to VFR completed. At this point, the memory structures of the VF can be released and the VF software can start its initialization flow.

As part of the software/hardware initialization flow, the PF software should check for potential race condition with another PF or EMP initiating a global reset (CORER, GLOBR or EMPR):

- Query the reset counters (*CORERCNT, GLOBRCNT* and *EMPRCNT*) in the GLGEN_RSTAT register
- Interrupt enable flow
- Read the *GLGEN_RSTAT* register for the device state (*DEVSTATE*) and the reset counters (*CORERCNT, GLOBRCNT* and *EMPRCNT*)
  - If *DEVSTATE* = Reset requested or Reset in progress then go to the global reset flow
  - If Device state = Device active then

  - If the updated values of the global reset counters equal to the value sampled at the beginning of this procedure then "all good". Software can continue with the rest of the software/hardware initialization flow
  - Else, then go to the global reset flow

38.13.1.2.4 FLR Flow

FLR resets a specific PF. In the case of SR-IOV, it also resets the VFs of this PF.

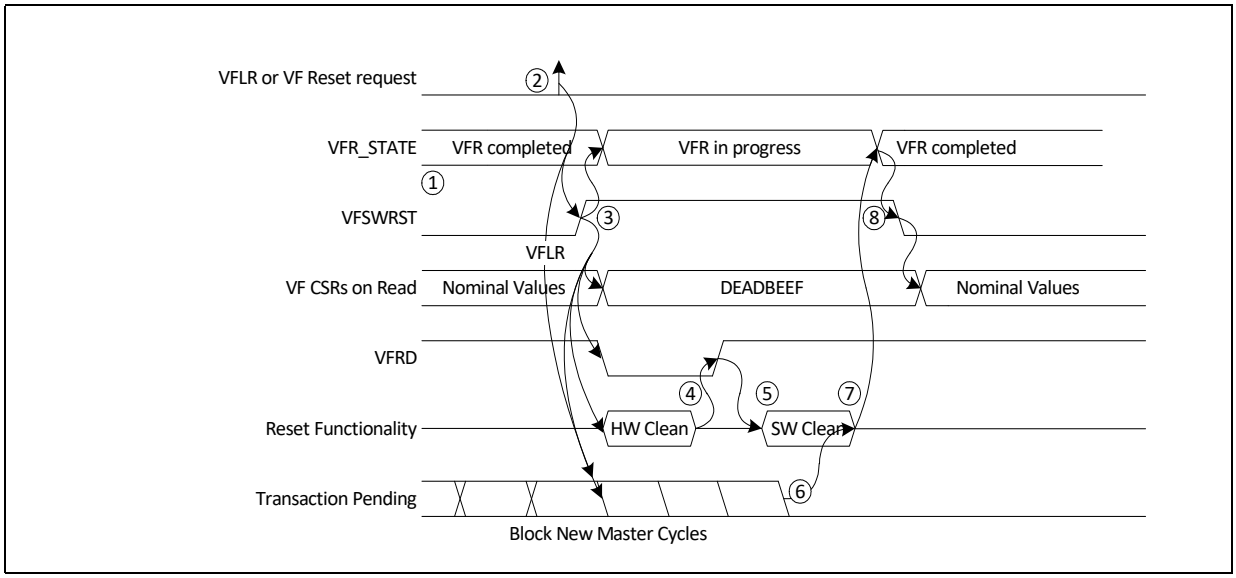
- The following steps are optional:
  - The operating system is expected to clear the BME bit in the Command register in the PCI configuration register of the PF.
  - For SR-IOV, the operating system is expected to also clear the BME bit in the VF Command register in the PCI configuration register of all VFs of this PF.
  - As a response, hardware avoids any new master cycles of the PF and its VFs (including MSI-X initiation).
  - The operating system should poll the Transaction Pending bit in the Device Status register of the PF until it is cleared.
  - For SR-IOV, the operating system should poll also the *Transaction Pending* bit in the VF Device Status register of all VFs of the PF until they are cleared.
  - Note that all the previous steps are expected and recommended but not enforced by the PCI specification.
- The operating system sets the *FLR* bit in the Device Control register of the PF. The operating system is required by PCIe specification to wait 100 ms before it can assume that the FLR sequence is completed by hardware.
- The PF hardware response as follows:
  - Initialize the PCIe configuration space of the PF including clearing the BME bit of the PF and the VFE bit.
    - By clearing the BME, hardware avoids any further master accesses on the PCIe bus and trash any completions for the PF.
    - By clearing the VFE bit, all VFs of the PF avoid any further master accesses on the PCIe bus and discards any completions targeted for these VFs and become hidden on the PCIe bus.
    - As part of the PCIe configuration initialization, the completion timeout is set to its hardware default.
    - Once all pending requests of the PF and its VFs are completed or completion timeout expires (whichever comes first), the *Transaction Pending* bits in the PCIe configuration space are cleared.
  - Auto-set the PFSWR bit in the PFGEN_CTRL register (triggering a PFR described in the section that follows).

Once the *Transaction Pending* bit in the PCIe configuration space of each VF is cleared, the operating system can release all VF memory structures and unload the VF driver (if required). Once the *Transaction Pending* bit in the PCIe configuration space of the PF is cleared, the operating system can release all PF memory structures and unload the PF driver (if required).

38.13.1.2.5 VFR/VFLR Flows

The VF reset flow is shown in Figure 38-27 and detailed in the sections that follow.

Figure 38-27.VF Reset (VFR) Flow



VF Reset Request by the VF Driver

The VF driver requests the VF reset from its parent PF as follows and is shown in Figure 38-27. See also Section that describes the PF response to the VF reset request.

- If needed, clear the RSS tables of VSIs assigned to the VF using *Set RSS Key* and *Set RSS LUT* commands with zero values.
- The VFR_STATE in the VFGEN_RSTAT register in this step is expected to be VFR completed. Note that the VF software gets a control over its VF only after any prior VF reset flow is completed (step 1 in Figure 38-27).
- The VF driver initiates a request to its parent PF to initiate a VF reset. The mechanism for sending this request is outside the scope of this document. It could be done using a VF-to-PF mailbox (Admin command) or any other software based sideband channel (step 2 in Figure 38-27).
- After that, the VF polls the VFR_STATE in the VFGEN_RSTAT register until it is set by the PF to VFR completed (step 8 in Figure 38-27 and explained in Section).
- The VF software proceeds activating the function.

VF Reset Request by the Operating System (VFLR)

The VF reset initiation by the operating system is described as follows and shown in Figure 38-27.

- The following steps are optional:
  - Clear the BME bit in the VF Command register.
  - As a response, hardware avoids any new master cycles of the VF (including MSI-X initiation). Old completions are trashed by hardware.
  - The operating system polls the *Transaction Pending* bit in the VF Device Status register until it is cleared.

- The operating system sets the *FLR* bit in the VF Device Control register (step 2 in Figure 38-27).
- The operating system is required by PCIe specification to wait 100 ms before it can assume that the VFLR sequence is completed by hardware.
- If required, the operating system brings up a new VF (or the same VF). The VF software driver should poll the VFR_STATE in the VFGEN_RSTAT register until it equals to VFR completed (set by the PF software).
- The VF software proceeds activating the function.

Hardware responses to VFLR are as follows:

- Initialize the PCIe configuration space of the VF including the *Bus Master Enable* flag. The *Transaction Pending* bit is cleared when there are no more pending completions.
  - Once the *Transaction Pending* bit in the VF Device Status register is cleared, the operating system can release all VF memory structures and unload the VF driver (if required).
- Set internally the VFSWR bit in the VPGEN_VFRTRIG register of the VF, which initiates a VFR (described in Section).
- Set the matched VFLRE flag in the GLGEN_VFLRSTAT registers and initiate an interrupt to the parent PF. The PF response to the VFLR interrupt is described in Section .

VF Reset Flow by the PF Software Driver

The PF software is called to initiate the VFR by the VF software driver or as a result of a VFLR interrupt. See Figure 38-27.

- Specific step for a VFLR interrupt:
  - The PF software queries the VFLRE flags in the GLGEN_VFLRSTAT registers (of the VFs that it owns).
- Specific step for a VF reset request by the VF driver:
  - PF software sets the *VFSWR* bit in the VPGEN_VFRTRIG register of the VF (step 3 in Figure 38-27).

The hardware response to setting the *VFSWR* bit is listed in Table 38.1. Note that as part of the VF registers, the VFGEN_RSTAT register is cleared as well reflecting VFR in progress state. On top of it, hardware also gates read accesses to the CSRs of the VF returning DEADBEEF or DEADBEAF values and also gates any master accesses.

- When the reset flow is completed, the hardware sets the VFRD flag in the VPGEN_VFRSTAT register.

Once the VFRD flag in the VPGEN_VFRSTAT register is found active, the PF software proceeds with the VF reset flow (step 5 in Figure 38-27).

- Writes 1b to clear, to the matched bit in the GLGEN_VFLRSTAT registers for the VF under reset. Note that the GLGEN_VFLRSTAT registers are composed of 128 bits for the 128 VFs. All PFs have access to the bits of all VFs. However, it is expected that the PFs writes 1b to clear to those bits that match its VFs and only to those bits.
- Disable the receive queues of the VF following the fast queue disable flow per each queue. Note that the hardware auto-disable the transmit queues of the VF.
- Clear the interrupt settings of the VF.
- Clear the queue mapping tables of the VF's VSIs (VSIQTABLEs).

- Clear the filters in the FD table that were assigned by the PF for the VF.
- Remove all the MAC/VLAN filters from the VSIs of the VF and then remove these VSIs.
- If needed, clear the RSS tables of VSIs assigned to the VF using *Set RSS Key* and *Set RSS LUT* commands with zero values.
- The PF driver checks the pending transactions of its VF (by setting the VF index and the offset of the VF Device Status register in the PF_PCI_CIAA register and then polling the pending flag in the PF_PCI_CIAD register) (step 6 in Figure 38-27).
- The following steps in this bullet item are part of re-enabling the VF. It can be executed at this phase before notifying the VF that the reset flow is completed or at a later phase (depending on software implementation):
  - Add the VSIs for the VF (including its Transmit and Receive queues and its Scheduler).
  - Add the MAC/VLAN filters for the VSI.
  - Enable the VFLAN_QTABLE by setting the VPLAN_MAPENA and then program the VFLAN_QTABLE to the queues of the VF.
- The PF software completes the flow by notifying the VF that the reset flow is completed. It sets the VFR_STATE in the VFGEN_RSTAT register to VFR completed and clears the *VFSWR* bit in the VPGEN_VFRTRIG register (step 7 in Figure 38-27).

Hardware response to cleared *VFSWR* flag:

- The VF CSRs are unlocked. As a result, the VF software can see the updated VFR_STATE in the VFGEN_RSTAT register that equals to VFR completed (step 9 in Figure 38-27).
- Master cycles are not blocked anymore by the reset logic.
- The hardware is ready to be used by the VF.

38.13.1.2.6 VMR Flow

Before initiating a VM reset the PF software should disable the interrupts associated with this VM. The VM interrupt causes might share the same interrupt with the PF. In this case the PF software need only to remove the VM interrupts causes from those interrupt linked list. Then the PF software can initiate the VM reset and re-enable the interrupts of the PF.

The PF RDMA driver should also clear all the QH entries associated with this VM and disable PE enable for this VM. PE support may be re-instated after the reset.

The PF sets the *VMSWR* bit in the VSIGEN_RTRIG register:

- Hardware response to VMR is the same as its response to VFR (other than invalidating the VFQTABLE's and VFxxx registers, which the VM does not own).

The PF polls the VMR done indication in the VSIGEN_RSTAT register and then executes the following complementary steps (as the VFR flow):

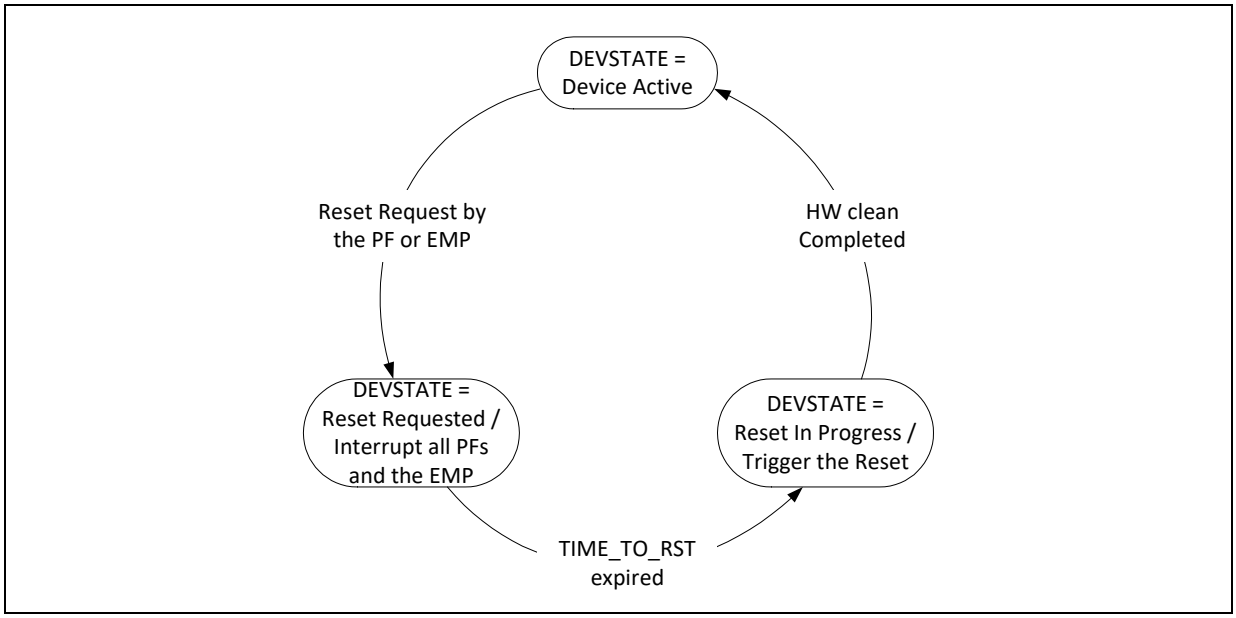
- Disable the transmit and receive queues using the fast queue disable flow.
- Optionally clear the queue mapping tables, FD filters and VSI context.
- The PF polls the *Transactions Pending* flag of the VM verifying that there are no transaction pending of the VM as follows: Set the VSI index in the PFPCI_VMINDEX and then poll the PFPCI_VMPEND register.

- The PF completes the flow by clearing the *VMSWR* bit in the VSIGEN_RTRIG register.

38.13.1.2.7 Core Global and EMP Reset Flows

The global resets can be initiated by the PF software or the EMP firmware. It is expected to be used as a mechanism to resolve potential hardware locks or synchronization lose, bringing the 10 GbE controller to a known functional state. These global resets impact all PFs (and their VFs) as well as the EMP subsystem (EMP reset only). Therefore, graceful flow is enabled by hardware as shown in Figure 38-28 and described in the following sections. The software initiates the global resets by the GLGEN_RTRIG register. The EMP can access the same register or use an internal register.

Figure 38-28.Global Resets Flow



Software and Firmware Interface

GLOBR / CORER — Setting the GLOBR flag or CORER flag in the GLGEN_RTRIG register triggers a graceful global/core reset, respectively.

EMPFWR — EMP firmware watchdog expiration or EMP setting internal EMP reset flag, triggers a graceful EMP reset (EMPR).

DEVSTATE — Global device state exposed to all PFs in the GLGEN_RSTAT registers. The DEVSTATE can be at one of the following states: device active, reset requested; reset in progress.

RESET_TYPE — The RESET_TYPE reflects the last reset cause initiated to the 10 GbE controller. It changes its state once any of the following reset sources is triggered. The RESET_TYPE can be at one of the following states: POR, CORER, GLOBR and EMPR.

RST_CNT — The GLGEN_RSTAT reflects the following counters: CORERCNT, GLOBRCNT and EMPRCNT. These fields are 2-bit counters each. They count the matched reset completion events since POR.

CORER Flow

The PF software or EMP firmware initiates a graceful core reset by setting the CORER flag in the GLGEN_RTRIG register and polling the 10 GbE controller state in the GLGEN_RSTAT register.

Hardware response:

- Changes the GLGEN_RSTAT register are as follows:
  - Set DEVSTATE to reset requested.
  - Set RESET_TYPE to CORER indicating the requested reset.
  - Set TIME_TO_RST by the GLGEN_RSTCTL.GRSTDEL value and start counting down.
- Initiates a GRST interrupt to all PFs and EMP that the reset is about to be fired by hardware.
- The following flags are cleared by the 10 GbE controller at the beginning of the reset flow:
- The CONF_*_DONE bits in GLNVM_ULD are cleared.
- Clear the entire transmit and receive data path, avoid any further master accesses on the PCIe bus and discard any completions for the entire device, and abort any transmission in progress (including packets sent by the EMP).
- Reset internal device logic excluding EMP cluster, PCI interface and MAC/PHY cluster as listed in Table 38-1.
- Once the reset flow is completed, update the GLGEN_RSTAT register as follows:
  - Set DEVSTATE to device active.
  - Increment the CORERCNT by one.
- The 10 GbE controller repeats the following flow for all previous *_*_DONE flags:
  - If the NVM is not valid:
    - Set the CONF_*_DONE flag in GLNVM_ULD register.
  - If the NVM is valid:
    - Load the matched block from the shadow memory.
    - If EMP_*_REQD is set:

      Load any parameters from the Alternate Structure into the matched block(s).

      Perform any configuration of the matched block(s).

      Set the EMP_*_DONE flag.
    - Set the CONF_*_DONE flag.
  - At this point, the EMP might start responding the Get Version Admin command, which is blocked until this stage.

Following the GRST interrupt, all PFs and the EMP poll the 10 GbE controller state in the GLGEN_RSTAT register.

- Keep polling the register as long as DEVSTATE is not equals to device active. Note that as long as DEVSTATE equals to reset requested, the TIME_TO_RST indicates the remaining delay to the internal reset triggering by hardware. Note the PF drivers should avoid any CSR slave accesses other than the one required to query the 10 GbE controller state.

- The GLGEN_RSTAT also indicates the number of the initiated global resets via the GLGEN_RTRIG register (CORER, GLOBR, EMPR). These counters might be needed in case a function initiated consecutive global reset before all other functions had the chance to realized that the previous reset was completed. It would be good practice to avoid too frequent global resets to avoid such cases.
- Check the RESET_TYPE that indicates the reset that was initiated and follow the required initialization flow.
  - Note that also the function that initiated the reset should check the RESET_TYPE since it might reflect a stronger reset initiated by another function.
- Check that the hardware completed to auto-load its settings from the NVM shadow and the alternate structure by polling the CONF_*_DONE flags in GLNVM_ULD register.
  - When checking for the hardware configuration to be done, software should either wait for a response to a Get Version Admin command or poll on the CONF_*_DONE bits.
- The PFs proceeds with their software initialization flow.

GLOBR Flow

Global reset is initiated by the PFs or the EMP by setting the GLOBR flag in the GLGEN_RTRIG. The GLOBR flow is identical to the CORER flow with the following changes:

- GLOBR also initializes the MAC/PHY units.
- GLGEN_RSTAT.RESET_TYPE is set by the hardware to GLOBR (rather than CORER).
- Increment the GLOBRCNT by one (rather than CORERCNT).
- Loading the 10 GbE controller setting from the NVM is listed by the GLOBR column in Table 38-2.

EMPR Flow

EMP reset is expected to be used by the EMP as a mechanism to resolve potential hardware locks or potential lost of synchronization between the firmware and hardware that are not expected to be resolved by CORER nor by GLOBR. The EMPR impacts also all PFs and their VFs, therefore the following graceful flow is recommended.

The EMP flow is identical to the CORER flow with the following changes:

- During normal operation, the EMPR can be initiated only by the EMP by setting an internal EMP reset flag.
- EMPR initializes also the MAC/PHY units as well as the EMP cluster.
- GLGEN_RSTAT.RESET_TYPE is set by hardware to EMPR (rather than CORER).
- Increment the EMPRCNT by one (rather than CORERCNT).
- Loading the 10 GbE controller setting from the NVM is listed by the EMPR column in Table 38-2.

38.13.1.2.8 Auto-Load Shadow RAM

This phase of auto load determines if a properly-configured Flash device is attached:

- If the attached Flash device is not properly configured, the default hardware settings are kept. The power-on flow continues without loading from the NVM.

  - The NVM is expected to be reprogrammed. See Section 38.11.8.2 for more details on how this is done.
- If the attached Flash device is properly configured, auto load proceeds with its first stage. NVM modules loaded to the on-die shadow RAM are read from the NVM.

Once the shadow RAM is ready, the *GLNVM_GENS.FL_AUTO_RD* bit is set. The bit is also set when the NVM is found blank.

38.13.1.2.9 Auto Load Into Device Units

This stage loads the NVM configuration into the device units, either directly from the NVM or indirectly through shadow RAM.

Loading the following NVM modules is tracked and reported as:

- PHY analog
- PCIR registers auto load
- PCIe Transaction Layer (TL) shared
- PCIe ALT auto load
- CORER registers auto load
- GLOBR registers auto load
- POR registers auto load
- EMPR registers auto load

The following CSR fields track the progress of loading the NVM modules following power on and the various resets:

- GLNVM_GENS.FL_AUTO_RD, when set, indicates that the shadow RAM was loaded from the NVM and is ready for use.
- EMP_*_REQD bits in the GLNVM_EMPRQ register (one bit per relevant module) indicate whether the EMP involvement is required during the initialization of the module.
- GLNVM_ULD.CONF_*_DONE bits (one per module previously identified), when set, indicate that the respective units are initialized and ready for use. These bits are identical to the EMP_*_DONE bits.

Default hardware values:

- GLNVM_GENS.FL_AUTO_RD = 0b
- GLNVM_EMPRQ.EMP_*_REQD = 0b
- GLNVM_EMPLD.EMP_*_DONE = 0b
- GLNVM_ULD.CONF_*_DONE = 0b

The flow during a power-on stage is as follows.

- If the NVM is found blank:
  - All GLNVM_ULD.CONF_*_DONE bits are set by the 10 GbE controller.
- If the NVM is found valid:
  - GLNVM_GENS.FL_AUTO_RD is set by firmware when the shadow RAM was loaded from the NVM.
  - GLNVM_EMPRQ.EMP_*_REQD is loaded from the NVM.

— After EMP firmware loads, during its initialization sequence, it sets the GLNVM_EMPLD.EMP_*_DONE bit (for each of the CORER and GLOBLR domains). As a result, hardware sets the GLNVM_ULD.CONF_*_DONE bit to 1b.

38.13.1.2.10 Firmware Initialization

There are two sets of firmware loaded for the 10 GbE controller. The first is for the EMP. This firmware is first to load and is required for all Product Name deployments. It is optionally followed by loading PE firmware required for RDMA and UDA operation.

Once EMP firmware loads, the following Product Name features are enabled:

- The external Ethernet link is active.
- Default internal switching components have been configured.
- Communication with the BMC is possible if supported by the 10 GbE controller and the system (optional).
- OEM specific manageability agents are active and responding to commands from the Ethernet fabric (optional).
- Admin queues for all enabled PCI functions are ready for commands. The EMP responds to each function's Get Version AQ command to indicate that it is safe for software to start using the 10 GbE controller for software device driver initialization (see Section 38.13).
- HMC default profile has been configured.

Once EMP firmware is up and running, the 10 GbE controller can be configured via its management interfaces, and certain device capabilities are then enabled or disabled (such as soft SKUing).

PE firmware is responsible for processing configuration requests for RDMA and UDA functionality that is provided by the PE. PE firmware load begins once the EMP firmware has finished loading along with initialization, and only after primary reset was de-asserted. Loading PE firmware is initiated through the GLPE_FWLDCTRL register. When PE firmware is loaded and initialized, the GLPE_CPUSTATUS0-2 registers indicate that it is safe to start using PE capabilities.

PE firmware is loaded only when the PE is enabled in the GLGEN_PE_ENA.PE_ENA CSR bit, loaded from the NVM.

38.13.1.2.11 MAC Address Initialization

Two sets of station LAN MAC addresses are supported:

- A station MAC address per PF. These addresses are used by the internal switch for L2 filtering of Rx packets, by the Rx classification filters, and for WoL purposes. The addresses are loaded from the NVM.
  - The station MAC addresses for WoL are loaded by the EMP into the *PRTPM_SAL* and *PRTPM_SAH* registers (see Section 38.17.3).
    - Default — A single PF address is loaded per port into *PRTPM_SAL[0,Port]* and *PRTPM_SAH[0,Port].*
- A station MAC address per Ethernet port. These addresses are used for link-level functionality such as flow control frames.
  - The *PRTGL_SAL* and *PRTGL_SAH* registers contain these addresses for the four 10 GbE MACs.

— The *PRTMAC_HSEC_CTL_TX_SA_PART1* and *PRTMAC_HSEC_CTL_TX_SA_PART2* registers contain these addresses for the 40 GbE MACs.

The following table lists how each address can be set or read and where it is stored:

MAC Address Type	Where Stored	How Reported	How Modified
LAN MAC Address (Factory)	NVM PF allocations section		N/A
LAN MAC Address (Current)	NVM PF allocations section; alternate RAM	Manage MAC address read (PF LAN SA)	Write alternate AQ command: Alternate LAN MAC address (LS). Alternate LAN MAC address (MS). or NC-SI Set Address OEM command or manage MAC address write (update LAA only).
Port MAC Address	GLOBR registers auto-load module (PRTGL_SAL/H)	Manage MAC address read (port SA)	Manage MAC address write AQ command (update port address). ***Note:*** Every function within a port is allowed to use this command – the last command takes precedence.
LLDP MAC Address	Use the port MAC address for SA, chassis and port IDs.		
WoL MAC Address	NVM PF allocations section, alternate RAM (PRTPM_SAL/H)	Manage MAC address read (PF WoL SA)	Manage MAC address write AQ command (update LAA and WoL address).
WoL MAC Address Preserve on PFR	RAM	Manage MAC address read (WoL_preserve_on_PFR)	Manage MAC address write AQ command (WoL_preserve_on_PFR). Set by software before D0->Dr/D3 transition, cleared by software after returning to D0). Zero by default after POR/EMPR.
LAA MAC Address	GLOBR registers (PRTGL_SAL/H)	Manage MAC address read (PF LAA SA)	Manage MAC address write AQ command (update LAA address). ***Note:*** Changing the LAA only does not impact the address used to detect magic packets.

The following rules apply:

- The *PF_NUM* field in *PRTPM_SAH* is deducted from the PF issuing the command.
- The *MC_MAG_EN* field in *PRTPM_SAH* is loaded from the NVM and is not affected by the command.
- The *AV* field in *PRTPM_SAH* is set by the EMP when writing a MAC address.

The per-PF MAC address used by the internal switch is managed via the Remove MAC, VLAN Pair and Add MAC, VLAN Pair commands.

Manage MAC Address Read Command

This command is used by the PF driver to read the per-PF station MAC address.

Indirect command:

Name	Bytes.Bits	Value	Remarks
Flags	0-1		See Section 38.27.5.2.1 for details.
Opcode	2-3	0x0107	Command opcode.
Datalen	4-5	0x00	Must be 0x0, value is ignored.
Return Value/ VFID	6-7	0x00	Return value. Zeroed by the software device driver. Written by firmware.

Name	Bytes.Bits	Value	Remarks
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Command Flags	16-17	Reserved	Zeroed by the software device driver.
Reserved	18-23	0x0	Reserved.
Data Address High	24-27	Buff Addr	High bits of return buffer address.
Data Address Low	28-31	Buff Addr	Low bits of return buffer address.

Manage MAC Address Read Response

A firmware acknowledge to the Manage MAC Address Read command.

Indirect response:

Name	Bytes.Bits	Value	Remarks
Flags	0-1		See Section 38.27.5.2.2 for details.
Opcode	2-3	0x0107	Command opcode.
Datalen	4-5	0x00	Must be 0x0, value is ignored.
Return Value/ VFID	6-7	Return Value	Return value. Firmware supplies in the *Return Value* field indication on the completion of the Manage LAA command. 0x0 = No error. Others = Error detected in the command.
Cookie High	8-11	Cookie	Opaque value is copied by firmware from the manage MAC Address Read command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware from the manage MAC Address Read command.
Command Flags	16.0-16.3	Reserved	Zeroed by the EMP.
	16.4	LAN Address Valid	
	16.5	SAN Address Valid	
	16.6	Port Address Valid	
	16.7	WoL Address Valid	
	17.0	MC_MAG_EN	If set, multicast magic packets generate a WoL event (if WoL is enabled for this function).
	17.1	WoL_preserve_on_PFR	Set to preserve WoL MAC on PFR (set by software before D0->Dr/D3 transition, cleared by software after returning to D0 state). Zero by default after POR/EMPR.
	17.2-7	Reserved	
Reserved	18-23	0x0	Reserved.
Data Address High	24-27	Buff Addr	High bits of return buffer address.
Data Address Low	28-31	Buff Addr	Low bits of return buffer address.

Note: All MAC addresses are in big endian order.

Address	Offset	Description
PF LAN SA	0-5	Current device value of the PF LAN MAC address. Validated by *LAN Address Valid* flag. This address is returned from LAA address if valid; otherwise, from alternate RAM if valid; otherwise, from the NVM if valid.
Port SA	12-17	Current device value of the Port MAC address. Validated by the *Port Address Valid* flag.
PF WoL SA	18-23	Current device value of the PF WoL MAC address. Validated by the *WoL Address Valid* flag.

Manage MAC Addresses Write Command

This command is used by the PF driver to write the per-PF station MAC address.

Direct command:

Name	Bytes.Bits	Value	Remarks
Flags	0-1		See Section 38.27.5.2.1 for details.
Opcode	2-3	0x108	Command opcode. 0x0107 is used for reads; 0x0108 is used for writes.
Datalen	4-5	0x00	Must be 0x0, value is ignored.
Return Value/ VFID	6-7	0x00	Return value. Zeroed by the software device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Command Flags	16-16.7	Reserved	Zeroed by driver.
	17.0	MC_MAG_EN	Set to allow multicast magic packets generates a WoL event (if WoL is enabled for this function). Valid only if Write Type = 11.
	17.1	WoL_preserve_on_PFR	Set to preserve WoL MAC on PFR (set by software before D0->Dr/D3 transitions, cleared by software after returning to D0 state). Zero by default after POR/EMPR.
	17.2-5	Reserved	
	17.7:6	Write Type	00b = Update LAA only. 01b = Update LAA and WOL address. 10b = Reserved 11b = Reserved.
SAH	18-19	See Remarks	High 16 bits of the MAC address (big endian order). Example: if MAC address = 11:22:33:44:55:66 then SAH = 0x1122.
SAL	20-23	See Remarks	Low 32 bits of the MAC address (big endian order). Example: if MAC address = 11:22:33:44:55:66 then SAL = 0x33445566.
Data Address High	24-27	Buff Addr	High bits of return buffer address.
Data Address Low	28-31	Buff Addr	Low bits of return buffer address.

Manage MAC Address Write Response

A firmware acknowledge to the Manage LAA command.

Direct response:

Name	Bytes.Bits	Value	Remarks
Flags	0-1		See Section 38.27.5.2.2 for details.
Opcode	2-3	0x0108	Command opcode.
Datalen	4-5	0x00	Must be 0x0, value is ignored.
Return Value/ VFID	6-7	Return Value	Return value. Firmware supplies in the *Return Value* field indication on the completion of the Manage LAA command. 0x0 = No error. Others = Error detected in the command.
Cookie High	8-11	Cookie	Opaque value is copied by firmware from the manage MAC Address Write command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware from the manage MAC Address Write command.
Command Flags	16-17.5	Reserved	Zeroed by the EMP.
	17.7:6	Write Type	00b = Update LAA only. 01b = Update LAA and WOL address. 10b = Update port address. 11b = Reserved.
Reserved	18-23	0x0	Reserved.
Data Address High	24-27	Buff Addr	High bits of return buffer address.
Data Address Low	28-31	Buff Addr	Low bits of return buffer address.

38.13.1.2.12 Power-on Device State

This section describes the specific setting of each of the 10 GbE controller's components required for pre-boot operation. It describes the information loaded from the NVM (per component) and the state attained by each.

- **Manageability** — System management functionality, if enabled, is fully operational by the end of the EMP firmware initialization sub-stage. Some configuration is loaded from the NVM during the power on stage. Capabilities might include the following:
  - — Sideband interfaces
  - — Pass-through manageability (including packet filtering)
  - — Preparation for OS-to-BMC traffic
  - — Preparation for MCTP over PCIe operation
- **Internal MAC and PHY** — If either system management or APM WoL are enabled, then enabled LAN ports are brought up by firmware once EMP firmware initialization completes. Else, enabled LAN ports are brought up following PERST# de-assertion.
  - — See Section 38.13 for more detail.
- **AQ** — A queue (Tx and Rx pair) is activated per enabled PCI function. For example, an AQ is needed for BIOS to change the switch or scheduler configuration.
  - — See Section 38.13 for more detail.
- **Internal Switch** — The internal switch is configured from the NVM for basic switching capabilities to the EMP and the enabled PCI functions. First, the switch

programmable logic is loaded, followed by configuration of a basic switch topology. The topology is for basic L2 functionality.

— See Section 38.21.7.17.2 for more details.

- **DCB** — The NVM loads the LLDP and DCBx setting into the 10 GbE controller. Once EMP firmware initializes and link is up, firmware engages in DCBx negotiation. Firmware then makes the appropriate changes in the 10 GbE controller's configuration based on the outcome of the DCBx protocol. At this stage, DCB capabilities (TCs, ETS, PFC) might be enabled.
- **Tx Scheduler** — As the switch VSIs are enabled, firmware allocates Tx scheduler queue sets per each PF VSI based on the NVM configuration. Each queue set is configured to default behavior. Firmware also generates the required handles to enable system software to update the configuration of each queue set. If DCBx protocol runs, the outcome of the protocol exchange might translate into changes in scheduler configuration.

  — See Section 38.25.4.1 for more details.
- **Host Memory Cache (HMC)** — NVM settings supply the initial HMC configuration using a simple set of rules that enable equal distribution of HMC resources to all PFs that are connected to external Ethernet ports.
- **Power Management** — Some power management capabilities are supported pre-boot or during BIOS initialization.

  — The 10 GbE controller might be enabled for APM wake during power up. See Section 38.18.1 for more details on how APM Wake is configured.
- **LAN** — LAN queues are available for network boot in the BIOS initialization phase. Some LAN configuration is loaded from the NVM during power on, including partitioning of the LAN queues among PFs.
- **PE** — An NVM setting might disable PE operation in the device. The PE is not enabled during power on and BIOS initialization stages. It is configured and initialized in the driver load phase.

38.13.2 BIOS Initialization

This section describes how BIOS code might update the 10 GbE controller's configuration and the capabilities for network boot. The following sections are provided:

- How the BIOS code might change the 10 GbE controller's configuration on each boot.
- Some aspects of supporting network boot.
- The specific setting of each of the 10 GbE controller's components required for the BIOS Init stage.

38.13.2.1 Initial State

The state of the 10 GbE controller when the BIOS begins to access it is described later in this section.

BIOS code must check that the EMP completed device initialization. This is done through the Get Version AQ command described in Section 38.27.11.1.

38.13.2.2 Non-Persistent Configuration

During power up, the 10 GbE controller is factory configured from the NVM. However, the factory configuration might be overridden in two possible ways:

- Persistent configuration — System tools (such as software agents and SMCLP commands) might write into the NVM and change the factory defaults. It is also possible to add alternate values into the NVM so that the factory defaults are preserved and might be restored at a later time.
- Non-persistent configuration — An on-die alternate RAM structure is provided for alternate configuration that is not maintained between cold resets. During a cold reset sequence, the alternate structure might be written by either an external system management agent (via the device management interfaces) or by BIOS level code (like SMASH/CLP commands). The information in the alternate structure is kept during any reset other than cold reset, and is loaded into the 10 GbE controller following the appropriate resets (see the text that follows for more detail), overriding the respective configuration loaded previously from the NVM. This section describes the details of how the alternate structure is handled.

The following mechanisms are provided:

- The configuration is written into an on-die alternate structure:
  - The alternate structure is 8 KB, partitioned into 32-bit entries. Accessing the structure is done by addressing 32-bit entries. An address is therefore 11 bits (2 K Dwords), where address 0x000 points to the beginning of the 8 KB memory.
- The alternate structure is loaded into the device functional units in the following cases:
  - As part of the SMASH/CLP programming:
    - Legacy BIOS — Once the Initialization function is called for the 1st time in the flow.
    - UEFI — Once a Done Alternate Write command is received for each enabled LAN port.
  - As part of executing the following resets: PERST#, PCIR, EMPR, global reset, core reset. In other words, the contents of the alternate structure are only reset on cold resets.
- SMASH/CLP configuration might not be followed by a PCIe reset, and designers cannot rely on a PCIe reset to load the alternate module into the 10 GbE controller.
- Option ROM code and UEFI drivers communicate with the 10 GbE controller via a set of AQ commands:
  - The Read Alternate/Direct and Read Alternate — Indirect commands read from the alternate structure.
  - The Write Alternate/Direct and Write Alternate — Indirect commands write into the alternate structure.
  - The Done Alternate Write command signals the 10 GbE controller that the CLP strings phase completed for the entire device in Legacy BIOS mode and per LAN port in UEFI mode.

38.13.2.2.1 Alternate RAM Structure

The alternate structure is 8 KB, partitioned into 32-bit entries. Accessing the structure is done by addressing 32-bit entries. An address is therefore 11-bits (2 K Dwords), where address 0x000 points to the beginning of the 8 KB memory.

The structure is partitioned as follows:

Section	Address (Dword)	Size (Dword)	Content
Per PF	0 – 1023	1024 - 64 per PF	16 per-PF sections, one per PF. These sections are described in Section and can be accessed by each PF.
EMP	1024 - 1279	256	Reserved for EMP use, including error logging. Can be written and read only by the EMP. In debug mode, this section can be read by the PFs.
Reserved	1280 - 2047	768	Reserved.

Per PF Sections

Table 38-92. Per PF Alt RAM Content

Scope	Address (Dword)	Contents	Used By	PCIe ALT Module in Shadow RAM	GLOBR Registers Auto load	CORER Registers Auto load
PF	0	Current LAN MAC Address (LS)	HW, SW			X
PF	1	Current LAN MAC Address (MS)	HW, SW			X
Reserved	17-2	Reserved	N/A			
PF	18	SRIOV	HW	X		X
Reserved	63-19	Reserved	N/A			

Each entry is associated with one or more NVM modules and is loaded into the 10 GbE controller following reset events that load these modules. Loading into the PCIe units is an exception. Parameters to be loaded into PCIe units are written by the EMP into the PCIe ALT auto-load module in shadow RAM. This is done as part of the BIOS initialization flow. On later resets, the 10 GbE controller automatically loads the module into the PCIe units, saving the need for the EMP to intervene.

The PCIe ALT auto-load module is a regular hardware type 1 auto-load section as described in Section 38.19.2.1.1. The content is dynamically created according to the registers that requires an auto-load value different than the default value or the value loaded from the NVM.

Current LAN MAC address (offset 0x0, 0x1)

Lower Dword:

Field	Bit(s)	Description
MAC Address	31:0	MAC Address. Contains the LS 32-bit of the address.

Upper Dword:

Field	Bit(s)	Description
MAC Address	15:0	MAC Address - Contains the MS 16-bit of the address.
Reserved	30:16	Reserved.

Field	Bit(s)	Description
Valid	31	Valid Bit. 0b = The MAC address two Dwords are invalid and should be skipped. 1b = The MAC address two Dwords are valid and should be processed.

Upper Dword:

Actions taken on a change in this entry:

- When valid, it overrides the MAC address loaded from the NVM.

SRIOV (offset 0x12)

This section describes the SRIOV per PF alternate RAM content. The table that follows lists the specifics.

Field	Bit(s)	Description
SRIOV VFs	6:0	Number of SR_IOV VFs requested for the function. Can be any number between 0 and 127. Valid only when the *Auto* field is set to 0b.
Reserved	28:7	Reserved.
Enable	29	SR-IOV is enabled (global configuration - valid only in function 0).
Valid	31	Valid Bit. 0b = Dword is invalid and should be skipped. 1b = Entry is valid and should be processed.

- If the *Enable* bit is cleared (in function 0):
  - Write to the PCIe alternate auto-load section, GLPCI_CAPSUP.SRIOV_EN = 0.b.

- If the *Enable* bit is set (in function 0):
  - Write to the PCIe alternate auto-load section, GLPCI_CAPSUP.SRIOV_EN = 1b and GLPCI_CAPSUP.ARI_EN = 1b.

The EMP calculates the number of interrupt vectors requested per VF through the GLPCI_CNF2.MSI_X_VF_N registers (done per enabled VF):

- If the value in the GLPCI_CNF2.MSI_X_VF_N field equals the maximum number of vectors using the original number of VFs), the EMP updates MSI_X_VF_N to equal the number calculated along with the update number of enabled VFs. This is the case where the device requests the maximum number of possible vectors.
- Else, MSI_X_VF_N is not changed. This is the case where the device requests less than the maximum number of possible vectors.
- The EMP updates the following registers in shadow RAM PCIe ALT auto-load module:
  - PF_VT_PFALLOC_PCIE
  - GLGEN_PCIFCNCNT
  - GLPCI_CNF2
- The EMP updates the following register fields so hardware is updated immediately:
  - PF_VT_PFALLOC
  - GLGEN_PCIFCNCNT.PCIVFCNT
  - GLPCI_CNF2.MSI_X_VF_N

- As part of each internal CORER flow, the EMP updates the following registers:

— PF_VT_PFALLOC

— GLGEN_PCIFCNCNT_INT

- The Device/Function Capabilities command's *Virtual Function* field reflects the change in allocations.
- Update the number of VSIs and MAC addresses allocated to each PF to be two for each PF (LAN) and one for each VF allocated to this PF. The new allocation should be stored in the SR PF allocation section.

Note: After receiving a Done Alternate Write command, firmware checks the total number of VFs allocated is not larger than 128, and if it is larger, returns an ENOSPC error.

Offset 0x15

Field	Bit(s)	Description
Reserved	30:0	Reserved.
Valid	31	Valid Bit. 0b = Entries 25-22 are invalid and should be skipped. 1b = Entries 25-22 are valid and should be processed.

38.13.2.2.2 AQ Commands

The Table 38-93 lists the different AQ commands used to manage the alternate structure.

Table 38-93. List of AQ Commands for the Alternate Structure

Command	Opcode	Brief Description
Write Alternate - Direct	0x0900	Write up to two parameters into the alternate structure.
Write Alternate - Indirect	0x0901	Write a block of parameters into the alternate structure.
Read Alternate - Direct	0x0902	Read up to two parameters from the alternate structure.
Read Alternate - Indirect	0x0903	Read a block of parameters from the alternate structure.
Done Alternate Write	0x0904	Indication that all CLP strings (for the entire 10 GbE controller in legacy BIOS mode and per LAN port in UEFI mode) have been sent to the 10 GbE controller.
Clear Port Alternate	0x0906	Clear content of alternate RAM relevant to this port.

Write alternate - direct

The Write Alternate - Direct command writes to the alternate structure up to two parameters.

Table 38-94. Write Alternate - Direct Command (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1		See Section 38.27.5.1.1 for details.
Opcode	2-3	0x0900	Command opcode.
Datalen	4-5	0x00	N/A
Return Value/ VFID	6-7	0x00	N/A
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.

Table 38-94. Write Alternate - Direct Command (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
First Parameter Address	16-19		Address should be within the range allocated to the function inside the alternate structure. Accessing the alternate structure is done by addressing 32-bit entries.
First Parameter Data	20-23		
Second Parameter Address	24-27		Address should be within the range allocated to the function inside the alternate structure. Value of 0xFF..FF means only the first parameter is written. Accessing the alternate structure is done by addressing 32-bit entries.
Second Parameter Data	28-31		

Table 38-95. Completion For The Write Alternate - Direct Command

Name	Bytes.Bits	Value	Remarks
Flags	0-1		See Section 38.27.5.2.2 for details.
Opcode	2-3	0x0900	Command opcode.
Datalen	4-5	0x00	N/A
Return Value/ VFID	6-7		Some comments on specific errors: 0x0 = No error. ENOMEM - Out of memory (access outside the alternate structure). EACCES - Permission denied (access to another PF's area).
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Reserved	16-31		Might contain the values sent in the original command.

Write Alternate - Indirect

The Write Alternate - Indirect command writes a block of parameters to the alternate structure. The command defines the number of Dwords to be written and the starting address inside the alternate structure.

Table 38-96. Write Alternate - Indirect Command

Name	Bytes.Bits	Value	Remarks
Flags	0-1		See Section 38.27.5.2.1 for details.
Opcode	2-3	0x0901	Command opcode.
Datalen	4-5		Size of buffer accompanying the command (in bytes).
Return Value/ VFID	6-7	0x00	N/A
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Alternate Structure Address	16-19		Lowest address to be written into the alternate structure. Accessing the alternate structure is done by addressing 32-bit entries.
Alternate Structure Length	20-23		Number of Dwords to be written into the alternate structure.
Data Address High	24-27	Buff Addr	High bits of buffer address.
Data Address Low	28-31	Buff Addr	Low bits of buffer address.

The following completion is sent for the Write Alternate - Indirect command:

Table 38-97. Completion For The Write Alternate - Indirect Command

Name	Bytes.Bits	Value	Remarks
Flags	0-1		See Section 38.27.5.2.2 for details.
Opcode	2-3	0x0901	Command opcode.
Datalen	4-5	0x00	N/A
Return Value/ VFID	6-7		Some comments on specific errors: 0x0 = no error. ENOMEM - Out of memory (access outside the alternate structure). EACCES - Permission denied (access to another PF's area).
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Reserved	16-31		Reserved.

Read Alternate - Direct

The Read Alternate - Direct command reads from the alternate structure up to two parameters.

Table 38-98. Read Alternate - Direct Command

Name	Bytes.Bits	Value	Remarks
Flags	0-1		See Section 38.27.5.2.1 for details.
Opcode	2-3	0x0902	Command opcode.
Datalen	4-5	0x00	N/A
Return Value/ VFID	6-7	0x00	N/A
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
First Parameter Address	16-19		Address should be within the range allocated to the function inside the alternate structure. Accessing the alternate structure is done by addressing 32-bit entries.
Reserved	20-23	0x00	Reserved.
Second Parameter Address	24-27		Address should be within the range allocated to the function inside the alternate structure. Value of 0xFF..FF means only the first parameter is read. Accessing the alternate structure is done by addressing 32-bit entries.
Reserved	28-31	0x00	Reserved.

The following Completion is sent for the Read Alternate - Direct command:

Table 38-99. Completion For The Read Alternate - Direct Command (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1		See Section 38.27.5.2.2 for details.
Opcode	2-3	0x0902	Command opcode.
Datalen	4-5	0x00	N/A
Return Value/ VFID	6-7		Some comments on specific errors: 0x0 = no error. ENOMEM - Out of memory (access outside the alternate structure). EACCES - Permission denied (access to another PF's area).
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.

Table 38-99. Completion For The Read Alternate - Direct Command (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
First Parameter Address	16-19		Copied from command.
First Parameter Data	20-23		Data read.
Second Parameter Address	24-27		Copied from command. Value of 0xFF..FF means only the first parameter is read.
Second Parameter Data	28-31		Data read.

Read Alternate - Indirect

The Read Alternate - Indirect command reads a block of parameters to the alternate structure. The command defines the number of Dwords to be read and the starting address inside the alternate structure.

Name	Bytes.Bits	Value	Remarks
Flags	0-1		See Section 38.27.5.2.1 for details.
Opcode	2-3	0x0903	Command opcode.
Datalen	4-5		Size of buffer accompanying the command (in bytes).
Return Value/ VFID	6-7	0x00	N/A
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Alternate Structure Address	16-19		Lowest address to be read from the alternate structure. Accessing the alternate structure is done by addressing 32-bit entries.
Alternate Structure Length	20-23		Number of Dwords to be read from the alternate structure.
Data Address High	24-27	Buff Addr	High bits of buffer address.
Data Address Low	28-31	Buff Addr	Low bits of buffer address.

The following completion is sent for the Read Alternate - Indirect command:

Name	Bytes.Bits	Value	Remarks
Flags	0-1		See Section 38.27.5.2.2 for details.
Opcode	2-3	0x0903	Command opcode.
Datalen	4-5	0x00	Actual length of data returned by the command.
Return Value/ VFID	6-7		Some comments on specific errors: 0x0 = no error. ENOMEM - Out of memory (access outside the alternate structure). EACCES - Permission denied (access to another PF's area).
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Alternate Structure Address	16-19		Lowest address read from the alternate structure.
Alternate Structure Length	20-23		Number of Dwords read from the alternate structure.
Data Address High	24-27	Buff Addr	High bits of buffer address.

Name	Bytes.Bits	Value	Remarks
Data Address Low	28-31	Buff Addr	Low bits of buffer address.

Done Alternate Write

The Done Alternate Write command indicates to the 10 GbE controller that the CLP strings have been sent to it:

- Legacy BIOS mode — Sent once per device after all CLP strings have been sent to the 10 GbE controller (for all LAN ports). Following the command, firmware loads the contents of the alternate structure into the device functional units.
- UEFI mode — Sent once per each enabled LAN port. Once the command is received from all enabled ports, firmware loads the contents of the alternate structure into the 10 GbE controller's functional units.

Name	Bytes.Bits	Value	Remarks
Flags	0-1		See Section 38.27.5.2.1 for details.
Opcode	2-3	0x0904	Command opcode.
Datalen	4-5	0x00	N/A
Return Value/ VFID	6-7	0x00	N/A
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
BIOS Mode	16.0	See remarks	0b = Legacy BIOS. 1b = UEFI.
Reserved	16.7 - 16.1	0x00	Reserved.
Reserved	17-31	0x00	Reserved.

The following completion is sent for the Done Alternate Write command:

Name	Bytes.Bits	Value	Remarks
Flags	0-1		See Section 38.27.5.2.2 for details.
Opcode	2-3	0x0904	Command opcode.
Datalen	4-5	0x00	N/A
Return Value/ VFID	6-7		ENOSPC - More than 128 VFs are requested.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Reserved	16.0	0b	Reserved.
Return Flags	16.1		Reset Needed. When set, indicates that software should do a global reset for the alternate RAM content to take effect.
Reserved	16.2-31	0x00	Reserved.

Clear Port Alternate Write

The Clear Port Alternate command indicates to the 10 GbE controller that the alternate sections of all PF tied to the port. The port is inferred from the PF that sent the command.

Name	Bytes.Bits	Value	Remarks
Flags	0-1		See Section 38.27.5.2.1 for details.
Opcode	2-3	0x0906	Command opcode.
Datalen	4-5	0x00	N/A
Return Value/ VFID	6-7	0x00	N/A
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Reserved	16-31	0x00	Reserved.

The following completion is sent for the Clear Port Alternate command:

Name	Bytes.Bits	Value	Remarks
Flags	0-1		See Section 38.27.5.2.2 for details.
Opcode	2-3	0x0905	Command opcode.
Datalen	4-5	0x00	N/A
Return Value/ VFID	6-7		
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Reserved	16-31	0x00	Reserved.

38.13.2.2.3 Example of a SMASH/CLP flow - Legacy BIOS

The following pseudo code provides an example of how such a flow might be performed by legacy BIOS using SMASH/CLP commands. The following guidelines should be kept:

- If a certain PCI function is disabled via this mechanism, pre-boot software does not access any resource of that function (such as any CSR) once it sends the Done Alternate Write AQ command. The 10 GbE controller confirms the command, resets, and disables the function.
- By the time the SMASH/CLP commands are executed and a function is disabled, there is no Tx/Rx activity in the 10 GbE controller (since no queues have been initialized).
- All ports enabled in the NVM have the option ROM enabled (such as the lowest PCI function per port has an expansion ROM BAR).
- BIOS calls all CLP entry points for all functions before getting into the initialization phase.
- It is not guaranteed that a system reset is issued immediately following this sequence. For example, the configuration settings must take place even if such a reset is not issued.

- During the initialization phase, pre-boot software has to issue a global reset followed by a Start LLDP Agent AQ command to start the LLDP agent in firmware.

```
// Initial state:

//  The device is configured from NVM

// The following flow runs per each device

BIOS does PCI enumeration and discovers functions with Option ROM enabled

For each LAN Port with Option ROM enabled // sent to the lowest number PCI function
on the port

    BIOS loads the Option ROM into system RAM

    Set First-Init to true // When Init is later called, it's the 1st call to Init

    For each CLP string received by BIOS for a function within the port

        If the Option ROM for the port supports SMCLP entry point

            BIOS calls SMCLP entry point for port with CLP string

            SMCLP section in Option ROM processes the CLP string

            Option ROM keeps track of SMCLP status for the port

        End-If

    End-For

End-For

For each LAN port with Option ROM enabled

    BIOS calls INIT entry point for the port

End-For

SMCLP Entry Point

If CLP type is SET and action is “Return to default” // clear any pre-existing CLP
configuration

        Send the "Clear Port Alternate" admin command to invalidate all Alternate
Memory parameters for the port and its PFs

End-If

If CLP type is SET

    Generate “Write Alternate - Direct” admin command(s) to the device

    End-If

If CLP type is EXIT // Apply any configuration changes from previous commands that
have not

                        been applied

    Send appropriate admin commands with default values for the PF parameters not
specified by the CLP string that have a default behavior in this mode, different
than the hardware default.

End-If

End SMPCLP Entry Point
```

```
SMCLP INIT Entry point

If First-Init is true // the flow below should only be executed on the 1st call to
Init

    If SMCLP status is false // means no CLP strings have been received for any
functions for the device

        // Option ROM INIT configures the device in a standard operating mode (i.e.
not an OEM specific mode)

    End-if

    If SMCLP status is true // means at least one function had CLP strings

        // This is the time to load the CLP parameters from the Alternate Structure
into the device.

        Send "Done Alternate Write" command indicating end of CLP strings for the
device

    End-if

    // Global Reset should be done only once per device

    If Firmware required a reset in the "Done Alternate Write" response, Issue
Global Reset of the device

    Set First-Init to false

End-if

Continue with INIT code...

End SMCLP INIT Entry Point
```

38.13.2.2.4 Example of a SMASH/CLP Flow - UEFI

The following pseudo code provides an example of how such a flow might be performed by the UEFI drivers using SMASH/CLP commands. The following guidelines should be kept:

- If a certain PCI function is disabled via this mechanism, pre-boot software does not access any resource of that function (such as any CSR) once it sends the Done Alternate Write AQ command for that function. The 10 GbE controller confirm the command, resets, and disables the function.
- By the time the SMASH/CLP commands are executed and a function is disabled, there is no Tx/Rx activity in the 10 GbE controller (since no queues have been initialized).
- A UEFI driver is executed for each enumerated LAN port. For example, any port enabled in the NVM and not disabled by strapping.
- It is not guaranteed that a system reset is issued immediately following this sequence. For example, the configuration settings must take place even if such a reset is not issued.

```
// Initial state:

// The device is configured from NVM

// The following flow runs per each device

BIOS does PCI enumeration and discovers PCI functions

For each enabled LAN Port
```

```
   BIOS calls driver START entry point

   For each CLP string received by BIOS for a function within the port

       SMCLP section processes the CLP string

   End-For

   Driver sends a "Done Alternate Write" command indicating end of CLP strings for
the port

    If Firmware required a reset in the "Done Alternate Write" response, Issue
Global Reset of the device

End-For

SMCLP section in START

If CLP type is SET and action is "Return to default" // clear any pre-existing CLP
configuration

    Send the "Clear Port Alternate" admin command to invalidate all Alternate
Memory parameters for the port and its PFs

End-If

If CLP type is SET

   Generate "Write Alternate - Direct" admin command(s) to the device

   End-If

If CLP type is EXIT // Apply any configuration changes from previous commands

                    // that have not been applied

   Send appropriate admin commands with default values for the PF parameters not
specified by the CLP strings that have a default behavior in this mode, different
than the hardware default.

End-If

End SMPCLP section
```

38.13.2.2.5 Processing The Alternate Structure

If a Done Alternate Write AQ command is received, the *GLNVM_GENS.ALT_PRST* bit is set by the 10 GbE controller. This bit indicates that an alternate structure exists in the 10 GbE controller. As a result of each Done Alternate Write AQ command, the EMP loads the relevant alternate structure parameters into the 10 GbE controller according to the following sequence:

- EMP loads any required parameters from the alternate structure into the hardware.
- EMP ACKs the Done Alternate Write command.
- Global reset is performed:
  - — Initiated by software.
  - — Reset is done only once and not per each instance of the Done Alternate Write AQ command.
  - — Hardware is initialized, including loading of the relevant sections of the alternate structure into hardware.

38.13.2.3 Network Boot

Each PF can be a boot function and can independently support PXE, iSCSI boot:

- All functions are PXE boot (such as up to 16 such functions).
- Up to four functions might be iSCSI boot functions.

38.13.2.4 Device State

This section is limited to changes in the 10 GbE controller's state made in the BIOS initialization stage.

38.13.2.4.1 Switch / Tx Scheduler

Some parameters of the switch and Tx scheduler configuration might change by the contents of the alternate structure. Such changes take effect when the alternate structure is enabled.

38.13.2.4.2 LAN

Expansion ROM code might set LAN QPs for network boot. The sequence of setting a LAN QP is described in Section 38.38, "Programming Interface".

38.13.2.4.3 Interrupts

Interrupts need to be set if used for AQ or LAN queues operation. See Section 38.22.1.1 for the exact flow.

38.13.3 Driver Load

38.13.3.1 Introduction

38.13.3.1.1 Driver Load (Non-Virtualized)

As described earlier in this section, by the time the software device driver loads, the NVM configuration is already complete and PCIe configuration has taken place. During a software device driver load, the following sequence of commands is typically issued to the 10 GbE controller to initialize it for normal operation. The major initialization steps are:

1. Device driver probes the 10 GbE controller for resource allocations.
2. Disable interrupts.
3. Initialize the admin queue - see Section 38.13.
4. Initialize HMC - see Section 38.26.
5. Initialize MAC/PHY - see Section 38.11.1.
6. Initialize power management --Section 38.17.4 (Wake Up).
7. Initialize DCB (including Rx-PB) - see Section 38.24.4.
8. Initialize the switch and Tx scheduler - see Section 38.21.7.17 and Section 38.25.4.19.
9. Initialize statistics by reading the counter's initial values to serve as a baseline.
10. Init Protocol Engine - see Section 38.36.3.1.
11. Initialize 1588.
12. Enable interrupts.

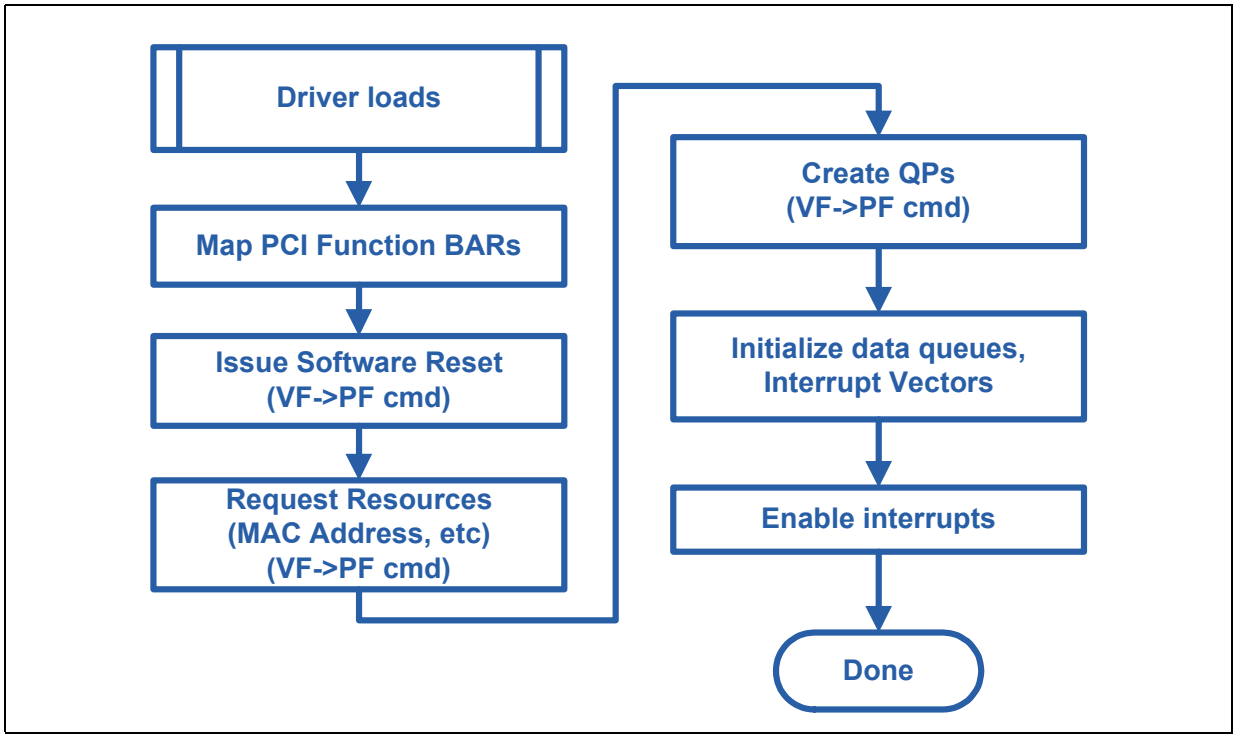
At this point, the 10 GbE controller is ready to initialize VSIs and LAN PE flows. These are done dynamically during operation:

- Initialize VSI.
- Initialize LAN QP; including its interrupts (see Section 38.30.2.2) or Init PE QP (see Section 38.14).
- Configure filters.

38.13.3.1.2 Driver Load (SR-IOV)

The 10 GbE controller approach for virtualized device drivers is to depend heavily on the PF device driver for management of chip resources. The device driver models for newer virtualized operating systems are moving in a direction that require such functionality. The following figure shows the high-level initialization flow for virtualized device drivers that use a VF in a guest operating system.

Figure 38-29.Virtualized Device Driver Initialization Flow



PF Initialization Details

In addition to the regular device driver initialization flow, the PF device driver should apply the following steps to enable support for VMs.

1. After the operating system enables virtual bridging, the PF device driver should create a VEB or a VEPA switching element using the following flow:
   a. Query the switch structure using the Get Switch Configuration command in order to get the SEID number to which this PF is connected. This SEID might be the port or an S-channel.
   b. Create a VEB/VEPA using the Add VEB command (Add VEB). The control port can be either the original VSI of the PF or a dedicated VSI.
   c. Connect the switch in place of the current VSI using the Insert Element command (Add VEB).

d. Connect default and mirror ports as needed using the Add VSI and Connect Elements commands. See Add VSI (0x0210) and Get VSI Parameters (0x0212).

e. Define mirroring rules using the Mirroring Rules commands (Mirroring).

f. Activate malicious driver protection through the GL_MDCK_RX, GL_MDCK_TDAT and GL_MDCK_TCMD registers.

2. When the VMM requests that a virtual port be created, the PF driver should:

a. Create a set of VSIs according to the flow described in the paragraphs that follow. The exact flow depends on the VMM-to-PF API.

b. Define the bandwidth allocated to the VSI and each of its TCs using the Scheduler Configuration commands (Section 38.25.4).

c. If the virtual port is a VF, allow VF access to the network by clearing the associated bit in *GL_VIRT_VFLRE* register.

VF Initialization Details

This section describes the flow used to initialize a VF. Only stages involving the hardware are detailed.

Software Reset

A VF software reset can be asserted only by the PF using the *VPGEN_VFRTRIG.VFSWR* field. Following a VF software reset, the VF should request re-initialization of the queues from the PF.

Following the reset, the PF should preform the clean-up steps described in Section 38.13.1.2.5.

Request Resources and Create Initialize Data Queues

On top of the resources statically allocated to a VF (interrupts, RSS table, etc.), a VF might have a set of VSIs. Each VSI contains objects such as:

- User priorities (transmit queue groups).
- Queue pairs
- Queuing filters

VSIs can be requested either for LAN or iWARP.

The resources should be requested in the following order (operations with the same stage number can be done in any order):

Step	Resource Requested	PF Action	Notes
1	VSI and UPs	Allocate VSIs according to allocation policies using the Add VSI AQ command. See Add VSI (0x0210).	The PF should activate the security features in the VSI (anti spoof, port based VLAN) according to the VF settings.
2	Forwarding table entries (MAC and VLANs)	Add the requested MAC and VLAN and MAC, VLAN pairs using the forwarding table configuration commands (Section 38.21.7.18.8).	
3	Data queue pairs	Create the requested queue contexts as described in Section 38.31.3.1 and Section 38.30.3.3.1.	
	PE resources	The PF flow is described in Section 38.36.3.1.	Before using any PE related resources, the HMC must be properly programmed. See Section 38.26 for more details on the HMC initialization. Virtual function HMC initialization must be performed by the PF driver on behalf of the VF driver

38.14 Device/Port/Function Configuration

38.14.1 General

The 10 GbE controller provides several mechanisms to configure which PCI functions and Ethernet ports are exposed:

- Through NVM configuration
- Bases on power management policy
- Through strapping pins

38.14.1.1 Port-to-function Mapping

The PFGEN_PORTNUM.PORT_NUM per-PF register field (loaded from the NVM) associates each PF with a port.

38.14.2 Disable Through Strapping Pins

For a LAN on Motherboard (LOM) design, it might be desirable for the system to provide BIOS setup capability for selectively enabling or disabling the 10 GbE controller PCI devices, PCI functions, or external Ethernet ports and the associated PCI function. It enables end users more control over system resource management and avoids conflicts with add-in NIC solutions. The 10 GbE controller provides support for selectively enabling or disabling one or more LAN PCI device(s) in the system.

The 10 GbE controller provides strapping pins to disable its external Ethernet ports and/or PCI functions:

- One pin (LAN_DIS_N) is sampled on RSMRST# and PCIe resets to disable Ethernet ports. The specific port(s) to be disabled is determined from NVM. Additional NVM configuration is provided to determine which PCI functions are disabled at the same time. The expected usage is to disable the PCI functions associated with the disabled ports.

- One pin (PCI_DIS_N) is sampled on RSMRST# and PCIe resets to disable PCI functions. The specific functions that are disabled are determined from the NVM.

Some guidelines on usage of LAN_DIS_N and PCI_DIS_N:

- During power up, the PCI_DIS_N and LAN_DIS_N pins are ignored until the NVM is read. From that point, the 10 GbE controller might disable all PCI functions if either PCI_DIS_N or LAN_DIS_N is asserted.
- De-assertion of the PCI_DIS_N or LAN_DIS_N pins requires the system to issue a power on reset/RSMRST#/PE_RST_N/in-band reset to the 10 GbE controller in order to re-enable any disabled PCI functions or external Ethernet ports.
- The PCI_DIS_N and LAN_DIS_N pins should maintain their state during system reset and system sleep states. It should also insure the proper default value on system power up. For example, one could use a GPIO pin that defaults to 1b (enable) and is on system suspend power (such that it maintains its state in S0-S5 ACPI states).

38.14.3 Port And Device Disable

The following mechanisms are provided to enable and disable Ethernet ports:

- NVM configuration
  - Ports are enabled or disabled from the NVM. The PRTGEN_CNF.PORT_DIS per-port register bit (loaded from the NVM) disables external ports other than port 0 (PORT_DIS for port 0 is hard wired to always enabled).
  - When cleared, the respective port is enabled
  - The hardware default value is for port 0 to be enabled and other ports to be disabled
- Soft SKU commands
  - Soft SKU commands might enable or disable LAN ports. The Soft SKU commands are executed either before or after PERST# was de-asserted and in any case, before the first access by BIOS to the 10 GbE controller (like before the first PCIe configuration cycle to the 10 GbE controller).
  - EMP updates the configuration of the ports enabled or disabled through the Soft SKU command by writing to the PRTGEN_CNF.EMP_PORT_DIS register bits.
- Power management
  - If the Ethernet ports are not required in Dr state (such as if manageability is disabled and APME WoL is disabled as well), then all ports are disabled during Dr state.
- Strapping (LAN_DIS_N)
  - When the LAN_DIS_N pin is asserted low, the PRTGEN_CNF.ALLOW_PORT_DIS per-port bits (loaded from the NVM) determine if the port is disabled.
  - The hardware default value for these bits is 0x0 (do not disable).
  - If a bit is set to 0b, LAN_DIS_N has no effect on the port.

Note: If a port is required for manageability purposes, it should not be disabled by the mechanisms previously described.

The result of the previous mechanisms is reflected in the PRTGEN_STATUS.PORT_VALID bit (per port), which denotes if the port is enabled. A bit is cleared (port disabled) if at least one of the previous mechanisms disables the port. The port is then powered down (including MAC, PCS, PHY).

If all ports are disabled, the 10 GbE controller is put in a power-down, reset state. Specifically, the 10 GbE controller does not respond to PCI configuration cycles, the PCIe link is in L3 state, and Ethernet ports are in power down. All PCI functions must be disabled as well via the mechanisms previously described (such as the proper NVM configuration must be set to disable for PFs).

38.14.3.1 Dynamic Port Shutdown

Another related, per-port status bit is the PRTGEN_STATUS.PORT_ACTIVE indication. This is a dynamic state indicating that the port is temporarily inactive and is powered down. When the port is inactive, re-activating it, does not require any reset and the related PCIe function is still active.

A port is active (PRTGEN_STATUS.PORT_ACTIVE = 1b) based on the following rules:

1. PRTGEN_STATUS.PORT_VALID = 1b, as previously described above AND the 10 GbE controller is in D0 state (PMCSR.PowerState = D0).
   - If PRTPM_GC.EMP_LINK_ON is set to 1b (such as the interface is being used for manageability) then the link is kept on. Specifically, hardware ignores disabling the link using the PRTGEN_CNF.ACTIVATE_PORT_LINK.
   - Else,
     - In systems where the application needs to prevent any traffic on link before the device driver is loaded, the PRTGEN_CNF2. ACTIVATE_PORT_LINK is loaded from the NVM with the value of 0b (disable).
     - Once the device driver loads, it uses the Set Link and Restart AN with the command bit 2 set (Enable Link). Following this, the EMP enables the link by writing the value of 1b to the PRTGEN_CNF2. ACTIVATE_PORT_LINK and starts the link bring-up sequence.
     - If the device driver is about to be removed or disabled, then before going down the device driver might disable the link by the Set link and Restart AN with the command bit 2 cleared (Disable Link). Following this, firmware disables the link by writing the value of 0b to the PRTGEN_CNF2. ACTIVATE_PORT_LINK.
2. PRTGEN_STATUS.PORT_VALID = 1b, as previously described, AND the 10 GbE controller is in Dr state (PMCSR.PowerState = D3).
   - If port is enabled for wake-up, follow the description in the wake-up section.
   - Else, behavior is defined by the EMP_LINK_ON bit:
     - The hardware default value for PRTPM_GC.EMP_LINK_ON is 0b (link should be down as it is not required for EMP functionality).
     - BMC might enable manageability by sending the appropriate command. As a result, EMP transitions to pass-through manageability-on state, enables the respective port (if disabled) by setting the PRTPM_GC.EMP_LINK_ON bit for the port, and starts the link bring-up sequence.
     - If the channel is disabled by the BMC (by sending the appropriate command) and EMP has no other needs for the LAN port in Dr state (like proxy), then the EMP reverts the PRTPM_GC.EMP_LINK_ON bit for the port to the value of EMP_LINK_ON bit in the NVM manageability module or is set to zero (in case of the Disable Channel command with the ALD bit set).

38.14.4 Function Disable

The following mechanisms are provided to enable and disable PCI functions.

- NVM configuration
  - PCI functions are enabled or disabled from the NVM. The PFPCI_FUNC.FUNC_DIS per-PF bit (loaded from the NVM) disables PCI functions (other than function 0. FUNC_DIS for function 0 is hardwired to enabled).
  - When cleared, the respective function is enabled.
  - The hardware default value is for function 0 to be enabled and other functions to be disabled.
- Strapping through PCI_DIS_N
  - When the PCI_DIS_N pin is asserted low, the PFPCI_FUNC.ALLOW_FUNC_DIS per-PF register bit (loaded from the NVM) determines if the function is disabled.
  - If this bit is set to 0b, PCI_DIS_N has no effect on the PCI function.
  - The hardware default value is 0b (keep enabled).
- Strapping through LAN_DIS_N
  - When the LAN_DIS_N pin is asserted low, the PFPCI_FUNC.DIS_FUNC_ON_PORT_DIS per-PF bit (loaded from the NVM) determines if the function is disabled.
  - If this bit is set to 0b, LAN_DIS_N has no effect on the PCI function.
  - The hardware default value is 0 (do not disable).
- Soft SKU commands
  - The soft SKU commands are executed either before or after PERST# was de-asserted and in any case, before the first access by BIOS to the 10 GbE controller (such as before the first PCIe configuration cycle to the 10 GbE controller).
  - When a LAN port is enabled or disabled via a Soft SKU command, the EMP identifies (through the PFGEN_PORTNUM registers) the PCI functions associated with the port.
  - EMP then updates the configuration of the functions enabled or disabled by writing to the PFPCI_FUNC.FUNC_DIS register bits.
  - PFPCI_FUNC.FUNC_DIS for function 0 is hardwired to enabled, so is not affected by this mechanism.
  - The GLGEN_PCIFCNCNT register reflects the number of enabled PFs as loaded from the NVM (determined by PFPCI_FUNC.FUNC_DIS only). The value is loaded from the NVM.

The result of the previous mechanisms is reflected in the PFPCI_STATUS1.FUNC_VALID bit (per PF), which denotes if the function is enabled. A bit is cleared (function disabled) if at least one of the previous mechanisms disables the function.

When a function is disabled:

- It does not respond to PCI configuration cycles (unless specified otherwise). Effectively, the function becomes invisible to the system.
- The *PME_En* bit is cleared to avoid issuing PME.

The Ethernet ports associated with disabled PCI functions are still available for manageability purposes.

38.14.4.1 Dummy Function

When PCI function 0 is disabled, it does not disappear from the PCIe configuration space. Rather, the function presents itself as a dummy function. The device ID and class code of this function changes to other values (dummy function device ID 0x10A6, class code 0xFF0000, with an option to load from the NVM) that claims 4 KB of memory. In addition, the function does not require any I/O space and does not require an interrupt line. All other PCI functions keep their respective locations.

38.14.5 Event Flow for Enable/Disable Ports and PCI Functions

This section describes the expected flow to disable or enable PCI functions or Ethernet ports. Following a power-on reset/RSMRST#/PE_RST_N/in-band reset, the LAN_DIS_N and PCI_DIS_N signals should be driven high (or left unconnected) for normal operation.

The following example assumes that PCI functions and/or Ethernet ports are being disabled during the BIOS initialization phase:

1. Following a power-up sequence, the LAN_DIS_N and PCI_DIS_N signals are driven high.
2. The PCIe link is established following PE_RST_N.
3. BIOS goes through PCI bus enumeration.
4. BIOS recognizes that the PCI functions in the 10 GbE controller should be disabled.
5. The BIOS drives the LAN_DIS_N or PCI_DIS_N signal to a low level.
6. The BIOS asserts PCIe reset, either in-band or via PE_RST_N.
7. As a result, the 10 GbE controller samples the LAN_DIS_N and PCI_DIS_N signals and disables the PCI functions and/or external Ethernet ports.
8. BIOS might do device enumeration a second time (the disabled PCI functions are invisible or changed to dummy function).
9. Proceed with normal operation.
10. Re-enable could be done by driving the LAN_DIS_N and PCI_DIS_N signals high and re-issuing a power-on reset/RSMRST#/PE_RST_N/in-band reset.

38.14.5.1 Multi-Function Advertisement

If all but one of the PCI functions are disabled, the 10 GbE controller is no longer a multi-function device. The 10 GbE controller normally reports a 0x80 in the PCI configuration header field header type, indicating multi-function capability. However, if only a single LAN is enabled, the 10 GbE controller reports a 0x0 in this field to signify single-function capability.

38.14.5.2 Legacy Interrupts Use

Each NVM PF configuration module specifies the interrupt line used for each PCI function. When more than one PCI function is enabled, the 10 GbE controller uses the INTA# to INTD# interrupts for interrupt reporting. The specific interrupt pin used is reported in the *PCI Configuration Header Interrupt Pin* field associated with each PCI function.

However, if only one PCI function is enabled, then the INTA# must be used for this PCI function, regardless of the NVM configuration. Under these circumstances, the *Interrupt Pin* field of the PCI header always reports a value of 0x1, indicating INTA# usage.

38.14.5.3 Power Reporting

When more than one PCI function is enabled, the PCI power management register block has the capability of reporting a common power value. The common power value is reflected in the *Data* field of the PCI Power Management registers. The value reported as common power is specified via the *LAN Power Consumption* NVM word (word 0x22), and is reflected in the *Data* field each time the *Data_Select* field has a value of 0x8 (0x8 = common power value select).

When only one PCI function is enabled, the 10 GbE controller appears as a single-function device, the common power value, if selected, reports 0x0 (undefined value), as common power is undefined for a single-function device.

38.15 Shared Resource Management

The 10 GbE controller is a device with multiple external Ethernet ports and that supports multiple PCI functions. Several on-chip resources are shared between device drivers that load the PCI functions exposed by the 10 GbE controller. Additionally, some resources for SR-IOV VFs are managed by the associated PFs. In general, the 10 GbE controller minimizes the requirement for coordination between device drivers running on different PCI PFs through a combination of resource partitioning and programming interface assistance for remaining resources that are shared.

The 10 GbE controller handles shared resources using a number of allocation mechanisms and/or access control mechanisms. Table 38-100 lists the supported mechanisms.

Table 38-100.Supported Mechanism (Sheet 1 of 2)

Class	Description
Dedicated	This resource is associated with a particular 10 GbE controller element and is always available without respect to any other configuration or setting. Examples of elements in this context could be a PCI function or external Ethernet port.
Administered	Administered resources can be re-assigned based on BMC, BIOS settings, or other OEM specific mechanisms. These resources can be changed with a PCI reset but do not change dynamically after the PCI reset.
Profiles	Resources that are allocated via profiles are typically divided up among different entities based on some combination of other input. A profile might dictate that a resource is divided among the active PCI functions in a particular manner. Another possible usage of a profile is to divide resources based on both the number of PCI functions and the number of external Ethernet ports. The division of resources that are allocated by profiles happens between the time that a PCI reset occurs and might be impacted by an initial device driver load, but resource allocation based on profiles require a PCI reset to change after the initial device driver load.
Pool	Resources that are allocated from pools are typically allocated in a fashion that guarantees no resource starvation to an individual consumer of a resource but enables flexible allocation of remaining resources to any consumer that requires additional resources beyond the minimum. Pools are used for resources that do not need to have a maximum number of resources allocated to all consumers at the same time and that the consumers are reasonably able to fail an allocation.
Service	Resources that are possibly allocated on a per-device or per-port basis but are accessed by a software service that has visibility to multiple device driver interfaces that do not need the 10 GbE controller to provide an arbitration mechanism. In these cases each PCI function associated with a given resource provides equal access to the resource as other PCI functions associated with the resource. This enables the software service to have the flexibility to access such resources from any device driver instance that it finds available and to switch between device driver instances as the device drivers are stopped and started. Resources that need to accesses directly by a device driver without support from an overlaying software service cannot be handled in this fashion.

Table 38-100.Supported Mechanism (Sheet 2 of 2)

Class	Description
Arbitrated	Resources that are allocated on a per-device or per-port basis need an access control mechanism that enables multiple consumers of the resource to operate in an independent manner. The 10 GbE controller does this by either providing a firmware interface to access the resource where firmware handles the requests one at a time or by other hardware based arbitration scheme.

Table 38-101.Used Mechanism (Sheet 1 of 2)

Resource	Accessible From PF or VF	Allocation Or Access Type	Description
External Ethernet Ports	PF	Administered and Arbitrated	External Ethernet ports are assigned to the 10 GbE controller PCI physical PCI functions based on NVM settings. PF drivers configure their respective ports via AQ commands.
NVM	PF	Arbitrated	NVM access requires acquiring ownership via an AQ command.
PCI VFs	N/A	Administered	The number of VFs that are active and the assignment of VFs to PFs is handled as part the PCI PF configuration listed previously in this table.
PCI and PCIe	PF and VF	Various	PCIe resources are dedicated or shared per the PCIe specification. Mostly loaded from the NVM and write-disabled to host software.
Admin Queues	PF and VF	Dedicated	Each function has a dedicated AQ command queue.
MSI-X Vectors	PF and VF	Profiles	The 10 GbE controller supports a fixed number of MSI-X vectors that are distributed evenly among PFs and VFs based on the number of active PFs and VFs.
PCIe Performance Counters	PF	Service	Performance counters are a shared resource among PFs (service mode.
HMC	PF and VF	Profiles	Each PF (and VF with PE resources) is allocated a separate PCI function FPM. Each function is allocated some number of SDs (configured via the NVM with an option to override via firmware). See Section 38.26 for more detail.
LAN Queues	PF and VF	Profiles	Queue pairs (Tx/Rx) are allocated to PFs based on NVM configuration. Allocation of QPs to VFs is dynamic and managed by each PF (see Section 38.39).
Hash Instances (RSS, etc.)	PF VF	Dedicated	The 10 GbE controller supports independent hash filters for each VSI.
Quad-hash Filters	PF and VF	Dedicated	Each PF (or VF) is allocated filter entries in FPM and are cached on-die. Caching is FCFS.
TimeSync	PF	Service	The 10 GbE controller supports TimeSync for all PFs per external Ethernet port. The 10 GbE controller provides an indication of which PF owns TimeSync per each LAN port. However, it is system software's responsibility to avoid contention between PFs. See Section 38.39.2.20 for more detail.
Internal Switching Elements	PF	Pool	The internal switch is configured through a combination of control mechanisms (such as NVM, SMASH/CLP) and PF drivers. The following resources are managed: VEBs, VSIs, filter entries, VSI lists, mirroring resources. The general scheme is a basic quota per PF (loaded from the NVM) and FCFS for all the remaining. PF configuration is done via AQ commands.
VSIs	PF	Service	See Section 38.25.5.3 for more detail on the resource allocation mechanism for this resource.
PE Resources	PF and VF	Profiles	QPs, CQs and CEQs are distributed between PFs and VFs based on resource profiles. A single AEQ is allocated per PCI function FPM is used per PF for memory-based resources.
Accelerated Port Bit Vector Table	PF and VF	Arbitrated	The 10 GbE controller provides a single Accelerated Port Bit Vector Table (APBVT) per external Ethernet port. The 10 GbE controller provides arbitration for APBVT manipulation between software device drivers running on separate PCI functions that are associated with the same APBVT.

Table 38-101.Used Mechanism (Sheet 2 of 2)

Resource	Accessible From PF or VF	Allocation Or Access Type	Description
PE Push Pages	PF and VF	Pool	The 10 GbE controller supports a pool of push pages tracking structures that are accessed through the push page array found in BAR0. Each PCI function that is enabled for PE push mode operations uses CQP to request a push page tracking structure using the CQP operation. Push mode pages are allocated by software per process and any process that cannot allocate a push page tracking structure reverts to using in-line operations instead of push operations.
Flow Director	PF Only	Profiles or Service	The 10 GbE controller supports a set of flow director filters, allocated between PFs. Each PF is allocated some guaranteed space. All remaining are allocated as FCFS counters and are allocated evenly among PFs.
Tx Scheduler	PF	Various	Queue sets are allocated with a basic VSI quota per PF (loaded from the NVM) and FCFS for all the remaining. The shared part of the scheduler (such as ETS) is controlled by the PF driver. The per-PF part of the scheduler (like VEB) is controlled by PF via the firmware AQ commands.
Receive Packet Buffer and DCB	PF	Profiles and Service	A receive packet buffer is logically partitioned across the enabled ports. The memory allocated to a port is configured across the TCs according to DCBX protocol resolution. It is done by the PF.
SDPs and LEDs	PF	Arbitrated	SDPs are associated with an Ethernet port or with the device. Association is provided from the NVM. Ownership of a shared SDP is via an AQ command. See Section 38.7 for more details regarding SDP usage with the 10 GbE controller.

38.15.1 Resource Profiles

The concept of resource profiles is used in order to make the 10 GbE controller's resource allocation mechanisms easier to understand. Resource profiles dictate how resources listed in Table 38-100 as profiles are distributed among PCI functions so that software drivers do not need to coordinate allocation of these resources. A resource profile is made up of the set of equations that take input parameters from the system configuration and distributes each resource. Each resource has a different equation for distribution (see the description column in Table 38-100). Also, the following tables list examples of the result calculations for a full set of resources. Since some of the resource distribution equations simply take the number of PCI functions as input while others might need additional input (such as the number of external Ethernet ports or the software usage model that a device driver needs to implement).

Resource	# External Ethernet Ports	# PCI Physical Functions	PCI Function to External Ethernet Port Assignment	# PCI Virtual Functions	Driver Usage Model
LAN Queues	No	Yes	No	Yes	No
MSI-X Vectors	No	Yes	No	Yes	No
Multicast MAC Address Filters	No	Yes	No	No	No
Internal Switching Elements[1]	Yes	Yes	Yes	No	No
Protocol Engine Resources	No	Yes	No	Yes	Yes

Notes:

1. Resource profiles only impact the default settings for internal switching elements. Run-time switch management software has the ability to override the default number of internal switching components and the internal switch topology.

The following table lists a few sample sets of typical input parameters that are used with the 10 GbE controller. The initial values of these resources are found in the NVM but might be overridden by OEM specific configuration mechanisms followed by a PCI reset. Note that the device driver usage model can be changed and locked by the first device driver to load following a PCI reset. All other inputs are set at PCI reset.

Scenario Name	# External Ethernet Ports	# PCI Physical Functions	PCI Function to External Ethernet Port Assignment	# PCI Virtual Functions	Driver Usage Model
Single Port Default	1	1	PF0 -> Port 0	128	Default
Dual Port Default	2	2	PF0 -> Port 0 PF1 -> Port 1	128	Default
Quad Port Default	4	4	PF0 -> Port 0 PF1 -> Port 1 PF2 -> Port 2 PF3 -> Port 3	128	Default
Octal Port Default	8	8	PF0 -> Port 0 PF1 -> Port 1 PF2 -> Port 2 PF3 -> Port 3 PF4 -> Port 4 PF5 -> Port 5 PF6 -> Port 6 PF7 -> Port 7	128	Default
Single Port SR-IOV VF Primary	1	1	PF0 -> Port 0	128	SR-IOV PF Primary
Single Port SR-IOV Even Distribution	1	1	PF0 -> Port 0	128	SR-IOV Even Distribution

The table that follows lists the resulting resource distribution for each of the input scenarios listed in the previous table. With the exception of the switch topology, none of the resource distribution can be changed after the initial device driver load without a PCI reset occurring. If the resource distribution was to change while an active device driver was running, unpredictable results can occur. For example, re-partitioning the host memory cache segment descriptor table while a device driver was active causes the 10 GbE controller to improperly interpret the host memory used for caching resource objects. The 10 GbE controller provides various mechanisms (such as the HMC resource profile locking mechanism) to prevent software from unknowingly causing these types of reconfigurations.

Resource	# LAN QPs Per PF	#LAN QPs Per VF	#MSI-X Vectors Per PF	#MSI-X Vectors Per VF	Default Internal Switch Topology	HMC and PE Resources Per PF	HMC and PE Resources Per VF
Single Port Default	512	8	129	5	PF0 -> Port 0	4096 SDs 256K PE QPs 128K PE CQs 128 PE CEQs	0 SDs 0 PE QPs 0 PE CQs 0 PE CEQs
Dual Port Default	256	8	129	5	PF0 -> Port 0 PF1 -> Port 1	2048 SDs 128K PE QPs 64K PE CQs 128 PE CEQs	0 SDs 0 PE QPs 0 PE CQs 0 PE CEQs
Quad Port Default	128	8	129	5	PF0 -> Port 0 PF1 -> Port 1 PF2 -> Port 2 PF3 -> Port 3	1024 SDs 64K PE QPs 32K PE CQs 128 PE CEQs	0 SDs 0 PE QPs 0 PE CQs 0 PE CEQs

Resource	# LAN QPs Per PF	#LAN QPs Per VF	#MSI-X Vectors Per PF	#MSI-X Vectors Per VF	Default Internal Switch Topology	HMC and PE Resources Per PF	HMC and PE Resources Per VF
Single Port SR-IOV Primary Distribution	128	8	129	5	PF0 -> Port 0	139 SDs 1K PE QPs 512 PE CQs 8 PE CEQs	123 SDs 8128 PE QPs 4064 PE CQs 4 PE CEQs
Single Port SR-IOV Even Distribution	128	8	129	5	PF0 -> Port 0	349 SDs 14560 PE QPs 7168 PE CQs 14 PE CEQs	220 SDs 14560 PE QPs 7168 PE CQs 14 PE CEQs[1]

Notes:

1. The SR-IOV Even Distribution is shown with 16 VFs enabled for Protocol Engine operation.

38.16 Power Management

This section defines how Power Management is implemented in the 10 GbE controller.

38.16.1 PCIe Power Management

38.16.2 Auxiliary Power Usage

Auxiliary power might be used for powering the 10 GbE controller while the system is in low power state ($D3_{COLD}$). The 10 GbE controller uses the AUX_PWR pin as an indication that auxiliary power is available to it. The 10 GbE controller uses this indication to advertise $D3_{COLD}$ wake up support in the PMC.PME_Support field and to set the Aux_Power_Detected bit in the PCIe capability structure Device_Status_Register. The AUX_PWR pin is strapped during Power On Reset (POR).

The amount of power required for the function, which includes the entire Network Interface Card (NIC) is advertised in the Power Management Data register, which is loaded from the NVM.

If AUX power usage during $D3_{COLD}$ is supported (as indicated below), all The 10 GbE controller's sticky bits will preserve their values and will only get reset by the power up reset (detection of power rising).

When AUX power is applied to The 10 GbE controller, the actual usage of AUX power during $D3_{COLD}$ is controlled by the following factors:

1. NVM loaded bits:

- EMP_LINK_ON — Bit per port loaded from NVM to PRTPM_GC register which indicates if the relevant port link should be established for EMP functionality. The 10 GbE controller will provide the proper clocking to establish the required port link/s at this state.

2. APME — Bit per function (loaded to PFPM_APM[PF]) indicates if the relevant PF associated to a port link should be established to enable APM WoL. The 10 GbE controller will provide the proper clocking to establish the required port link/s at this state.

PCIe configuration bits:

PME_En - The PME_En bit of the PMCSR PCI config space is used to determine if the 10 GbE controller is allowed to consume AUX power for WoL.

The following pseudo code defines AUX Power usage where APME and PME_En bits refer to a logical OR over all the PFs attached to the port:

```
If (AUX_PWR = 1)

    if ((APME or EMP_LINK_ON or PME_En) = 1)

        AUX power is used to preserve link functionality

    Else

        Link functionality is not preserved during AUX power supply
```

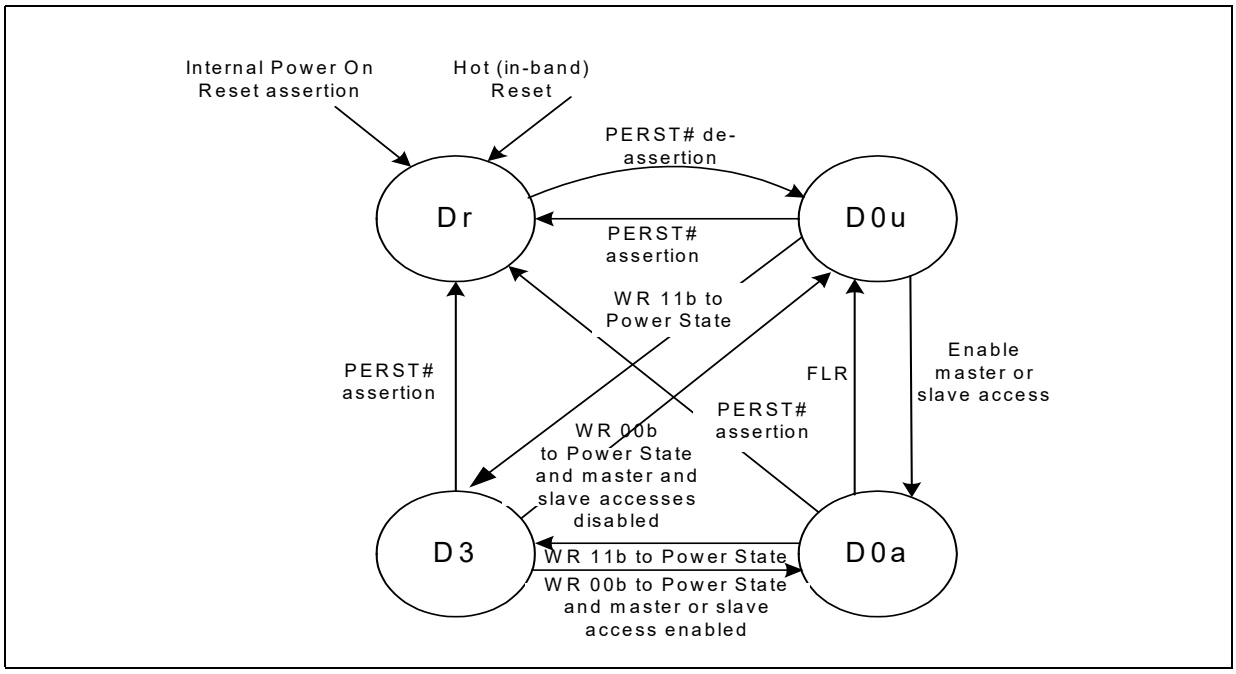
38.16.3 Power States

The 10 GbE controller supports the D0 and D3 architectural power states as described earlier. Internally, the 10 GbE controller supports the following power states:

- D0u (D0 un-initialized) - an architectural sub-state of D0.
- D0a (D0 active) - an architectural sub-state of D0.
- D3 - architecture state $D3_{HOT}$.
- Dr - internal state that contains the architecture $D3_{COLD}$ state. Dr state is entered when PE_RST_N is asserted, a PCIe in-band reset is received, or if the function is disabled.

Figure 38-30 shows the power states and transitions between them.

Figure 38-30.Power Management State Diagram



38.16.3.1 $D0_{uninitialized}$ State

The D0u state is an architectural low-power state. In this state, the device is out of reset (conventional and FLR) but did not complete the enumeration and configuration stage.

38.16.3.1.1 Entry to a D0u State

D0u is reached from either the Dr state (on de-assertion of internal PE_RST_N and auto load of hardware configuration done) or the $D3_{HOT}$ state (by configuration software writing a value of 00b to the *Power State* field of the PCI PM registers while master and slave accesses are disabled). De-assertion of internal *PE_RST_N* causes the entire state of the 10 GbE controller to be cleared except for bits defined as sticky in configuration space. PCIe configuration is loaded from the NVM, followed by establishment of the PCIe link and enumeration. Once this is done, configuration software can access the 10 GbE controller.

38.16.3.2 D0active State

A Product Name Function enters the $D0_{active}$ state each time any single or combination of the Function's Memory Space Enable, I/O Space Enable, or Bus Master Enable bits have been enabled by system software in the PCI Command configuration register.

In this state it can transmit and receive packets if properly configured by the software device driver. The PHY is enabled or re-enabled by the software device driver to operate/auto-negotiate to full line speed/power if not already operating at full capability.

Notes:

1. Wake behavior:
   a. An APM wake event (PM_PME message) due to reception of a Magic packet is not generated when the function is in D0 active state.
   b. Any APM wake up previously active remains active when moving from D3 to D0.
   c. WAKE# pin is never toggled for an APM wake event when a function is in D0.
2. If APM wake is required in D3, software driver should not disable APM wake-up via clearing the *PFPM_APM.APME* bit on entry into D0. Otherwise, APM wake following a system crash and entry into S3, S4 or S5 system power management state is not enabled. Following entry into D0, the software device driver can activate other wake-up filters by writing to the Wake Up Filter Control (*PFPM_WUFC*) register.

38.16.3.2.1 Entry to D0a State

D0a is entered from either the D0u state (by writing a 1b to the *Memory Access Enable* or the *I/O Access Enable* bit or the BME bit of the PCI *Command* configuration register) or from the $D3_{HOT}$ state (by configuration software writing a value of 00b to the *Power State* field of the PCI PM configuration registers while master or slave accesses is enabled). The receive and transmit flows of the appropriate LAN function are enabled.

38.16.3.3 D3 State (PCI-PM $D3_{HOT}$)

The 10 GbE controller functions can transition to D3 when the system writes a 11b to the Power State field of the *Power Management Control/Status Register (PMCSR)*. Any wake-up filter settings that were enabled before entering this state are maintained. The

10 GbE controller does not clear any bit in the PCIe configuration space of a function during the function's transition to D3 state. While in D3, the appropriate function of the 10 GbE controller does not generate master cycles.

Configuration and message requests are the only TLPs accepted by a function in the $D3_{HOT}$ state. All other received requests must be handled as unsupported requests, and all received completions are handled as unexpected completions. If an error caused by a received TLP (such as an unsupported request) is detected while in $D3_{HOT}$, and reporting is enabled, the link must be returned to L0 if it is not already in L0 and an error message must be sent. See section 5.3.1.4.1 in the *PCIe Base Specification*.

38.16.3.3.1 Entry to D3 State

Transition to D3 state is through a configuration write to the *Power State* field of the *PMCSR* PCIe configuration register.

Prior to transition from D0 to the D3 state, the device driver disables scheduling of further tasks to the 10 GbE controller; it masks all interrupts, it does not write to the Transmit Descriptor Tail register or to the Receive Descriptor Tail register.

If wake up capability is needed, system software should enable wake capability by setting to 1b the *PME_En* bit in the *PMCSR* PCIe configuration register of the PF. After Wake capability has been enabled Software device driver should set up the required wake up functionality using the flow described in Section 38.17.4 prior to the D3 transition.

If protocol offload (proxying) capability is required, the software device driver should enable proxying operation.

Notes:

1. The *PMCSR.PME_En* bit setting can be overridden via the *PFPM_APM.APME* bit.
2. If operation during $D3_{COLD}$ is required, even when Wake capability is not required (such as for manageability operation), system should also set the *Auxiliary (AUX) Power PM Enable* bit in the PCIe *Device Control register*.

The 10 GbE controller waits for the completion of all previous TLPs it has sent. Any receive packets that have not been transferred into system memory are kept in the device (and discarded later on D3 exit). Any transmit packets that have not be sent can still be transmitted (assuming the Ethernet link is up).

In preparation for a possible transition to $D3_{COLD}$ state, the device driver might disable up to three LAN ports out of four (LAN disable) and/or transition the remaining link(s) to GbE speed (if supported by the network interface).

38.16.3.3.2 Exit from D3 State

A D3 state is followed by either a D0u state (in preparation for a D0a state) or by a transition to Dr state (PCI-PM $D3_{COLD}$ state). To transition back to D0u, the system writes a 00b to the *PMCSR*. Transition to Dr state is through *PE_RST_N* assertion.

The *No_Soft_Reset* bit in the *PMCSR* in the 10 GbE controller is always set to 1b, to indicate that The 10 GbE controller does not perform an internal reset on transition from $D3_{HOT}$ to D0 so that transition will not disrupt the proper operation of other active Functions. Software is not required to re-initialize the function's configuration space after a transition from $D3_{HOT}$ to D0 (the function is in the D0 state).

The function is reset if the link state had transition to the L2/L3 ready state, on transition from $D3_{COLD}$ to D0, if FLR is asserted or if transition $D3_{HOT}$ to D0 is caused by assertion of PCIe reset (PE_RST pin).

38.16.3.4 Dr State ($D3_{COLD}$)

Transition to Dr state is initiated on several occasions:

- On system power up — Dr state begins with the assertion of the internal power detection circuit (RSMRST#) and ends after the completion of the NVM auto load.
- When the IP is in D3 or D0a state the transition to Dr state begins with a dedicated message. Upon the reception of this message the 10 GbE controller performs the following operations:
  - — Suspends the primary scheduling of new TLPs and waits for the completion of all previous TLPs it has sent.
  - — Responds to all pending non-posted requests.
  - — Completes the scheduling of any of the following pending events:
    - PCI error
    - Assert/de-assert INTx
- Indicate whether the IP needs to stay powered-on during Sx states (such as for supporting WoL or MNG) to the SoC. Note that the 10 GbE controller might indicate that it needs to stay powered-on during Sx if a long NVM transaction (like erase) is in progress.
- After a response is sent, the 10 GbE controller is ready to transition to Dr state and therefore takes additional steps to limit its operation.
  - — PCI error and assert/de-assert INTx messages are not scheduled for any further transmission.
  - — At this point the IP is ready to have its primary reset asserted.

Any wake-up or proxying filter settings that were enabled before entering this reset state are maintained.

The system can maintain PE_RST_N asserted for an arbitrary time. The de-assertion (rising edge) of PE_RST_N causes a transition to D0u state.

While in Dr state, the 10 GbE controller can maintain functionality (for WoL or manageability) or can enter a Dr Disable state (if no WoL and no manageability) for minimal device power. The Dr disable mode is described in the following section.

38.16.3.4.1 Dr Disable Mode

The 10 GbE controller enters a Dr disable mode on transition to $D3_{COLD}$ state when it does not need to maintain any functionality. The conditions to enter either state are:

- The device (all PCI functions) is in Dr state.
- APM WOL is inactive for all PCI functions.
- Proxying is not required for any PCI functions (PFPM_PROXYFC.PPROXYE is cleared to 0b).
- Pass-through manageability is disabled.

Entry into Dr disable is usually done on assertion of PCIe PE_RST_N. It can also be possible to enter Dr disable mode by reading the NVM while already in Dr state. The usage model for this later case is on system power up, assuming that manageability and wake up or proxying are not required. Once the device enters Dr state on power

up, the NVM is read. If the NVM contents determine that the conditions to enter Dr disable are met, the device then enters this mode (assuming that PCIe PE_RST_N is still asserted).

Exit from Dr disable is through de-assertion of PCIe PE_RST_N.

If Dr disable mode is entered from D3 state, the platform can remove the 10 GbE controller power. If the platform removes the 10 GbE controller power, it must remove all power rails from the device if it needs to use this capability. Exit from this state is through power-up cycle to the 10 GbE controller.

Note: The state of the TX_DIS (optical module Tx disable connected to SDP pins) or any of the other SDP pins is undefined once power is removed from the device.

38.16.3.4.2 Entry to Dr State

Dr-entry on platform power up is as follows:

- Assertion of the internal power detection circuit (RSMRST#). Device power is kept to a minimum by keeping the Network interfaces in low power.
- The NVM is then read and determines device configuration.
- If the *APM Enable* bit in the NVM is set, then APM wake up is enabled (for each port independently).
- If the *MNG Enable* bit in the NVM word is set, pass-through manageability is enabled.
- Each of the LAN ports can be enabled if required for WoL or manageability. See Section 38.17 for exact condition to enable a port.

Entry to Dr state from D0a state is through assertion of the PE_RST_N signal. An ACPI transition to the G2/S5 state is reflected in a device transition from D0a to Dr state. The transition can be orderly (such as a user selected a shut down operating system option), in which case the software device driver can intervene. Or, it can be an emergency transition (like power button override), in which case, the device driver is not notified.

Transition from D3 state to Dr state is done by assertion of PE_RST_N signal.

38.16.3.5 Power Gating

When the 10 GbE controller enters Dr disable mode, there is an opportunity to improve the power consumption of the device through power gating.

In order to do this the 10 GbE controller needs to communicate with the power management controller (PMC/RU) that it is ready to can be power gated.

At initialization time (before global Reset ACK is asserted), the 10 GbE controller's firmware determines whether to enable/disable power gating based on NVM loaded configurations. Since the indication is reflected to the power management controller when entering a Dr state, at initial boot, the 10 GbE controller is always powered on and then enters Dr state to indicate the required power state to the power management controller. If the 10 GbE controller needs to be powered-on during Sx states, it stays powered-on; otherwise, it is power gated. Refer to the Global Reset Flow section for more detail.

Note: The gbe_pmc_S5_enable signal is ignored by the 10 GbE controller when entering DeepSx state. In this case, the 10 GbE controller is always power gated.

The logic used by firmware to decide if power gating is allowed in S5 is:

FW_allows_PG_in_S5 =

(Manageability Mode = None) OR

(OEM Capabilities.CEM Capable) OR

(Pass through Mode = NONE and Control Mode = PCIe)

Once the 10 GbE controller is power gated, transition back to Dr state is managed as a standard power up.

The following steps describes the power gating flow:

1. The 10 GbE controller enters the Dr disable state indicating it can be power gated (gbe_pmc_S5_enable = 0). Firmware sets this bit via the FWPWRGATE.PWRGATE_EN aux register.
2. PMC recognizes that the IP should be power gated and sends a power gate I/O controller message to the IP.
3. The 10 GbE controller receives the force power gate message.
4. The Power Gating Control Block (PGCB) power gating sequence begins.

Note: When the IP is power gated, any SDPs and other miscellaneous I/Os being driven by the IP is invalid and should be assigned a default value by the SoC.

38.17 Network Interfaces Power Management

The 10 GbE controller disables a network interface and transitions the interface into low-power state in the following cases:

- All LAN ports are in LAN disable mode as a result of the LAN_DIS_N pin.
- Low power state functionality as described in Section 38.16.2.

When the 10 GbE controller is in low-power state, the 10 GbE controller asserts the respective TX_DIS pin (connected to an SDP pin) to enable powering down an external PHY or optical module as well.

38.17.1 BASE-T LPLU

When connected to an external BASE-T PHY The 10 GbE controller's firmware engine performs the LCD auto-negotiation operation by controlling the advertised speeds of the BASE-T PHY and essentially limiting the speeds being advertised in order to force both link partners to auto-negotiate to the lowest possible speed.

LCD auto-negotiation is attempted in the following order:

- 1 GbE
- 10 GbE

38.17.2 Wake Up Filters

The 10 GbE controller supports issuing a wake-up indication to the host after receiving error free packets when the device is in D3 low power state or in Dr with WoL enabled, using Magic packet filters set from the NVM. Note that support of wake up in Dr is only if the AUX_PWR bit is asserted.

38.17.2.1 Wake Up Filter Types

The following packet types are detected per PF by the 10 GbE controller as a possible wake up.

38.17.2.1.1 Magic Packet

Magic packets are defined in:

http://support.amd.com/TechDocs/20213.pdf as:

"Once the LAN controller has been put into the Magic Packet mode, it scans all incoming frames addressed to the node for a specific data sequence. This sequence indicates to the controller that this is a Magic Packet frame. A Magic Packet frame must also meet the basic requirements for the LAN technology chosen, such as Source Address, Destination Address (which may be the receiving station's IEEE address or a Multicast address which includes the Broadcast address), and CRC. A magic packet must also be a valid non error L2 packet. The specific data sequence consists of 16 repetitions of the IEEE address of this node, with no breaks or interruptions. This sequence can be located anywhere within the packet, but must be preceded by a synchronization stream. The synchronization stream allows the scanning state machine to be much simpler. The synchronization stream is defined as 6 bytes of 0xFF. The device will also accept a Broadcast frame, as long as the 16 repetitions of the IEEE address match the address of the machine to be awakened."

The 10 GbE controller expects the destination address to either:

- Be the broadcast address (FF.FF.FF.FF.FF.FF).
- Match the value of the port L2 Ethernet destination MAC address (DA) set by FW to the relevant filters PRTPM_SAL/H.

On power up firmware loads the PRTPM_SAL/H addresses from the NVM. Each address entry contains:

- MAC address for Magic packet filtering
- Address Valid (AV) indication
- Promiscuous Multicast magic filtering enable (MC) indicating the L2 destination address can be any multicast address
- The PF number (PF_NUM) to be reported if a Magic packet matches this filter

Firmware uses the PFGEN_PORTNUM registers to map the per PF MAC address to the relevant ports.

The 10 GbE controller searches for the contents of the port's Ethernet destination MAC address receive address as the embedded IEEE address. It considers any non-0xFF byte after a series of at least 6 0xFFs to be the start of the IEEE address for comparison purposes. For example, it catches the case of 7 0xFFs followed by the IEEE address). As soon as one of the first 96 bytes after a string of 0xFFs don't match, it continues to search for another set of at least 6 0xFFs followed by the 16 copies of the IEEE address later in the packet. Note that this definition precludes the first byte of the destination address from being FF.

A Magic packet's destination address must match the address filtering enabled in the configuration registers with the exception that broadcast packets are considered to match even if broadcast packet reception is not enabled. If APM wake up (wake up by a

Magic packet) is enabled in the NVM, the 10 GbE controller starts up with the Ethernet MAC destination address loaded from the NVM. This enables the 10 GbE controller to accept packets with the matching IEEE address before the software device driver loads.

Table 38-102.Magic Packet Structure

Offset	# Of Bytes	Field	Value	Action	Comment
0	6	Destination Address		Compare	MAC header – processed by main address filter.
6	6	Source Address		Skip	
12	T=(0/4/8)	Possible S-TAG		Skip	
12 + T	S=(0/4)	Possible VLAN Tag		Skip	
12 + T + S	D=(0/8)	Possible Length + LLC/SNAP Header		Skip	
12 + T+ S + D	2	Type		Skip	
Any	6	Synchronizing Stream	FF*6+	Compare	
any+6	96	16 Copies of Node Address	A*16	Compare	Compared to relevant Station Addresses (PRTPM_SAH, PRTPM_SAL) registers.

38.17.3 Wake Up and Virtualization

When operating in a virtualized environment, all wake-up capabilities are managed by a single entity (such as the VMM or an IOVM). In an IOV architecture, the physical (PF) driver controls wake up and none of the Virtual Machines (VMs) has direct access to the wake-up registers. The wake-up registers are not replicated per VF.

38.17.4 Wake Up Flows

38.17.4.1 Wake Up Enable Flow

A WoL register set exists in each PF. Each set consists of the following wake-up filters and registers:

1. *PFPM_WUC* - Wake Up Control Register.
2. *PFPM_WUS* - Wake Up Status Register.

Before entering D3, the software device driver should set the needed filters to send to the WoL VSI (retrieved from the Discover Function Capabilities command) in the switch using existing switch AQ commands and set the requested WoL filters using the Set WoL Filter AQ command. If needed, the software device driver should set the WoL VSI to promiscuous mode to enable firmware to receive all traffic. The switch AQ commands should enable software to set this VSI as filters destination.

Software can add up to four filter rules per port for this VSI. Subsequent filters are rejected by firmware.

If filters need to be deleted, the Clear all WoL Switch filters command should be used and not the regular delete filter commands.

Note: ACPI filters might be defined earlier in the software device driver life, but adding forwarding rules to the ACPI VSI should be done at the last moment to prevent overloading firmware with host traffic.

The ToQueue option is ignored for filters sent to the ACPI VSI. Only regular VLAN can be set for this VSI.

The filters set to ACPI VSI are cleared at PF reset or a PCI reset event.

Wake up is activated only when the device enters Dr.

At power up, values loaded from the NVM into the following per PF registers define enablement of APM wake-up functionality:

1. The NVM *APM Enable* bit is loaded to the *PFPM_APM.APME* register bit to enable:
   a. Detection of a Magic packet destined to the PF.

Note: The software device driver can query firmware via the Get Wake Reason AQ command to read the filter that woke up the system. If the wake up was due to a magic packet, then there is no reporting of the reason (PFPM_WUS register is not updated).

38.17.4.2 Wake Up Disable Flow

Once the 10 GbE controller wakes the system following transition to D0 the software device driver might disable WoL directly by:

1. Clearing all the pending wake-up status bits in the *Wake Up Status* (*PFPM_WUS*) register.
2. Clearing the relevant bits in the *PFPM_WUC register.*
3. Clear all WoL filters using the Set WoL Filter AQ command.

Note: This flow is not used by regular Intel drivers.

38.17.4.3 ACPI Wake-Up Flow

Once wake up is enabled, the 10 GbE controller monitors incoming packets, if a packet passes at least one of the enabled wake-up filters, the 10 GbE controller:

- Sets the *PME_Status* bit in the PFPM_WUS and PCI *PMCSR* register.
- If the *PME_En* bit in the *PMCSR* configuration register is set, asserts PE_WAKE_N and/or sends a PM_PME PCIe message.
- Sets one or more of the bits in the *PFPM_WUS* register. The 10 GbE controller sets more than one bit if a packet matches more than one filter.

Note: In addition, a link state change wake-up also causes similar results. Sets the *PFPM_WUS.PME_Status* and the *PMCSR.PME_Status* bit, asserts the PE_WAKE_N signal, and/or sends a PM_PME PCIe message and sets the relevant bit in the *PFPM_WUS* register.

The PE_WAKE_N signal remains asserted, PM_PME messages are periodically sent to the host, and the 10 GbE controller ignores any subsequent ACPI wake-up packets and link change events for that function until the operating system either writes a 1b to the *PME_Status* bit of the *PMCSR* register or writes a 0b to the *PME_En* bit or until de-assertion of PERST.

38.18 Protocol Offload (Proxying)

The 10 GbE controller provides a proxy capability to receive, respond or generate packets instead of the host, this is manly targeted (but not limited) to avoid spurious wake up events and reduce system power consumption when the device is in D3 low power state and system is in S3 or S4 low power states.

The 10 GbE controller supports protocol offload (proxying) of:

1. A single IPv4 ARP request per function.
   — Responds to IPv4 address resolution request with a host MAC (L2) address (as defined in RFC 826).
2. Two IPv6 Neighbor Solicitation (NS) requests per function, where each NS protocol offload request includes 2 IPv6 addresses, for a total of 4 possible IPv6 addresses per function.
   — Responds to IPv6 NS requests with a host MAC (L2) address (as defined in RFC 4861).
3. When NS protocol offload is enabled, the 10 GbE controller also supports protocol offload of up to two IPv6 Multicast-Address-Specific MLD (Multicast Listener Discovery) Queries (either MLDv1 or MLDv2) per function. In addition, the 10 GbE controller also responds to General MLD Queries, used to learn which IPv6 multicast addresses have listeners on an attached link.
   — MLD protocol offload is supported when NS protocol offload is enabled so that IPv6 routers discovers the presence of multicast listeners (that is, nodes wishing to receive multicast packets), for packets with the IPv6 NS solicited-node multicast address and continue forwarding these NS requests on the link.
   — MLD protocol offload is supported for either MLD Multicast Listener Query packets or MLD Multicast Address and Source Specific Query packets that check for IPv6 Multicast Listeners with the solicited-node multicast address placed in the IPv6 destination address field of the IPv6 NS packets that are off-loaded by the 10 GbE controller.
   — Responds to the IPv6 MLD Queries, with the solicited-node multicast address placed in the IPv6 destination address field of the IPv6 NS packets that are off-loaded by the 10 GbE controller (as defined in RFC 2710 and RFC 3810).

Note: When protocol offload is supported by the 10 GbE controller the Features Enable NVM word should be set such as the appropriate *Proxy Enable for Port X* bits are set. EMP assigns a VSI for each port that supports protocol offload.

When proxying is enabled over a port, the PRTPM_GC.EMP_LINK_ON bit is set to 1b by EMP for enabling the link establishment.

38.18.1 Proxying Filters

The 10 GbE controller supports issuing protocol offload (proxying) of packets using pre-defined filters per PF.

Proxy filters are enabled if the corresponding bit in the per PF *Proxying Filter Control* (*PFPM_PROXYFC*) register is set to 1b.

38.18.1.1 Proxy Pre-defined Filters

The following packet types are detected per PF by the 10 GbE controller's pre-defined filters:

- General and directed ARP/IPv4 request packet
- General and directed NS IPv6 packet
- MLD IPv6 packet

Each of these packet type filters is enabled if the corresponding bit in the per PF *Proxying Filter Control* (*PFPM_PROXYFC*) register is set to 1b.

38.18.1.1.1 ARP/IPv4 Request Packet

The 10 GbE controller supports receiving ARP request packets for proxying if the *Directed ARP* bit or the *ARP* bit is set in the Enabled Features in the Set Proxying Configuration AQC *Proxying Filter Control* (*PFPM_PROXYFC*) register.

- If the *Directed ARP* bit is set, a successfully matched packet is a packet that passes the packet structure comparisons as defined in Section 38.21.6.2.6, has a broadcast MAC address or passed the proxy MAC address filtering and the target IP address matches the IPv4 address programmed in the *IPv4 internal Address Table* programmed through Set Proxying Configuration command (see Section 38.18.5.1).
- If the *ARP* bit is set, a successfully matched packet is a packet that passes the packet structure comparisons as defined in Section 38.21.6.2.6, has a broadcast MAC address or passed the proxy MAC address filtering.

The 10 GbE controller handles ARP request packets that have VLAN tagging.

ARP packet structure is described in Section 38.21.6.2.6.

38.18.1.1.2 NS IPv6 Packet

The 10 GbE controller supports receiving IPv6 NS packets sent by a node to determine the link-layer address of a neighbor, or to verify that a neighbor is still reachable for wake up or proxying.

If the *NS* or *NS_DIRECTED* bits are set in the Enabled Features in the Set Proxying Configuration AQC *Proxying Filter Control* (*PFPM_PROXYFC*) register packet is sent to firmware for protocol offload.

- If the *NS_DIRECTED* bit is set a successfully matched packet that is directed to the EMP is a packet that passes the packet structure comparisons as defined in Section 38.21.6.2.5, matches one of the pre configured MAC address (unicast or multicast), a ICMPv6 type of 0x87 (NS), correct ICMPv6 checksum, and the IPv6 destination address field matches one of the possible four programmed IPv6 solicited-node multicast addresses programmed through Set NS Proxy Table Entry command (see Section 38.18.5.3). EMP is responsible to further filter the incoming packets on the ICMPv6 target IPv6 address.
- If the *NS* bit is set a successfully matched packet is a packet that passes the packet structure comparisons as defined in Section 38.21.6.2.5, matches one of the pre configured MAC address (unicast or multicast), a ICMPv6 type of 0x87 (NS) and a correct ICMPv6 checksum. EMP is responsible for the rest of the filtering as defined in the NS_DIRECTED section.

The 10 GbE controller handles NS IPv6 packets that have VLAN tagging.

NS packet structure is described in Section 38.21.6.2.5.

38.18.1.1.3 MLD IPv6 Packet

The 10 GbE controller supports receiving IPv6 IPV6 MLD packets sent by an IPv6 router to discover the presence of multicast listeners (that is, nodes wishing to receive multicast packets) on its directly attached links, and to discover specifically which multicast addresses are of interest to those neighboring nodes.

After receiving the following MLD packets (either MLDv1 as defined in RFC 2710 or MLDv2 as defined in RFC 3810):

1. Multicast Listener Query (ICMPv6 Type = decimal 130). Defined in MLDv1 and MLDv2.
2. Multicast Listener Report (ICMPv6 Type = decimal 131). Defined in MLDv1 and MLDv2.
3. Version 2 Multicast Listener Report Message (ICMPv6 Type = decimal 143). Defined in MLDv2 only.

If the *MLD* bit is set in the *Proxying Filter Control* (*PFPM_PROXYFC*) register packet is sent to firmware for protocol offload.

- If the *MLD* bit is set a successfully matched packet is a packet that passes the packet structure comparisons as defined in Section 38.21.6.2.5, matches one of the pre-configured MAC address (unicast or multicast), a ICMPv6 type of 0x82, 0x83 or 0x8F (MLD), correct ICMPv6 checksum.

The 10 GbE controller handles MLD IPv6 packets that have VLAN tagging.

MLD packet structure is described in Section 38.21.6.2.5.

Note: Currently, the Microsoft operating system API to driver for MLDv2 does not support specifying a source address list for filter mode. It means that no source address fields are inserted by the EMP in the MLDv2 reports returned. Such support has to be added by the EMP in the future if it is required for getting the Microsoft* logo.

38.18.2 Proxying and Virtualization

When operating in a virtualized environment, all proxying capabilities are managed by a single entity (such as the VMM or an IOVM). In an IOV architecture, the physical (PF) driver controls proxying and none of the VMs has direct access to the proxying registers. The proxying registers are not replicated per VF.

38.18.3 Protocol Offload (Proxying) Flow

A proxying register set exists per PF. Each set consists of the following proxying registers:

4. *PFPM_PROXYFC* - Proxying Filter Control register.
5. *PFPM_PROXYS* - Proxying Status register.

Proxying filters (not directly accessed as device registers):

6. MAC — L2 address table that holds four MAC addresses.
7. *IP4* — IPv4 Address table that holds one IPv4 addresses.
8. *IP6* — IPv6 Address table that holds four IPv6 addresses.

The PF driver can enable proxying by sending commands to firmware via the Admin command queue and receiving firmware responses via the Admin Event queue.

Note: Proxy filters values are preserved across resets. The filters can be added/removed using Admin commands. The actual filtering can be enabled/disabled using the PFPM_PROXYFC register.

38.18.3.1 Protocol Offload Activation

To enable protocol offload, the PF software device driver should implement the following flow:

9. Software device driver should make sure that all packets that are needed for proxy offload are forwarded to the WoL and proxy VSI as exposed in Discover Function Capability AQC response before enabling proxy offload.
10. Software device driver programs the relevant proxying filtering tables and registers by sending *Admin Queue commands to add IPv4/IPv6 table entries and set the PFPM_PROXYFC control register. The s*oftware device driver sends to firmware the information required to execute a specific protocol offload via the *Set (protocol offload type) Proxy Table Entry command* via the Admin queue. The supported Proxy Table Entry commands are:
    — Set Proxying Configuration command (see Section 38.18.5.1).
    — Set NS Proxy Table Entry command (see Section 38.18.5.3).
    a. Firmware loads the table entries into internal memory: It loads the solicited-node multicast addresses into the device internal IPv6 proxy tables. A multicast address is derived from a unicast IPv6 address posted by software by prefixing as follows:

       FF02::1:FFXX:XXXX, where XXXXXX are the 24 LS bits of the unicast IPv6 address.
    b. Firmware clears all pending proxying status bits in the *Proxying Status* (*PFPM_PROXYS*) register.
    c. Firmware programs the *Proxying Filter Control* (*PFPM_PROXYFC*) register to define packets that should be forwarded to the EMP for protocol offload.
    d. Firmware responds with a *Set Proxy Table Entry/SetNS Proxy Table Entry Responses* (see Section 38.18.5.2 and Section 38.18.5.4) via the Admin queue to acknowledge completion of the command and indicate if command was completed successfully.
11. Following the successful completion of the previous flow, hardware directs the configured packet types to the EMP and firmware creates the proxy responses according to the configuration.

When the 10 GbE controller is in D0 power state, the software device driver might read the PFPM_PROXYS register directly. When the 10 GbE controller is in D0 power state the software device driver might send a *Set Proxying Configuration Command* (see Section 38.18.5.1) with all or some of the proxying conditions cleared to disable some or all proxying functionality.

38.18.3.2 Proxying Packet Flow

Once proxying is enabled by setting the *PFPM_PROXYFC.PPROXYE* bit to 1b, the 10 GbE controller monitors incoming packets by filtering them with all of the proxying filters enabled in the *PFPM_PROXYFC* register. If a packet passes one of the enabled proxying filters and does not pass any of the enabled wake-up filters, the 10 GbE controller:

1. Executes the relevant protocol offload for the packet and not forward the packet to the host. For NS/MLD proxying, firmware must complete the filtering by looking for a match between the target IPV6 address carried inside the ICMPv6 payload and the addresses loaded by software via the Set NS Proxy Table Entry command.

2. Sets one or more bits in the *Proxying Status (PFPM_PROXYS)* register according to the proxying filters matched.
   — Note that the 10 GbE controller sets more than one bit in the *PFPM_PROXYS* register if a packet matches more than one filter.
3. Wakes the system and forwards a packet that matches the proxying filters but cannot be supported to the host for further processing if configured to do so by the software device driver via the *Set Proxying Configuration command*.

Notes:

1. When the device is in D3, a packet that matches one of the enabled proxying filters as defined in the *PFPM_PROXYFC* register and one of the enabled wake-up filters as defined in the *PFPM_WUFC* register wakes up the system and protocol offload (proxying) occurs.
2. Protocol offload is not executed for illegal packets with CRC errors or checksum errors and the packets are silently discarded.
3. Once a packet that meets the criteria for proxying is received, the 10 GbE controller should respond to the request after less than 60 seconds.

38.18.3.3 Disabling Protocol Offload

To disable protocol offload the software device driver should implement the following steps:

1. If IPv6 addresses were configured to this PF — Send a *Set NS Proxy Table Entry command* (see Section 38.18.5.3) with the relevant IPv6 table entries to be removed.
   a. Firmware clears the table entries from FLU and internal memory so that the proxying resources are available for other PFs.
   b. Firmware responds with a *Set NS Proxy Table Entry Response* (see Section 38.18.5.4) via the Admin queue to acknowledge completion of the command and indicate if the command was completed successfully.
2. If IPv4 addresses were configured to this PF — Send a *Set Proxying Configuration command* (see Section 38.18.5.1) with the relevant IPv4 table entry to be removed.
   a. Firmware clears the table entry from FLU and internal memory so that the proxying resources are available for other PFs.
   b. Firmware responds with a *Set Proxying Configuration Response* (see Section 38.18.5.2) via the Admin queue to acknowledge completion of the command and indicate if the command was completed successfully.
3. Send a *Set Proxying Configuration Command* (see Section 38.18.5.1) with all of the proxying conditions cleared to disable all proxying functionality.
   a. Firmware clears all proxying status bits in the *Proxying Status* (*PFPM_PROXYS*) register.
   b. Firmware clears any required bits in the *PFPM_PROXYFC* register.
   c. Firmware responds with a *Set Proxying Configuration Response* (see Section 38.18.5.2) via the Admin queue to acknowledge completion of the command and indicate if command was completed successfully.

Note: Steps #2, #3 can be combined to a single Admin queue message.

38.18.4 WoL Related Commands

38.18.4.1 Set WoL Filter AQ

This command sets a filter for the port attached to the PF that sent the command.

Table 38-103.Set WoL Filter Command (Opcode: 0x0120)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x120	Command Opcode.
Datalen	4-5	144/0	Length of buffer.
Return value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Filter Index	16-17		17.7: Filter type: flex/magic. 0b = set a flex filter. 1b = Set the magic packet filter. 17.6-16.3: Reserved. 16.2 – 16.0: The filter to set: 0-7. Relevant only if not magic packet.
Command Flags	18-19	0x0	Bit 15: Action: 0b = clear filter. 1b = set filter. Bit 14: No WoL in TCO traffic: 0b = Pass through packets might cause a wake up. 1b = Pass through packets are not candidate for wakeup. Bit 13: WoL filter preserve on PFR. Relevant for set filter action only. Set by software before D0->Dr/D3 state transition, cleared by software after returning to D0 state. 0b = Not preserve WoL filter on PFR (default upon POR/EMPR). 1b = Preserve WoL filter on PFR. Bit 12:0: Reserved.
Valid Flags	20-21	0x0	Bit 15 = Filter Action is valid. Bit 14 = No WoL in TCO traffic action is valid. Bit 13:0 = Reserved.
Reserved	22-23	0x0	Reserved.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

Table 38-104.Buffer Content (for ACPI Filters)

Name	Bytes.Bits	Remarks
Filter	0 - 127	128 bytes of the filter. Should be compared to the content of the first 128 bytes of the packet. Each byte is compared if its mask is set. Byte 0 corresponds to the first byte of the packet.
Mask	128 – 143	16 bytes of mask – 1 bit per byte of the filter. Bit = 0: Compare byte. Bit = 1: Skip byte.

Table 38-105.Set WoL Filter Response (Opcode: 0x0120)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0120	Command opcode
Datalen	4-5	0	Length of buffer
Return value/VFID	6-7		Return value. ERANGE: Filter Index is larger than 7.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-23		Reserved.
Data Address high	24-27		Address of buffer.
Data Address low	28-31		

38.18.4.2 Get Wake Reason AQ

This command can be used by the software device driver to receive information of the reason of a wake-up event.

Note: This command provides only reasons related to the firmware-based filtering. Hardware-based filters are exposed in the PFPM_WUS register.

Table 38-106.Get Wake Up Reason Command (Opcode: 0x0121)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x121	Command opcode.
Datalen	4-5	0	
Return value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-31	0x0	Reserved

Table 38-107.Get Wake Up Reason Response (Opcode: 0x0121) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0121	Command opcode
Datalen	4-5	0	
Return value/VFID	6-7		Return value. ENOENT: No WoL event recorded
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-17		
WoL Reason	18-19		18 = Index of matching filter (0xFF – no filter matched). 19 = Reserved.

Table 38-107.Get Wake Up Reason Response (Opcode: 0x0121) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Reserved	20-31		Reserved.

38.18.5 Protocol Offload Related Commands

The following section describes Admin commands related to protocol offload. The Protocol Offload commands can only be generated by PFs.

38.18.5.1 Set Proxying Configuration Command

Description — Software device driver requests to set the *PFPM_PROXYFC* register and supply the ARP offload parameters for the PF.

Command Type — Indirect Command.

Table 38-108.Set Proxying Configuration Command Table Entry

Name	Bytes.Bits	Value	Remarks
Command Flags	0-1.2	Reserved	Must be 0x0, value is ignored.
	1.3	Initialize IPv4 Table	Set to 1b to initialize/clear the IPv4 address table, this bit has the same effect as removing IPv4 address (bit 1.7 set and a valid Table ID).
	1.4	Unsupported Control	0b = Discard unsupported proxy packet. 1b = Wake the system on reception of unsupported proxy packet (packet is still discarded).
	1.5	Enable	Enable PFPM_PROXYFC.
	1.6	Add IPv4	Add IPv4 address entry.
	1.7	Remove IPv4	Remove IPv4 address entry.
Table ID	2-3	Table ID	16 bit Table index ID, valid only to remove IPv4 address entries (bit 1.7 is set).
Enabled Offloads	4-7	Enabled Offloads	bit 5 = ARP Directed offload is enabled. bit 11 = ARP offload is enabled. All other bits are reserved.
IPADDR	8-11	IP Address	Local IP address in Little Endian order.
MACADDR	12-17	MAC Address	MAC address in Big Endian order.

Table 38-109.Set Proxying Configuration Command Descriptor Fields (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0x0	
Opcode	2-3	Opcode	0x0104.
Datalen	4-5	Length of Buffer	Length of table entry.
Return value	6-7	Return Value	Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, ife copied by firmware into the completion of this command.
Reserved	16-19	Reserved	Must be 0x0, value is ignored.
Reserved	20-23	Reserved	Must be 0x0, value is ignored.
Data Address High	24-27	Data Address High	Data Address High. High 32 address bits (bits 63:32) of buffer address that holds the *Set Proxying Configuration command* structure.

Table 38-109.Set Proxying Configuration Command Descriptor Fields (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Data Address Low	28-31	Data Address Low	Data Address Low. Low 32 address bits (bits 31:0) of buffer address that holds the *Set Proxying Configuration command* structure.

38.18.5.2 Set Proxying Configuration Response

Description — Acknowledge to *Set Proxying Configuration command*.

Response Type — In direct response.

Table 38-110.Set Proxying Configuration Command Descriptor Fields

Name	Bytes.Bits	Value	Remarks
Reserved	0-1	Reserved	Value is ignored.
Table ID	2-3	Table ID	16 bit Table index ID provided by the EMP.
Reserved	4-17	Reserved	Value is ignored.

Table 38-111.Set Proxying Configuration Response Descriptor Fields

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0x0	
Opcode	2-3	Opcode	0x0104.
Reserved	4-5	0x0	Must be 0x0, value is ignored.
Return Value	6-7	Return Value	Return Value. Firmware supplies in the *Return value* field indication on the completion of the *Set Proxying Configuration command*. 0x0 = No error. Others = Error detected in the command.
Cookie High	8-11	Cookie	Opaque value, copied by firmware from the *Set Proxying Configuration command*.
Cookie Low	12-15	Cookie	Opaque value, copied by firmware from the *Set Proxying Configuration command*.
Reserved	16-31	Reserved	Reserved.

38.18.5.3 Set NS Proxy Table Entry Command

Description — PF driver supplies IPv6 NS protocol offload information via the *Set NS Proxy Table Entry command*.

Command Type — Indirect command, opcode and address of the IPv6 NS table entry command structure are placed in the descriptor, all other values are inserted in the structure.

Table 38-112.Set NS Proxy Table Entry Command Descriptor Fields (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0x0	
Opcode	2-3	Opcode	0x0105.
Datalen	4-5	Data Length	Length in bytes of buffer.
Return value	6-7	0x0	Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.

Table 38-112.Set NS Proxy Table Entry Command Descriptor Fields (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Cookie Low	12-15	Cookie	Opaque value, is be copied by firmware into the completion of this command.
Reserved	16-19	Reserved	Must be 0x0, value is ignored.
Reserved	20-23	Reserved	Must be 0x0, value is ignored.
Data Address High	24-27	Data Address High	Data Address High. High 32 address bits (bits 63:32) of buffer address that holds the *IPv6 NS table entry command* structure.
Data Address Low	31-29	Data Address Low	Data Address Low. Low 32 address bits (bits 31:0) of buffer address that holds the *IPv6 NS table entry command* structure.

Following is the IPv6 NS table entry command structure placed in the buffer pointed to by the descriptor.

Table 38-113.IPv6 NS Table Entry Command Structure

Name	Bytes.Bits	Value	Remarks
Table index	0-1	Table ID	16-bit table index ID 0, valid only to remove MAC 0 address entry (bit 8.1 is set).
Table index	2-3	Table ID	16-bit table index ID 1, valid only to remove MAC 1 address entry (bit 8.3 is set.
Table index	4-5	Table ID	16- bit table index ID 2, valid only to remove IPv6 0 address entry (bit 8.5 is set).
Table index	6-7	Table ID	16-bit table index ID 3, valid only to remove IPv6 1 address entry (bit 8.7 is set).
Control	8.0	Add MAC 0	Add MAC 0 address entry.
Control	8.1	Remove MAC 0	Remove MAC 0 address entry.
Control	8.2	Add MAC 1	Add MAC 1 address entry.
Control	8.3	Remove MAC 1	Remove MAC 1 address entry.
Control	8.4	Add IP 0	Add IP 0 address entry.
Control	8.5	Remove IPv6 0	Remove IPv6 0 address entry.
Control	8.6	Add IP 1	Add IP 1 address entry.
Control	8.7	Remove IPv6 1	Remove IPv6 1 address entry.
Control	9.0	Command Sequence Reference	0x0 for IPv6 0/1, 0x1 for IPv6 2/3.
Control	9.1	Initialize IPv6 Table	Set to 1b to initialize/clear the IPv6 address table, this bit has the same effect as removing all IPv6 addresses.
Control	9.2	Initialize MAC Table	Set to 1b to initialize/clear the MAC address table, this bit has the same effect as removing all MAC addresses.
Control	9.3	Enable NS	NS offload is enabled.
Control	9.4	Enable NS Directed	NS directed offload is enabled.
Control	9.7-9.5	Reserved	Must be 0x0, value is ignored.
MAC Address 0	10-15	MAC Address 0	Exact MAC Address 0 in Big Endian order.
MAC Address 1	16-21	MAC Address 1	Exact MAC Address 1 in Big Endian order.
MAC Address	22-27	Local MAC Address	Local MAC address in Big Endian order.
IPv6 addr 0	28-43	IPv6 Address	IPv6 address 0 in Little Endian order.
IPv6 addr 1	44-59	IPv6 Address	IPv6 address 1 in Little Endian order.

38.18.5.4 Set NS Proxy Table Entry Response

Description — Firmware response to the *Set NS Proxy Table Entry command*.

Response Type — Indirect response, Opcode, return value and address of the IPv6 NS table entry response structure are placed in the descriptor, all other values are inserted in the structure.

Table 38-114.Set NS Proxy Table Entry Response Descriptor Fields

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0x0	
Opcode	2-3	Opcode	0x0105.
Datalen	4-5	Data Length	Data length of buffer in bytes.
Return Value	6-7	Return Value	Return Value. Firmware supplies in the *Return value* field indication on the completion of the *Set NS Proxy Table Entry command*. 0x0 = No error. Others = Error detected in the command.
Cookie High	8-11	Cookie	Opaque value, copied by firmware from the *Set NS Proxy Table Entry command*.
Cookie Low	12-15	Cookie	Opaque value, copied by firmware from the *Set NS Proxy Table Entry command*.
Reserved	16-19	Reserved	Must be 0x0, value is ignored.
Reserved	20-23	Reserved	Must be 0x0, value is ignored.
Data Address High	24-27	Data Address High	Data Address High. High 32 address bits (bits 63:32) of buffer address that holds the *IPv6 NS table entry command* structure.
Data Address Low	31-29	Data Address Low	Data Address Low. Low 32 address bits (bits 31:0) of buffer address that holds the *IPv6 NS table entry command* structure.

Following an IPv6 NS table entry response structure is placed in the buffer pointed to by the descriptor.

Table 38-115.IPv6 NS Table Entry Response Structure

Name	Bytes.Bits	Value	Remarks
Table index	0-1	Table ID	16 bit table index ID for MAC address 0.
Table index	2-3	Table ID	16 bit table index ID for MAC address 1.
Table Index	4-5	Table ID	16 bit table index ID for IPv6 address 0.
Table Index	6-7	Table ID	16 bit table index ID for IPv6 address 1.
Control	8-9	Reserved	Must be 0x0, value is ignored.
MAC Address 0	10-15	MAC Address 0	Exact MAC address 0 in Big Endian order.
MAC Address 1	16-21	MAC Address 1	Exact MAC address 1 in Big Endian order.
Mac Address	22-27	MAC Address	Local MAC address in Big Endian order. Supplied in the *Set NS Proxy Table Entry command*.
IPv6 Addr 0	28-43	IPv6 Address	IPv6 address 0 in Little Endian order. Supplied in the *Set NS Proxy Table Entry command*.
IPv6 Addr 1	44-59	IPv6 Address	IPv6 address 1 in Little Endian order. Supplied in the *Set NS Proxy Table Entry command*.

38.19 Non-Volatile Memory Map

38.19.1 NVM Organization

38.19.1.1 Hierarchical NVM Map

Table 38-116 lists in a hierarchical order of all the modules and sub-modules that are defined in the 10 GbE controller's NVM map. They are referred to separately as NVM sections. All details concerning these sections are found in Section 6.2.

Note that the root section named Init Module (NVM header) is not listed in Table 38-74. Details about this section can be found in Section 38.19.2.

Table 38-116.Hierarchical NVM map (Sheet 1 of 4)

Address of Pointer in Parent Section	1st Level Section	2nd Level Section	3rd Level Section	Description
0x04	PHY Module			Auto load section of the internal PHY configuration.
0x06	RO PCIR Registers Auto-load			Contains RO parameters that configure the PCIe transaction layer (note that it is currently empty).
0x37	VLAN Configuration Block			Original factory default MAC addresses defined for LAN and RDMA.
0x39	Software RDMA MAC Address			MAC addresses to be used for RDMA
0x28	Reserved			Reserved
0x16	PBA Block			The PBA block contains the complete PBA number including the dash and the first digit of the 3-digit suffix.
0x30	PXE Setup Options			Setup options for the PXE driver that is defined per PCIe function.
0x31	PXE Configuration Customization Options			Configuration customization options for the PXE driver that is defined per PCIe function.
0x2F	VPD Module			Vital Product Data (VPD) loaded by an OEM. It contains RO and RW information about a NIC/ LOM.
0x17	Boot Configuration Block			Contains the required setup used for boot operations.
0x08	PCIR Registers Auto-load			Default setup to registers and internal memories that load on PCIR events.
0x38	POR Registers Auto-load			Default setup to registers that load on POR events.
0x3C	CORER Registers Auto-load			Default setup to registers and internal memories that load on CORER events.
0x3B	GLOBR Registers Auto-load			Default setup to registers that load on GLOBR events.
0x4F	Core Mem Config			Read margin settings to the device's core memories. RO module.
0x4A	Protocol Engine Auto-load Section			Default setup to registers and internal memories that load on CORER events related to the Protocol Engine.
0x48	EMP SR Settings			This section contains the modes of operation of the EMP that must be stored in the shadow RAM.

Table 38-116.Hierarchical NVM map (Sheet 2 of 4)

Address of Pointer in Parent Section	1st Level Section	2nd Level Section	3rd Level Section	Description
	0x05	SR PF Allocations		This section contains the allocation of resources per PF that must be stored in shadow RAM.
	0x0F	MNG MAC Addresses		This section contains the Pass Through LAN Ethernet MAC addresses, 4 addresses per port.
	0x10	PF MAC Addresses		This section contains the PF Ethernet MAC addresses, one address per PF.
0x07	Auto Generated Pointers			Pointers to type 1/2 words used by EMP and software. The module must be mapped to shadow RAM word address 0x7D80.
0x3E	PCIe ALT Auto-load			Modified factory default settings to registers that load on PCI reset events. The module must be mapped to shadow RAM word address 0x7E00. Any change performed to the module contents in shadow RAM is not dumped into the Flash memory.
0x04	PHY Analog			Configure the analog section of the SerDes PHY.
0x53	PE Settings			Default settings to PE registers.
0x09	EMP Global			This section contains two sub-sections: 1. Vendor Specific Settings: Settings for external PHY or modules. This part has a proprietary format in order to enable advanced PHY setting. 2. List of Qualified Modules: Parameter list of up to 16 modules. Per-module list holds OUI, revision and version numbers. These settings are loaded into the external devices via the MDIO interface.
0x0E	Manageability Module Header			This section contains parameters related to the manageability functionality such as connection type and others. It also points to sub-sections configuring the filters and the sideband interfaces.
	0x03	Pass Through Control Words Structure 0		This section contains the initial setting of the pass-through filters for a port. This section is used only in SMBus legacy pass-through mode. This section is repeated per port.
	0x04	Pass Through Control Words Structure 1		
	0x05	Pass Through Control Words Structure 2		
	0x06	Pass Through Control Words Structure 3		
	0x08	Flexible TCO Filter Configuration Structure		This section contains the setting of the manageability flex filters for all ports.
	0x07	Sideband Configuration Structure		This section describes the setting of the different pass-through interfaces (SMBus, NC-SI and MCTP).
	0x0B	OEM Section		This section is the header of the OEM specific data identifying the OEM for which this data is defined.
0x48	EMP SR Settings			This section contains the modes of operation of the EMP stored in shadow RAM

Table 38-116.Hierarchical NVM map (Sheet 3 of 4)

Address of Pointer in Parent Section	1st Level Section	2nd Level Section	3rd Level Section	Description
	0x05	PF SR Allocations		This section contains the allocation of resources per PF and the PF MAC address.
0x0F	EMP Settings			This section contains the modes of operation of the EMP.
	0x07	PHY Capability Data Structure 0		The PHY capabilities data structure contains all the parameters to control the internal and external PHY operation.For example, interface type and mode, etc. Data structure is repeated per port.
	0x08	PHY Capability Data Structure 1		
	0x09	PHY Capability Data Structure 2		
	0x0A	PHY Capability Data Structure 3		
	0x0B	External-to-Internal PHY Mapping 0		This section provides the internal PHY mode that is selected for each external PHY/module counter part. The 10 GbE controller provides multiple PHY interface for connecting different types of external PHYs and optical or copper modules. End users can plug in different types of modules into the SFP+ or QSFP+ connectors and choose from several connectivity options when using external PHYs. For example a 10GBASE-T PHY could use a a SFI or KR connection to connect to the 10 GbE controller. Firmware has to read the PHY or module ID and choose the appropriate interface to operate with that external module. Further, when working with an external 10GBASE-T PHY, auto-negotiation might result in several speed modes.For example, when a 10GBASE-T PHY can auto-negotiate to 1 GbE. This section is repeated per port.
	0x0C	External-to-Internal PHY Mapping 1		
	0x0D	External-to-Internal PHY Mapping 2		
	0x0E	External-to-Internal PHY Mapping 3		
	0x0F	LLDP Configuration		Default settings to the embedded LLDP agent.
	0x14	LLDP OEM TLVs		A set of fixed-structure LLDP TLVs for EMP use.
0x46	2nd Free Provisioning Area			Free area used as the new bank when updating 8 KB long modules that are mapped outside the shadow RAM.
0x05	Expansion/Option ROM (OROM)			It contains pre-boot code and settings read by BIOS. Pre-boot code is authenticated by BIOS during initialization before being executed by an EMP on update.
0x0C	PE Image			It contains the PE processor code.
0x0B	EMP Image			It contains the EMP processor code.
0x40	1st Free Provisioning Area			Free area used as the new bank when updating 1160 KB long modules.
0x42	Reserved 4th Free Provisioning Area			Free area to be used as the new bank when updating future 64 KB long modules mapped outside the shadow RAM.
0x44	Reserved 3rd Free Provisioning Area			Free area to be used as the new bank when updating future 16 KB long modules mapped outside the shadow RAM.
0x49	Feature Configuration			Holds the currently used set of selections for the current NVM (shadow RAM).

Table 38-116.Hierarchical NVM map (Sheet 4 of 4)

Address of Pointer in Parent Section	1st Level Section	2nd Level Section	3rd Level Section	Description
0x4D	Configuration Metadata			Adaptive NVM Configuration Meta Data for each feature.
0x4E	Immediate Value			Adaptive NVM Immediate Value. Holds the currently used values of the immediate fields. This section keeps copies of the values (shadow RAM).
0x4B	Reserved			Reserved.
0x50	Firmware Log Area			Scratch pad area used by firmware to log events.

38.19.2 NVM Header

Table 38-117 lists the format of the NVM header section. It includes words with a specific content and pointers to the first level of NVM modules. See Section 38.20 for more details.

- **Bold** items are pointers to modules.
- The Auth / RO column indicates the following:
  - — Auth — The pointed module is authenticated and the pointer is read-only
  - — RO — The pointed module and/or the pointer are read-only, or the word is read-only
  - — No — The word or the module is read/write

  A read-only item can be written in blank Flash programming mode.
- The CSR Format column indicates whether the pointed module uses the formats described in Section 38.19.2.1.
- The module in the shadow RAM column indicates whether the module is mapped into the basic banks mirrored into the internal shadow RAM. If marked as No, the most significant bit of the pointer (bit 15) must be set to 1b to indicate that the pointer value (in bits 14:0) is expressed in 4 KB sector units. Otherwise, it is expressed in word units.
- Maximum provisioned size indicates the following:
  - — Modules mapped in shadow RAM — How much has been accounted for the module in the shadow RAM, though the module can exceed this limit as long as the entire shadow RAM contents is not exceeding its size.
  - — Modules mapped outside shadow RAM — This is the maximum size the module can take because it has to fit within the associated free provisioning area.
  - — The maximum size provisioned for the NVM header itself is 256 words.

Table 38-117.NVM Header Map (Sheet 1 of 4)

Word Address	Pointed Module Name / Word Name	Auth/ RO	Accessed by	Loading Trigger	CSR Format	Module is in Shadow RAM	Typical Size	Maximum Provisioned Size
0x00	NVM Control Word 1	RO	EMP	POR				
0x01	RO Commands Version	RO	EMP	EMPR				
0x02	Reserved							
0x03	Reserved							

Table 38-117.NVM Header Map (Sheet 2 of 4)

Word Address	Pointed Module Name / Word Name	Auth/ RO	Accessed by	Loading Trigger	CSR Format	Module is in Shadow RAM	Typical Size	Maximum Provisioned Size
0x04	PHY Analog	RO	HW	POR	No	No	6 KB	8 KB
0x05	Expansion/Option ROM	Auth	BIOS/HW	**PCIR**[1]	No	No	496 KB	1160 KB
0x06	RO PCIR Registers Auto-load	RO	EMP	PCIR	Yes	Yes	64 bytes	128 bytes
0x07	Auto Generated Pointers	No	EMP/SW	POR	Yes	Yes	16 bytes	64 bytes
0x08	PCIR Registers Auto-load	RO	EMP	PCIR	Yes	Yes	512 bytes	1024 bytes
0x09	EMP Global	**RO**[2]	EMP	GLOBR	No	No	512 bytes	8 KB
0x0A	Reserved							
0x0B	EMP Image	Auth	EMP	EMPR	No	No	1248 KB	1248 KB
0x0C	PE Image	Auth	PE	CORER	No	No	384 KB	512 KB
0x0D	Reserved							
0x0E	Manageability	RO[4]	EMP	EMPR	No	Yes	1 KB	8 KB
0x0F	EMP Settings	RO[4]	EMP	CORER[3]	No	No	1 KB	8 KB
0x10	SW Compatibility Word 1	No	SW/BIOS	POR[2]				
0x11	SW Compatibility Word 2	No	SW/BIOS	POR[2]				
0x12	SW Compatibility Word 3	No	SW/BIOS	POR[2]				
0x13	SW Compatibility Word 4	No	SW/BIOS	POR[2]				
0x14	SW Compatibility Word 5	No	SW/BIOS	POR[2]				
0x15	PBA Flag	No	SW/BIOS	POR[2]				
0x16	PBA Block	No	SW/BIOS	POR[4]	No	Yes	12 bytes	12 bytes
0x17	Boot Configuration	No	BIOS	POR[7]	No	Yes	3072 bytes	3072 bytes
0x18	NVM Image Revision	No	SW/BIOS/ EMP	POR				
0x19	SW Reserved Word 2	RO	SW/BIOS	POR[2]				
0x1A	SW Reserved Word 3	RO	SW/BIOS	POR[2]				
0x1B	SW Reserved Word 4	RO	SW/BIOS	POR[2]				
0x1C	SW Reserved Word 5	RO	SW/BIOS	POR[2]				
0x1D	SW Reserved Word 6	RO	SW/BIOS	POR[2]				
0x1E	SW Reserved Word 7	RO	SW/BIOS	POR[2]				
0x1F	SW Reserved Word 8	RO	SW/BIOS	POR[2]				
0x20	SW Reserved Word 9	RO	SW/BIOS	POR[2]				
0x21	SW Reserved Word 10	RO	SW/BIOS	POR[2]				
0x22	SW Reserved Word 11	RO	SW/BIOS	POR[2]				
0x23	SW Reserved Word 12	RO	SW/BIOS	POR[2]				
0x24	SW Reserved Word 13	RO	SW/BIOS	POR[2]				
0x25	SW Reserved Word 14	RO	SW/BIOS	POR[2]				
0x26	SW Reserved Word 15	RO	SW/BIOS	POR[2]				
0x27	SW Reserved Word 16	RO	SW/BIOS	POR				
0x28	SW Reserved Word 17	No	SW/BIOS/ EMP	CORER	No	Yes	98 bytes	128 bytes
0x29	SW Reserved Word 18	RO	SW/BIOS	POR[2]				
0x2A	SW Reserved Word 19	RO	SW/BIOS	POR[2]				

Table 38-117.NVM Header Map (Sheet 3 of 4)

Word Address	Pointed Module Name / Word Name	Auth/ RO	Accessed by	Loading Trigger	CSR Format	Module is in Shadow RAM	Typical Size	Maximum Provisioned Size
0x2B	SW Reserved Word 20	RO	SW/BIOS	POR[2]				
0x2C	SW Reserved Word 21	RO	SW/BIOS	POR[2]				
0x2D	SW Reserved Word 22	RO	SW/BIOS	POR[2]				
0x2E	SW Reserved Word 23	RO	SW/BIOS	POR[2]				
0x2F	VPD Area	**RO**[5]	EMP/VPD SW	**POR**[7]	No	Yes	256 bytes	1024 bytes
0x30	PXE Setup Options	No	BIOS	**POR**[7]	No	Yes	34 bytes	64 bytes
0x31	PXE Configuration Customization Options	No	BIOS	**POR**[7]	No	Yes	34 bytes	64 bytes
0x32	PXE Version	No	BIOS	POR[2]				
0x33	IBA Capabilities	No	BIOS	POR[2]				
0x34	SW Reserved Word 24 - Original EETRACK ID1	No	BIOS	POR[2]				
0x35	SW Reserved Word 25 - Original EETRACK ID2	No	BIOS	POR[2]				
0x36	iSCSI Option ROM Version	No	BIOS	POR[2]				
0x37	Software Alternate MAC Address	No	SW/BIOS/ BMC	POR[7]	No	Yes	218 bytes	256 bytes
0x38	POR Registers Auto-load	No	EMP	POR	Yes	Yes	1536 bytes	2048 bytes
0x3A	Reserved for EMPR Registers Auto-load	RO	EMP	POR	Yes	Yes	0 bytes	32 v
0x3B	GLOBR Registers Auto-load	No	EMP	GLOBR	Yes	Yes	800 bytes	1024 bytes
0x3C	CORER Registers Auto-load	No	EMP	CORER	Yes	Yes	41300 bytes	45056 bytes
0x3D	PHY Configuration Scripts	RO	EMP	POR	No	No		
0x3E	PCIe ALT Auto-load	RO[6]	EMP	PCIR[2]	Yes	Yes	512 bytes	1KB
0x3F	SW Checksum	No	SW					
0x40	1st Free Provisioning Area	RO[4]	EMP	EMPR[2,3]	No	No	1160 KB	1160 KB
0x41	1st Free Provisioning Area Size	RO	EMP	EMPR				
0x42	Reserved 4th Free Provisioning Area	RO[4]	EMP	EMPR[2,3]	No	No	64 KB	64 KB
0x43	Reserved 4th Free Provisioning Area Size	RO	EMP	EMPR				
0x44	Reserved 3rd Free Provisioning Area	**RO**[4]	EMP	**EMPR**[2,3]	No	No	16 KB	16 KB
0x45	Reserved 3rd Free Provisioning Area Size	RO	EMP	EMPR				
0x46	2nd Free Provisioning Area	**RO**[4]	EMP	**EMPR**[2,3]	No	No	8 KB	8 KB
0x47	2nd Free Provisioning Area Size	RO	EMP	EMPR				
0x48	EMP SR Settings	No	EMP	**EMPR**[7]		Yes	82 bytes	128 bytes
0x49	Feature Configuration	RW[2]	EMP	EMPR	No	Yes		
0x4A	Protocol Engine Auto-Load	RO	EMP	CORER[8]	Yes	Yes	1 KB	1 KB
0x4D	Configuration Metadata	RO	EMP	EMPR	no	No	100 KB	128 KB
0x4E	Immediate Fields	RO	EMP	EMPR	No	Yes		

Table 38-117.NVM Header Map (Sheet 4 of 4)

Word Address	Pointed Module Name / Word Name	Auth/ RO	Accessed by	Loading Trigger	CSR Format	Module is in Shadow RAM	Typical Size	Maximum Provisioned Size
0x4F	Core Mem Config	RO	EMP	**POR**	Yes	Yes	1 KB	1 KB
0x50	Firmware logging Area	RO	EMP		No	No	512 KB	512 KB
0x51	Firmware logging Area size	RO	EMP		No			
0x52	POR CORER Autoload	RW	EMP POR	Yes	Yes			128 bytes
0x53	PE Settings	No	PE	CORER	No	No	16 bytes	8 KB
0x54	Reserved	RO	EMP	POR	No			
0x55	Soft SKUs	RO	EMP		No			
0x56 - 0xFF	Spare NVM Header Words	No		*POR*[2]				

Notes:

1. Only the pointer is loaded not the module's contents.
2. RO pointer, while the pointed area is RW.
3. Some items load on CORER, others on EMPR.
4. Pointer and module contents are loaded only into shadow RAM.
5. VPD pointer is a RO pointer, regardless to its own value and to the value of the GLPCI_CAPCTRL.VPD_EN bit. If VPD Write Enable bit is cleared in NVM, VPD area cannot be modified via the NVM Update Admin command, but only via the VPD register set in the PF config space.
6. The module must be mapped to the shadow RAM word address 0x7E00. Any change performed to the module contents in shadow RAM is not dumped into Flash memory. Software cannot directly modify the module contents via NVM update commands, but only indirectly via the Alternate Structure admin commands.
7. Pointer and module contents.
8. Loaded after a CORER and the PE is enabled.

38.19.2.1 Structure Of Hardware Modules

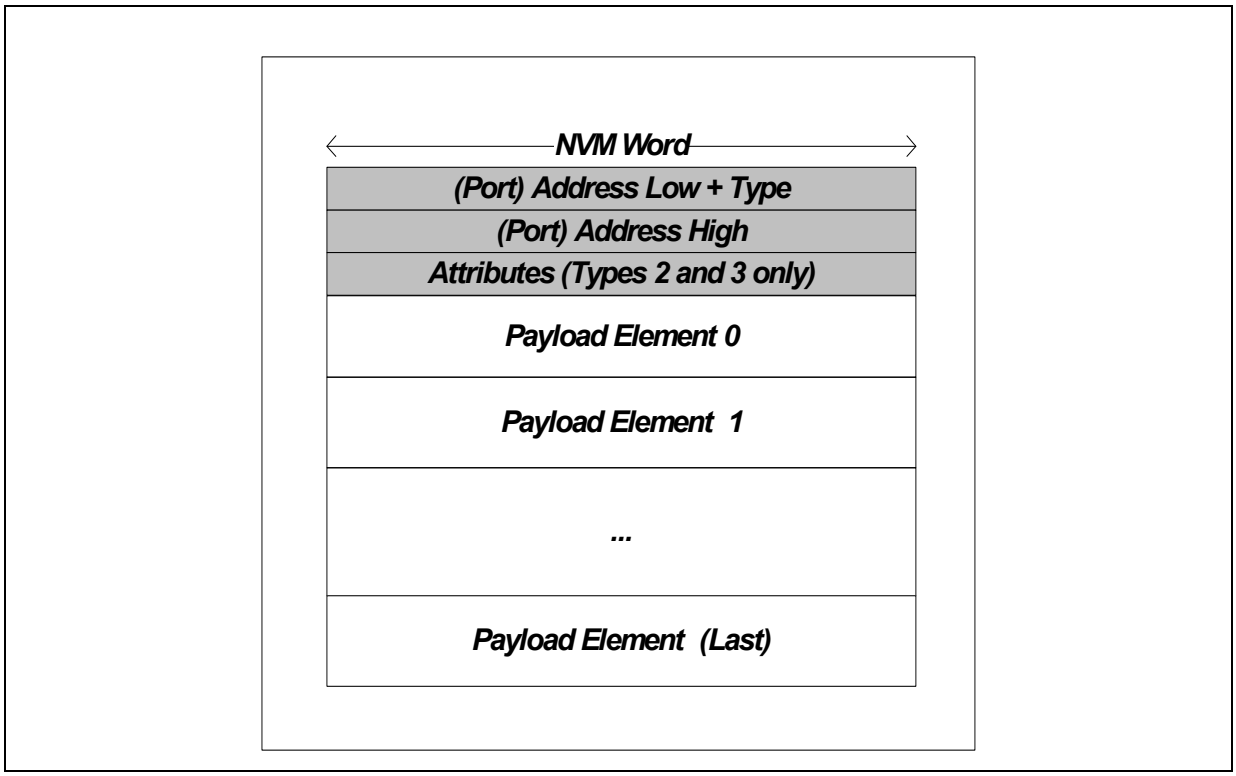
An NVM module read by the EMP to configure hardware (as listed Table 38-75) is built according to one of the following structures:

- Fixed functionality structure — Each NVM word is allocated to one or more fixed fields and the contents of the NVM word are loaded to fixed CSRs in the 10 GbE controller. In Table 38-75, these modules are referred as CSR format = No modules.
- Flexible functionality module — The structure defines the device address or addresses into which the NVM words are loaded. Therefore, the module might be used for different purposes based on its contents. In Table 38-75, these modules are referred as CSR format = Yes modules.

The remainder of this section describes the structure of the contents of a flexible functionality module. The general structure of the module is shown in Figure 38-31.

A CSR format module can be made of a mix of repeating Type 1, 2, 3 and 4 segments. Each segment is described by its own header (the shaded fields in Figure 38-31).

Figure 38-31.Structure of a Flexible Functionality Module



Description of the fields in Figure 38-31:

- **(Port) Address** — A 28-bit value that defines the starting address to which data Dwords are loaded. Serves as either a direct address to load into (Types 1 and 2) or as an indirect address referred to as a port address (Type 3 and 4). A port is made of a 32-bit address register followed by data registers of total size width. Address low is a 12-bit field mapped to word[15:4].
- **Type** — A 4-bit command field mapped to word[3:0] bits that defines the type of flexible module. Descriptions are as follows:
  - — Type 1 (0001b) — Loads a single 32-bit data segment
  - — Type 2 (0010b) — Load a set of 32-bit data Dwords into consecutive addresses
  - — Type 3 (0011b) — Loads a set of 32-bit data Dwords through an address port
  - — Type 4 (0100b) — Loads, through an address port, a sequence of data blocks into arbitrary addresses
- **Attributes** — A 16-bit optional word that defines attributes of the module
  - — **Width** — A 3-bit field mapped to word[2:0] bits that describes the width (in 32-bit Dwords) of a data element
    - 000b = Data is 32-bit wide
    - 001b = Data is 64-bit wide
    - 010b = Data is 128-bit wide
    - 011b = Data is 256-bit wide
    - Else = Reserved
  - — **Skip** — A 2-bit field mapped to word[4:3] bits that describes the number of address bytes to be skipped for getting the address of the next payload element (Type 2) or port register involved (Type 3 or 4).

- 00b = Skip 4 bytes
- 01b = Skip 8 x 4 bytes = 32 bytes
- 10b = Skip 32 x 4 bytes = 128 bytes
- 11b = Reserved

— **Length** — An 11-bit field mapped to word[15:5] bits that contains the number of data elements, each of Width bits

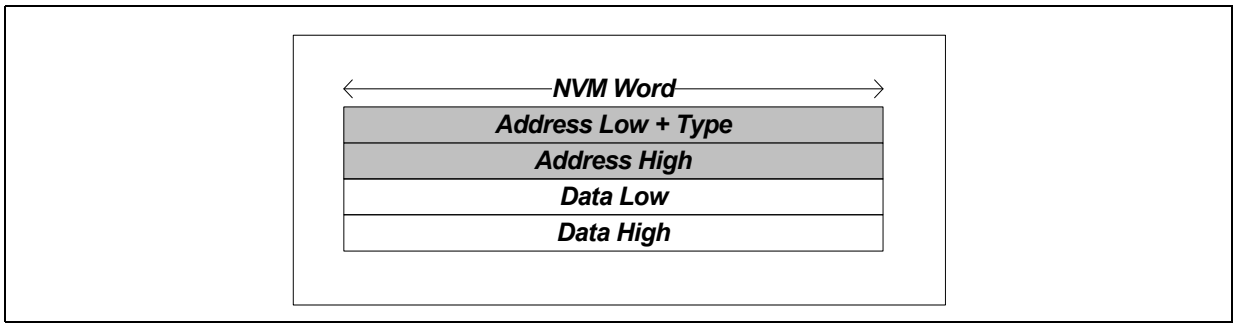
- **Payload Elements** — A sequence of address or data elements to be loaded into the 10 GbE controller. The structure of the *Payload* field varies with each type and is described in the sections that follow.

38.19.2.1.1 Type 1 Module

Used to specify a single 32-bit data segment. The *Address* field defines the address into which the 32-bit data is loaded. See Figure 38-32.

The common use of a Type 1 module is to concatenate a list of such structure to load a set of 32-bit values into various CSRs.

Figure 38-32.Structure of a Type 1 Module



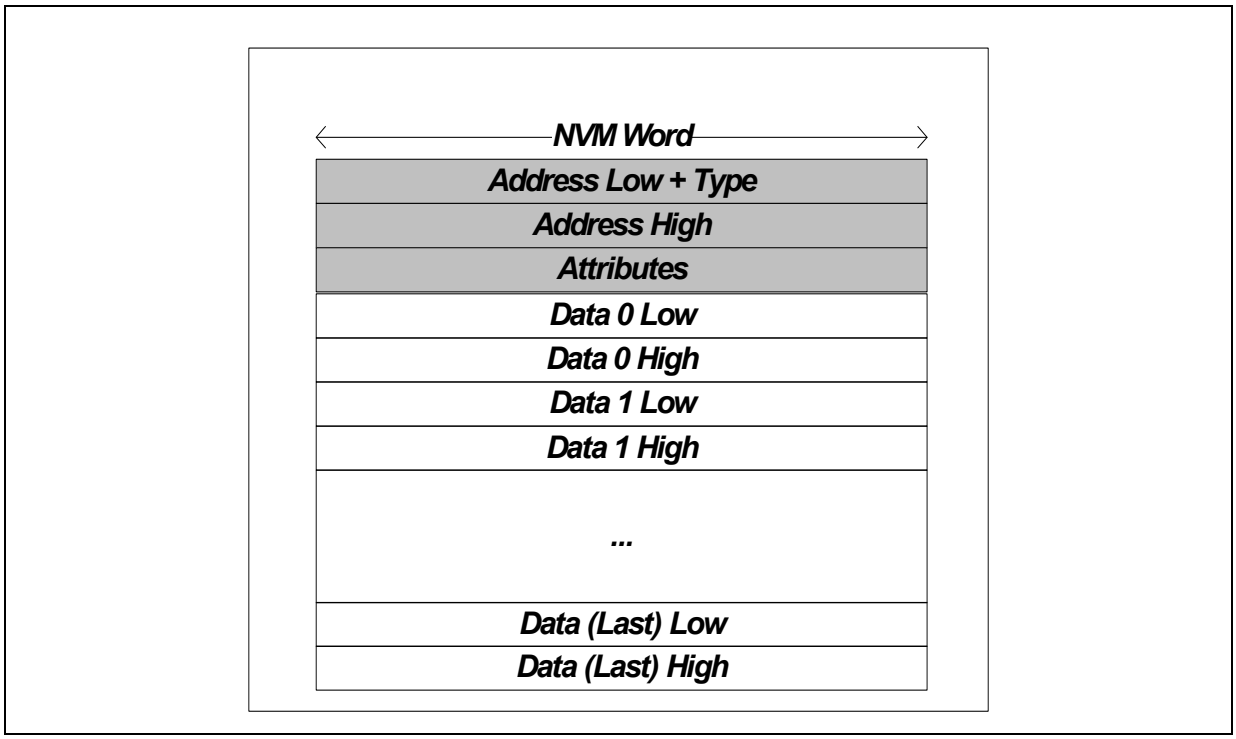
38.19.2.1.2 Type 2 Module

Used to load a sequence of data into consecutive addresses, such as a memory array. See Figure 38-33.

The *Address* field defines the first address to be loaded. Data is loaded as width-size elements into consecutive addresses. The *Attributes* field applies the following information:

- Width = 000b
- Skip = see previous description
- Length = see previous description

Figure 38-33.Structure of a Type 2 Module



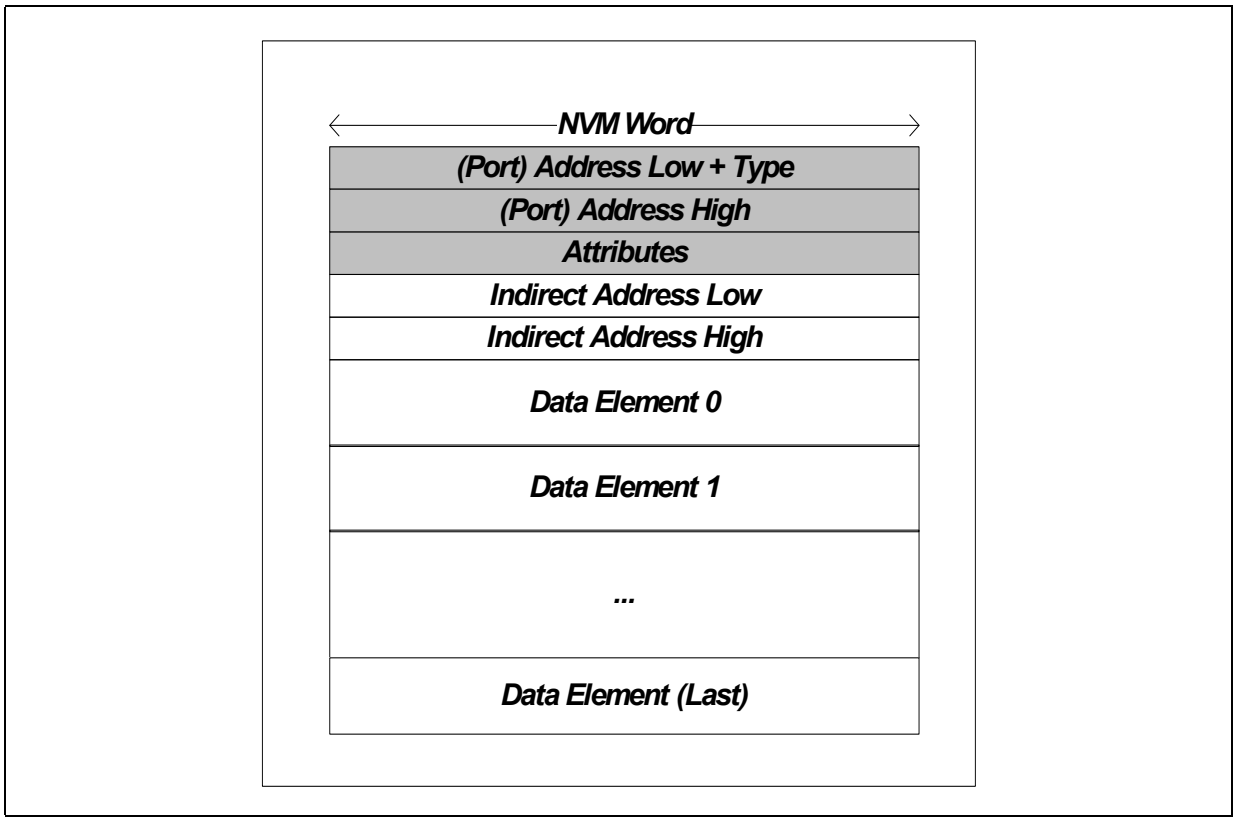
38.19.2.1.3 Type 3 Module

Used to load a sequence of data through an address port.

Figure 38-34 shows the structure of a Type 3 module. The following fields have special values:

- The ***Port Address*** field defines the address of the port address register through which address/data is loaded. The Port Address register is followed by the Port Data registers.
- The ***Attributes*** field applies with the following fields:
  - Width — The size of a single data element written into the port during one port load cycle
  - Length — Number of data elements written through the port
  - Skip — Encodes the number of bytes between the addresses of the port registers involved
- The ***Indirect Address*** field defines the first consecutive address to write to. It is written into the Port Address register for the first item and from then on the indirect address is increased (by the 10 GbE controller) with each write of a new data element from the list. Consecutive data elements are written into the port.
- The ***Data Elements*** field are made of length elements, each of the width DWords. Each port load cycle handles one data element.

Figure 38-34.Structure of a Type 3 Module



The following figures describe two cases of using a Type 3 module:

- Figure 38-35 shows a case of loading 32-bit values into a Global (GL) port.
- Figure 38-36 shows a case of loading 64-bit values into a PRT port.

Figure 38-35.Access Through a Type 3 Module of 32-Bits Wide Data Elements

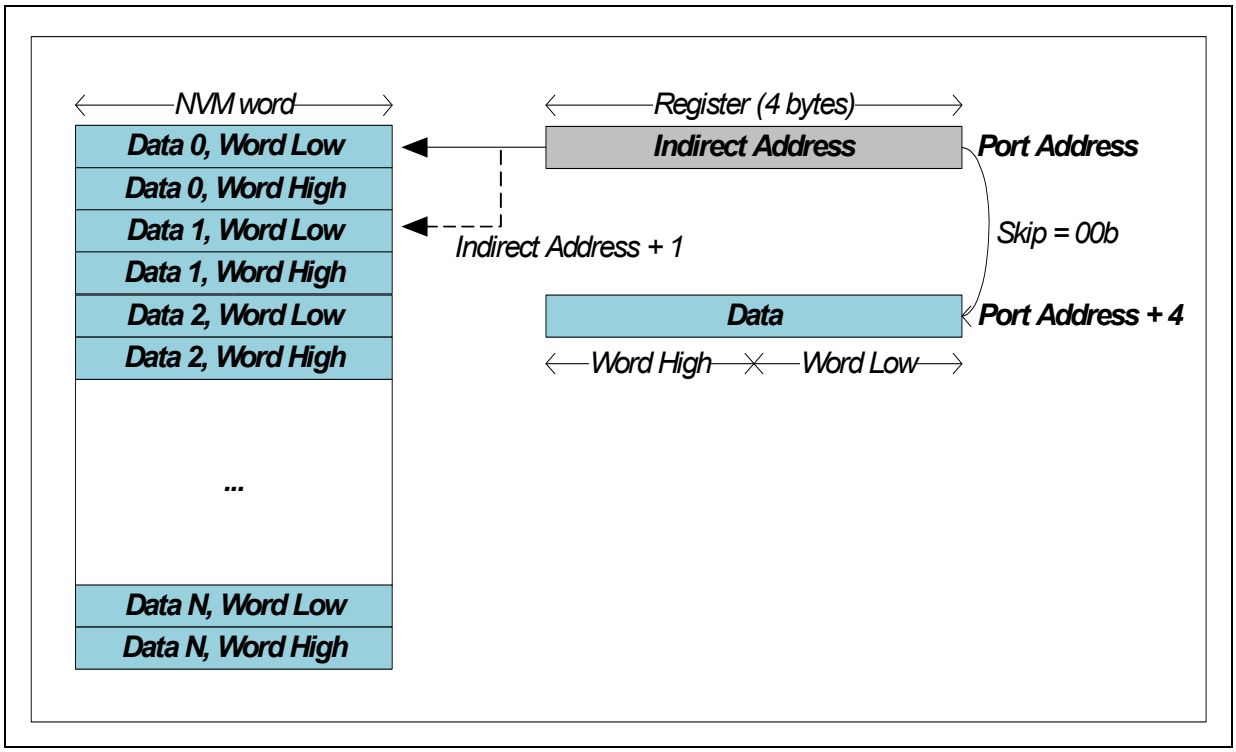
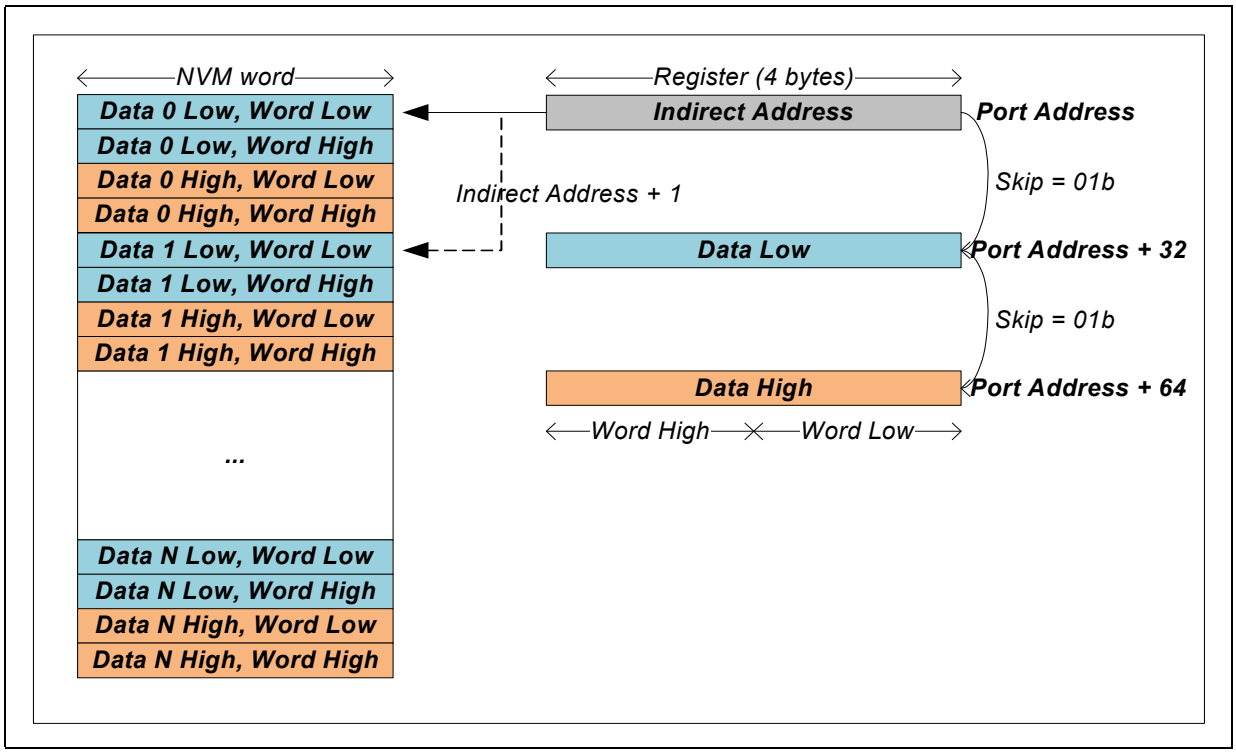


Figure 38-36.Access Through a Type 3 Module of 64-Bits Wide Per Port Data Elements



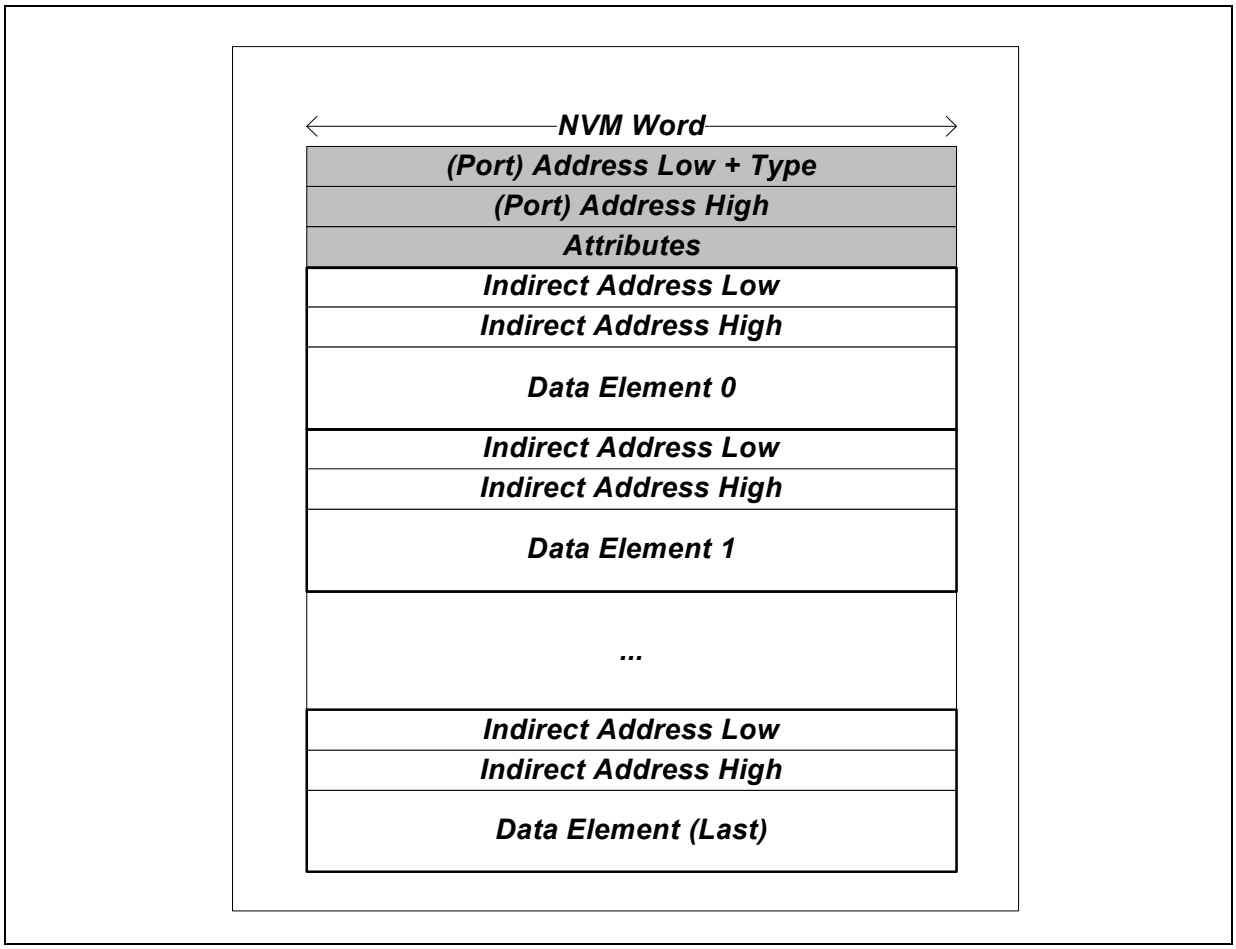
38.19.2.1.4 Type 4 Module

Used to load, through an address port, a sequence of scattered data elements at arbitrary addresses.

Figure 38-37 shows the structure of a Type 4 module. The following fields have special values:

- The ***Port Address*** field defines the address of the port address register through which address/data is loaded. The port address register is followed by the port data registers.
- The ***Attributes*** field applies with the following fields:
  - — Width — The size of a single data element written into the port during one port load cycle
  - — Length — Number of data elements written through the port, not including the indirect address words (the words colored in grey in Figure 38-37 and Figure 38-38).
  - — Skip — Encodes the number of bytes between the addresses of the port registers involved
- The ***Indirect Address*** fields define the address that the following data element is written into. It is written into the Port Address register.
- The ***Data Element*** is written into the port data register. The sequence of writing (the indirect address and its data element) into the port is repeated per each data element (length times).

Figure 38-37.Structure of a Type 4 Module



The following figures show two cases of using a Type 4 module:

- Figure 38-38 shows a case of loading 32-bit values into a global (GL) port.
- Figure 38-39 shows a case of loading 64-bit values into a PRT port.

Figure 38-38.Access Through a Type 4 Module of 32-Bits Wide Data Elements

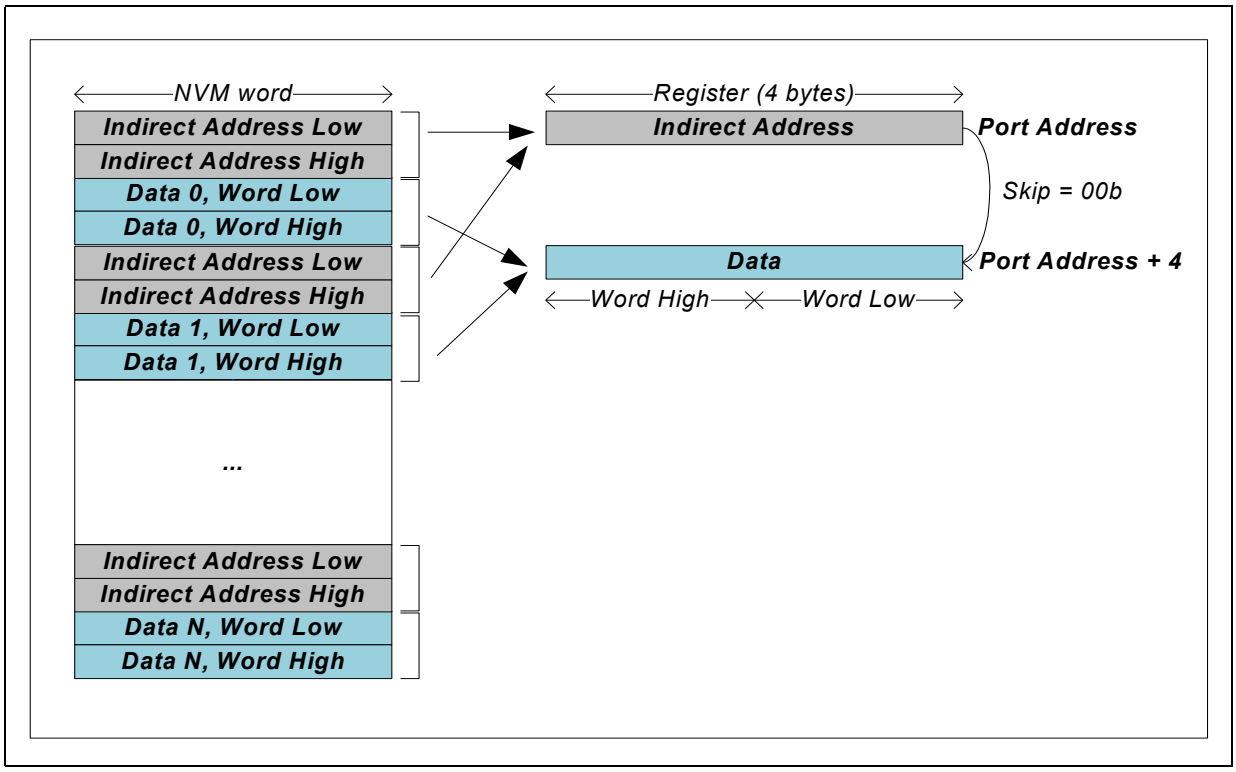
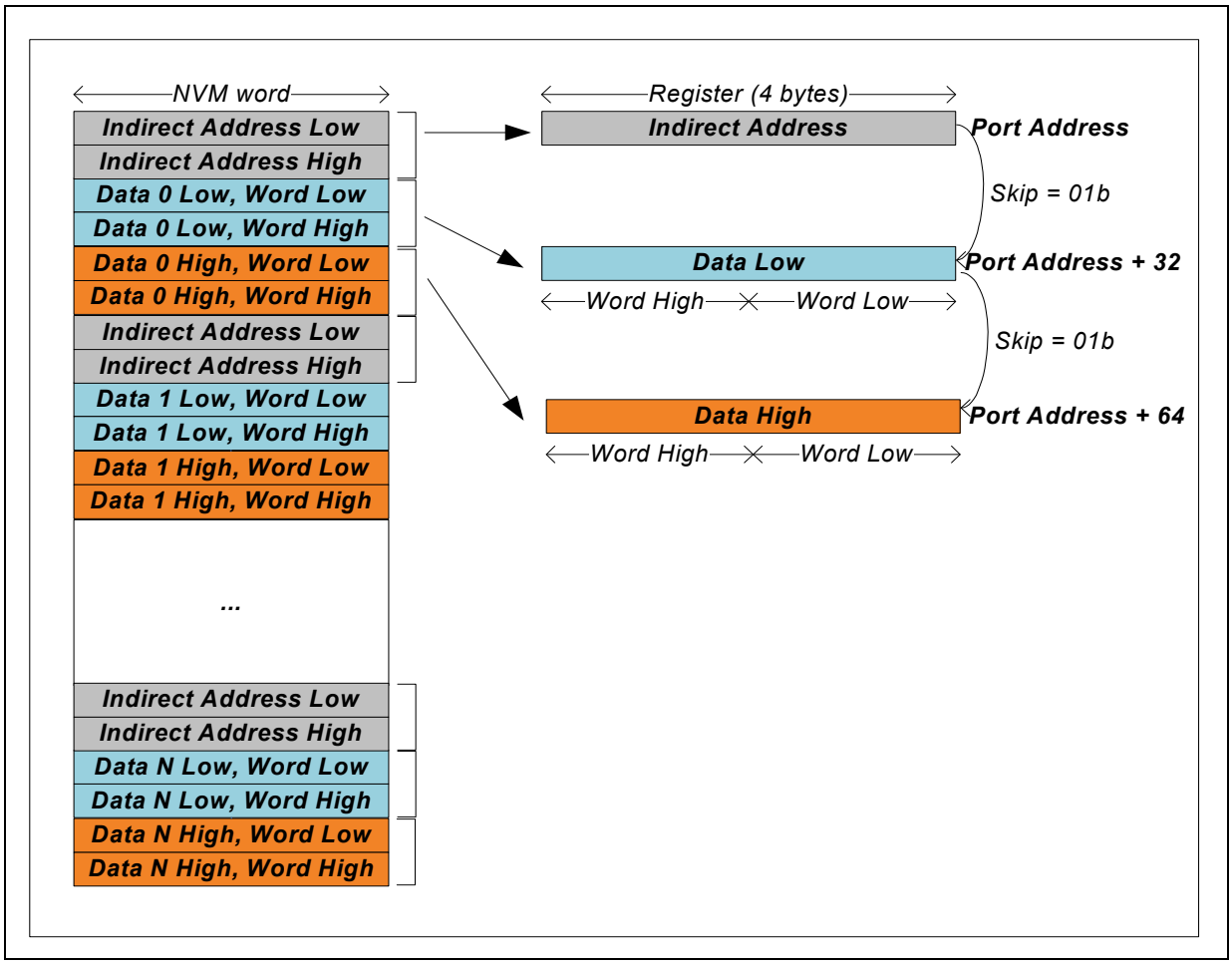


Figure 38-39.Access Through a Type 4 Module of 64-Bits Wide Per Port Data Elements



38.19.2.1.5 Auto Generated Pointers

Type 1 and Type 2 items are generated by the auto map generation tool from the project's register database. These items are automatically mapped into the Registers Auto-load NVM modules according to their loading trigger (see Table 38-75) and with no possibility to perform manual changes. Until the list of Type 1/2 items stabilizes, each invocation of the tool might lead to a different mapping offset in the destination module.

EMP code and software tools cannot accommodate these variations in mapping offset, and hence, they need a fixed method for accessing some predefined Type 1/2 items. The auto-generated pointers module is automatically defined by the auto map generation tool for this purpose. It is pointed to by NVM word 0x07. It lists the mapping address in NVM of the pre-defined Type 1/2 items via a 2-word structure per item.

The word address in shadow RAM of an item is given by the sum of the contents of its two associated words: pointer + offset. If the item is relative to an array of registers and/or to a register with a scope different than global, the offset is given for the first data word of the Type 1/2 array, for array instance [0] and scope instance [0].

In the auto-generated pointers module, the location of the 2-word structures relative to an item is made invariant along the project's life. The alias name of the Type 1/2 register appears in the names of the two words in the structure.

Following is the list of auto-generated pointers, listed in the order by which they appear in the module:

1. PFPM_APM
2. PRTPM_GC
3. GLGEN_STAT
4. GLPCI_SERL
5. GLPCI_SERH
6. PRTGL_SAL
7. PRTGL_SAH
8. GLPCI_CAPSUP
9. PRTDCB_MFLCN
10. PRTDCB_FCCFG
11. PFGEN_PORTNUM
12. PFPCI_FUNC2
13. PFPCI_CLASS
14. PF_VT_PFALLOC
15. GLGEN_PCIFCNCNT
16. GLPCI_REVID
17. PFPCI_DEVID
18. GLPCI_SUBVENID
19. PFPCI_SUBSYSID
20. GLPCI_VENDORID
21. GLPCI_CNF2

38.19.2.2 NVM Integrity Checks By Software

This section describes the NVM integrity fields inserted in the Product Name NVM map to provide a basic detection of a faulty Flash part.

A software checksum check is performed by the PF driver just after completion of its initialization sequence. It covers only modules mapped into shadow RAM. If the checks fails, another try is attempted, and if it fails again, the PF driver disables Tx/Rx operations with the 10 GbE controller.

Modules mapped outside the shadow RAM include a CRC8 field. EMP is responsible to check the CRC8 of the modules it loads/updates from/to Flash. It also provides software tools with the ability to check the integrity of these modules upon request because the PF driver does not include this check in its initialization sequence.

38.19.2.2.1 Software Checksum

The *Software Checksum* field covers the entire 64 KB shadow RAM contents (reserved words included) with the exception of the VPD area and to the PCIe ALT auto-Load module, which are skipped.

The *Software Checksum* word is located at word 0x3F. Its value is computed such that after adding all the covered words, including the *Software Checksum* word itself, the sum is 0xBABA.

Each time software tools are modifying one of these areas, it must update the *Software Checksum* field accordingly.

Each time EMP is updating the contents of one of these areas by its own initiative (not as part of an NVM Update AQ command) or for handling an MC command received over SMBus or NC-SI, it must update the *Software Checksum* field accordingly.

The CRC polynomial used is: X^8+X^2+X+1.

Similarly, the following modules include CRC8 fields in their header and in the headers of their sub-modules:

- PE settings module
- EMP global module
- Manageability module
- EMP settings module

EMP is responsible to check CRC8 validity of these modules (and their included sub-modules) on every EMPR, excepted to PE settings module, which is handled by PE on CORER events. If CRC8 is invalid after two tries (every try takes 260 ms for a 800 KB long EMP image), EMP must report the error in the GL_MNG_FWSM.EXT_ERR_IND and CRC_ERROR_MODULE register fields by posting to the MC an AEN # 0x82 - NVM error (enable bit = 18) over NC-SI and by using the Status Data Byte 2[5] in the SMBus alert.

If the CRC8 error occurred on the EMP code, ROM-EMP must report the error only to the host by setting a bit in a register and must not run the code loaded from RAM. It must also open the hardware NVM security to enable fixing the issue via software tools.

If the CRC8 error occurred on the manageability module, NC-SI/SMBus interfaces are disabled.

If the CRC8 error occurred on the EMP global module then the settings are not loaded to the PHY.

CRC8 of PE Image and PE Settings modules are checked by PE itself. When an error occurs on PE Image, the PE processor is held in reset and the error is reported in GLPE_FWLDSTATUS register. If the error occurs on PE Settings, the settings are not used and the error is reported in GLPE_FWLDSTATUS register.

Each time software tools are updating one of these modules, they must update its CRC8 field accordingly.

EMP must check the concerned CRC8 fields before committing an NVM Update command that was explicitly addressed for one of these modules. If a CRC8 field read from the free provisioning area of the Flash is not valid, the command completes with the EIO (Flash defect status).

38.19.2.2.2 NVM Integrity Summary

Table 38-118.NVM Integrity Summary Table

NVM Module	Integrity Check	When[1]	By	Number of Tries	Action on Error
Shadow RAM (excluding VPD area)	Checksum	Driver Init	PF Driver	2	Disable Rx/Tx paths for the host (but not for MC) and report the failure to user/ admin.
VPD area	No	N/A	N/A	N/A	N/A
PCIe Analog	CRC8	EMPR	EMP	2	Report the failure in GL_MNG_FWSM.EXT_ERR_IND. Post to the MC an AEN # 0x82 - NVM error (enable bit = 18) over NC-SI and then use the Status Data Byte 2[5] in the SMBus alert.
PHY Analog	CRC8	EMPR	EMP	2	Report the failure in GL_MNG_FWSM.EXT_ERR_IND. Post to the BMC an AEN # 0x82 - NVM error (enable bit = 18) over NC-SI and then use the Status Data Byte 2[5] in the SMBus alert.
PE Settings	CRC8	CORER	PE	2	PE reports the failure by setting bits in GLPE_FWLDSTATUS. PE will not use the settings read from this module.
EMP Global	CRC8	EMPR	EMP	2	Report the failure in GL_MNG_FWSM.EXT_ERR_IND. Post to the MC an AEN # 0x82 - NVM error (enable bit = 18) over NC-SI and then use the Status Data Byte 2[5] in the SMBus alert. Do not load the data into the PHY.
Manageability	CRC8	EMPR	EMP	2	Report the failure in GL_MNG_FWSM.EXT_ERR_IND. Disable NC-SI/SMBus.
EMP Settings	CRC8	EMPR	EMP	2	Report the failure in GL_MNG_FWSM.EXT_ERR_IND. Post to the BMC an AEN # 0x82 - NVM error (enable bit = 18) over NC-SI and then use the Status Data Byte 2[5] in the SMbus alert.
Option ROM	Yes	Boot Time	BIOS		
PE Image	CRC8	CORER	PE	1	PE reports the failure by setting bits in GLPE_FWLDSTATUS. PE is held in reset state.
EMP Image	CRC8	EMPR	ROM-EMP	2	Report the failure in GL_MNG_FWSM.EXT_ERR_IND. Do not run the RAM code. Open hardware NVM security by clearing the FLA.LOCKED bit. Check CRC8 before committing an EMP/PE image update. In case of an CRC8 error, report the error in the AQ completion and reject the update.

Notes:

1. Besides the integrity check performed by EMP before committing NVM Update commands.

38.19.2.3 Header of NVM Modules

38.19.2.3.1 Header of all NVM Modules Mapped to Shadow RAM

Modules read by hardware do not contain pointers to sub-modules and they are not authenticated.

Table 38-119.NVM Header of Modules Mapped to Shadow RAM

Number of Words	Field or Segment Name	Description and Comments
1	Module Length	Length of the module contents expressed in words (*Module Length* field excluded). It must be set to N. Modules are size limited to the size of the shadow RAM (64 KB).
N	Word 1	
	Word 2	
	...	
	Word N	

38.19.2.3.2 Header/Trailer of Authenticated NVM Modules

This section concerns the following modules:

1. EMP image (pointed by NVM word 0x0B).
2. PE image (pointed by NVM word 0x0C).
3. Option ROM (pointed by NVM word 0x05).

For the last module, the first 330 words listed in Table 38-120 are mapped at the end of the area allocated to the module (at its trailer), though for the sake of the authentication, these words are mapped at the module's header, as listed in Table 38-120.

Since authenticated modules are by definition, they cannot be modified *in the fields because* no holes are present in such modules.

In Table 38-120, fields colored in cyan are protected by the authentication signature.

Table 38-120.Header of Authenticated NVM Modules (Sheet 1 of 2)

Number of Words	Field or Segment Name	Description and Comments
64	CSS Header	
128	RSA Public Key	This field is skipped due to SHA256 Hash computing.
2	RSA Exponent	This field is skipped due to SHA256 Hash computing.
128	Encrypted SHA256 Hash	This field is skipped due to SHA256 Hash computing.
1	10 GbE Controller Blank NVM Device ID	A unique Intel-provided device ID that identifies the 10 GbE controller controller among other Intel controllers. It must be set to 0x3A84 in the 10 GbE controller.
2	Max Module Area	It is the maximum Flash area expressed in words that can be used by the module, starting from CSS header (included). It is set to 580 K words (1160 KB) for an EMP and PE image modules and to 4 K words (8 KB) for other authenticated modules.
2	Current Module Area	It is the Flash area expressed in words that is currently used by the module, starting from CSS header (included).

Table 38-120.Header of Authenticated NVM Modules (Sheet 2 of 2)

Number of Words	Field or Segment Name	Description and Comments
1	Module Format Version + CRC8	Bit 15 = CRC8 field is used. Set to 1b if a CRC8 is computed over the module, set to 0b otherwise. It must always be set to 0b in the OROM module. Bits 14:8 = Module format version. Set to 0x02touse the currently-defined format. Bits 7:0 = CRC8 value computed over the entire area allocated to the module (1160 KB for EMP image), starting from CSS header (excluding the fields not covered by the RSA authentication signature) and including all the remaining bytes of the area (excluding this word that is skipped for the sake of CRC8 computing.
1	Code Revision	Bits 15:8 = Major revision number. Bits 7:0 = Minor revision number. Major revision number of EMP and of the PE images shall match one to the other. Otherwise PE is maintained disabled by EMP. For other authenticated modules it is just the image revision number (not necessarily referring to any embedded code).
1	Reserved Spare Word	Must be zeroed.
2 or 1	Parent Module Length	Length of the parent module contents expressed in words, module header and *Parent Module Length* field excluded. It excludes all the descendant modules. Modules read by firmware are NOT size limited to 128 KB. • For modules parsed by firmware (EMP image and PE Images), this length field is two words long and it covers for the firmware code, its padding words, and for the EMP image also the last 4 KB sector reserved for the RO commands section. It must be set to 0x0006DEB4. • For modules parsed by hardware (PCIe analog and PHY analog modules), this length field is one word long and it does not include padded words.
N	Parent Word 1	Firmware module content formatting is specific to each module.
	Parent Word 2	
	...	
	Parent Word N	

38.19.2.4 Trailer of the EMP Image Module

In Table 38-121, fields colored in cyan are protected by the authentication signature.

The last 4 KB sector of the EMP image has the following format:

Table 38-121.RO Commands Section Format (Sheet 1 of 2)

Number of Words	Field or Segment Name	Description and Comments
1	RO Commands Version	Default is 0xFFFF, which means the section is empty and the remaining words are discarded.
1	10 GbE Controller Blank NVM Device ID	A unique Intel-provided device ID that identifies the 10 GbE controller among other Intel controllers. It must be set to 0x3A84 in the 10 GbE controller.
1	Minimum EMP Code Revision	Minimum EMP code revision number required for being able to parse the RO commands section. It must be lower or equal to the code revision number read from the EMP image header listed in Table 38-120.
1	RO Commands Length	Length in words (N), starting from next word.

Table 38-121.RO Commands Section Format (Sheet 2 of 2)

Number of Words	Field or Segment Name	Description and Comments
N	RO Commands Word 1	Format of the RO commands is described in Section 38.19.2.4.1.
	RO Commands Word 2	
	...	
	RO Commands Word N	

38.19.2.4.1 Format of the RO Commands

The RO command words can contain the following two structures:

1. Shadow RAM Word Write Command (2 words)
2. CSR Write Command (4 words)

Each command starts with a type field.

Table 38-122 describes the different commands types:

Table 38-122.RO Commands Types

Type	Description
xxx1b	Word auto load.
0010b	CSR auto load.
Other	Invalid type, parsing is stopped here.

Figure 38-40.Shadow RAM Word Write Command

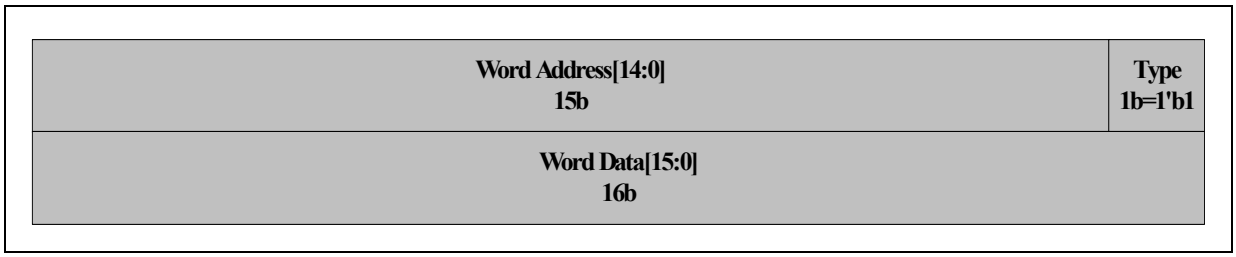
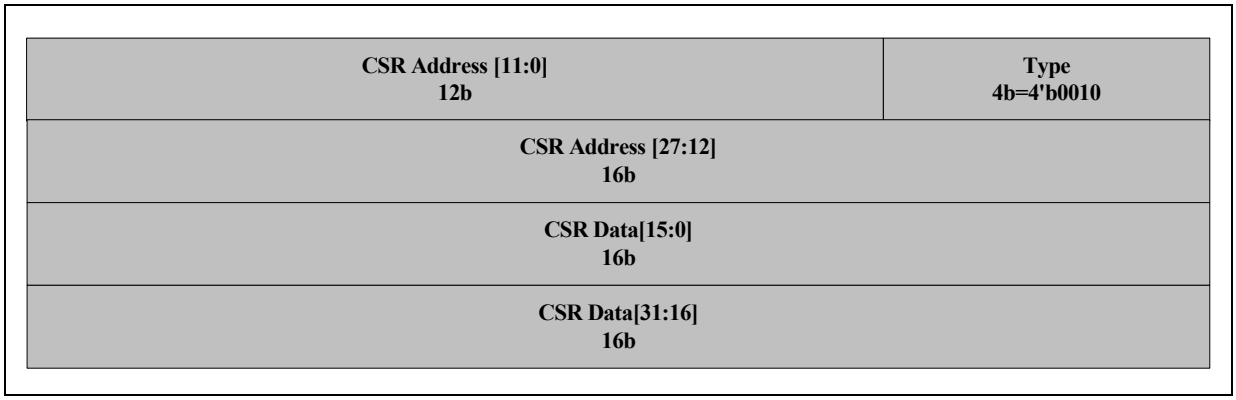


Figure 38-41.CSR Write Command



38.20 NVM General Summary Table

Table 38-123.NVM General Summary Table (Sheet 1 of 3)

Word Address	Used By	Word Name	Page
0x0000	HW	NVM Control Word 1	2505
0x0001	FW	RO Commands Version	2505
0x0002	Internal	Reserved	
0x0003	HW	Reserved_0	2505
0x0004	HW	Internal PHY PLL Configuration Module Pointer	2506
0x0005	HW	OROM Pointer	2506
0x0006	HW	RO PCIR Registers Auto-load Module Pointer	2506
0x0007	FW	Auto Generated Pointers Pointer	2506
0x0008	HW	PCIR Registers Auto-load Module Pointer	2507
0x0009	FW	EMP Global Module Pointer	2507
0x000A	HW	Reserved_1	
0x000B	FW	EMP Image Pointer	2507
0x000C	Internal	Reserved	
0x000D	HW	CSR Protected List Pointer	2507
0x000E	FW	Manageability Module Pointer	2507
0x000F	FW	EMP Settings Module Pointer	2508
0x0010	SW	SW Compatibility Word 1	2508
0x0011	SW	SW Compatibility Word 2	2508
0x0012	SW	SW Compatibility Word 3	2508
0x0013	SW	SW Compatibility Word 4	2509
0x0014	SW	SW Compatibility Word 5	2509
0x0015	SW	PBA Flags	2509
0x0016	SW	PBA Block Pointer	2509
0x0017	SW	Boot Configuration Start Address	2509
0x0018	SW	Software Reserved Word 1 - Dev Starter Version	2510
0x0019	SW	Software Reserved Word 2	2510
0x001A	SW	Software Reserved Word 3	2510
0x001B	SW	Software Reserved Word 4	2510
0x001C	SW	Software Reserved Word 5	2510
0x001D	SW	Software Reserved Word 6	2511
0x001E	SW	Software Reserved Word 7	2511
0x001F	SW	Software Reserved Word 8	2511
0x0020	SW	Software Reserved Word 9	2511
0x0021	SW	Software Reserved Word 10	2511
0x0022	SW	Software Reserved Word 11	2511
0x0023	SW	Software Reserved Word 12	2511
0x0024	SW	Software Reserved Word 13	2512
0x0025	SW	Software Reserved Word 14	2512
0x0026	SW	Software Reserved Word 15	2512
0x0027	SW	Software Reserved Word 16	2512

Table 38-123.NVM General Summary Table (Sheet 2 of 3)

Word Address	Used By	Word Name	Page
0x0028	SW	Software Reserved Word 17	
0x0029	SW	Software Reserved Word 18 - Map Version	2512
0x002A	SW	Software Reserved Word 19 - NVM_Image_Version	2512
0x002B	SW	Software Reserved Word 20 - NVM Structure Version	2513
0x002C	SW	Reserved	
0x002D	SW	Software Reserved Word 22 - EETRACK ID 1	2513
0x002E	SW	Software Reserved Word 23 - EETRACK ID 2	2513
0x002F	SW	VPD Module Pointer	2513
0x0030	SW	PXE Setup Options Pointer	2514
0x0031	SW	PXE Configuration Customization Options Pointer	2514
0x0032	SW	PXE Version	2514
0x0033	SW	IBA Capabilities	2515
0x0034	SW	Software Reserved Word 24 - Original EETRACK ID 1	2515
0x0035	SW	Software Reserved Word 25 - Original EETRACK ID 2	2515
0x0036	SW	iSCSI Option ROM Version	2516
0x0037	SW	VLAN Configuration Block Pointer	2516
0x0038	HW	POR Registers Auto-load Module Pointer	2516
0x0039	Internal	Reserved	
0x003A	HW	EMPR Registers Auto-load Pointer	2516
0x003B	HW	GLOBR Registers Auto-load Pointer	2517
0x003C	HW	CORER Registers Auto-load Pointer	2517
0x003D	HW	Reserved	
0x003E	Internal	Reserved	
0x003F	SW	Software Checksum	2517
0x0040	FW	1st Free Provisioning Area Pointer	2517
0x0041	FW	1st Free Provisioning Area Size	2517
0x0042	FW	Reserved for 4th Free Provisioning Area Pointer	2518
0x0043	FW	Reserved for 4th Free Provisioning Area Size	2518
0x0044	FW	Reserved for 3rd Free Provisioning Area Pointer	2518
0x0045	FW	Reserved for 3rd Free Provisioning Area Size	2518
0x0046	FW	2nd Free Provisioning Area Pointer	2518
0x0047	FW	2nd Free Provisioning Area Size	2519
0x0048	HW	EMP SR Settings Pointer	2519
0x0049	HW	Feature Configuration Pointer	2519
0x004A	HW	PE CORER Registers Auto-Load Pointer	2519
0x004B	FW	Reserved	
0x004C	FW	Reserved	
0x004D	HW	Configuration Metadata Pointer	2519
0x004E	HW	Immediate Values Pointer	2520
0x004F	HW	Core Mem Config Pointer	2520
0x0050	FW	FW Scratch Pad Area Pointer	2520
0x0051	FW	FW Scratch Pad Area Size	2520

Table 38-123.NVM General Summary Table (Sheet 3 of 3)

Word Address	Used By	Word Name	Page
0x0052	HW	POR CORER Registers Auto-load Pointer	2520
0x0053	Internal	Reserved	
0x0054	HW	Internal PHY 0 Configuration Module Pointer	2521
0x0055	HW	Soft SKUs	2521
0x0056	HW	Internal PHY 1 PLL Configuration Module Pointer	2521
0x0057	HW	Internal PHY 1 Configuration Module Pointer	2521
0x0058	HW	Internal PHY Configuration Override Module Pointer	2614
0x0059	HW	SW-Data-Recovery	2526
0x005A	HW	PCIR-Data-Recovery	2529
0x005B + 1*n, n=0...164	HW	Spare NVM Header Words	

38.20.1 Init Module Section Summary Table

This is the NVM header module that contains pointers to all other 1st level sections. It includes also words that are relative to the entire NVM map. See Section 38.20 for details on the inner structure.

38.20.1.1 NVM Control Word 1 (0x0000)

Bits	Field Name	NVM Image Value	Description
15:8	Reserved	0x02	Reserved.
7:6	NVM Validity	01b	The *Signature* field indicates to the device that there is a valid NVM present. If the *Signature* field is not 01b, the other bits in this word are ignored, no further NVM read is performed, and the default values are used for the configuration space IDs. 00b = NVM not present 0. 01b = NVM present. 10b = NVM not present 2. 11b = NVM not present 3.
5:0	Reserved	0x09	Reserved.

38.20.1.2 RO Commands Version (0x0001)

Bits	Field Name	NVM Image Value	Description
15:0	RO Commands Version	0xFFFF	Contains the version of the RO Commands section mapped to the last 4 KB sector of the EMP Image.

38.20.1.3 Reserved_0 (0x0003)

Bits	Field Name	NVM Image Value	Description
15:0	Reserved_0	0x0000	Reserved.

38.20.1.4 Internal PHY PLL Configuration Module Pointer (0x0004)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	0b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	Internal PHY Configuration Module Pointer	0x0	Points to the Internal PHY PLL Configuration section.

38.20.1.5 OROM Pointer (0x0005)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	1b	Pointer Type. Only the 4 KB sector unit is supported for this pointer. 0 = Word units. 1 = 4 KB sector units.
14:0	PCIe Expansion/Option ROM Pointer	0x7FFF	Points to the OROM section.

38.20.1.6 RO PCIR Registers Auto-Load Module Pointer (0x0006)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	0b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	PCIR Registers Auto-Load Module Pointer	0x7FFF	Points to the RO PCIR Registers Auto-Load Module section. For more detail on the RO PCIR Registers Auto-Load Module inner structure, see Section 38.20.2.

38.20.1.7 Auto Generated Pointers Pointer (0x0007)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	0b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	Auto Generated Pointers Module Pointer	0x7FFF	Points to the Auto Generated Pointers Module section. For more detail on the Auto Generated Pointers Module inner structure, see Section 38.20.31.

38.20.1.8 PCIR Registers Auto-Load Module Pointer (0x0008)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	0b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	PCIe TL Shared Module Pointer	0x7FFF	Points to the PCIR Registers Auto-Load Module section. For more detail on the PCIR Registers Auto-Load Module inner structure, see Section 38.20.18.

38.20.1.9 EMP Global Module Pointer (0x0009)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	1b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	EMP Global Module Pointer	0x7FFF	Points to the EMP Global Module section. For more detail on the EMP Global Module inner structure, see Section 38.20.32.

38.20.1.10 EMP Image Pointer (0x000B)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	1b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	EMP Image Pointer	0x7FFF	Points to the EMP Image section.

38.20.1.11 CSR Protected List Pointer (0x000D)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	0b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	CSR Protected List Pointer	0x7FFF	Points to the CSR Protected List section. For more detail on the CSR Protected List inner structure, see Section 38.20.3.

38.20.1.12 Manageability Module Pointer (0x000E)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	1b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	Manageability Configuration Module Pointer	0x7FFF	Points to the Manageability Module Header section. For more detail on the Manageability Module Header inner structure, see Section 38.20.33.

38.20.1.13 EMP Settings Module Pointer (0x000F)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	1b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	EMP Module Pointer	0x7FFF	Points to the EMP Settings Module Header section.

38.20.1.14 SW Compatibility Word 1 (0x0010)

Five words in the NVM image are reserved for compatibility information. New bits within these fields are defined as the need arises for determining software compatibility between various hardware revisions.

Bits	Field Name	NVM Image Value	Description
15:12	Reserved	0x0	Reserved.
11	LOM	0b	Indicates whether the NVM attached to LAN silicon contains dedicated module for option ROM. Used by option ROM update applications. 0 = NIC (Attached flash contains module for option ROM). 1 = LOM (Attached flash has no module for option ROM).
10	Server	1b	Legacy. Not currently used. 0 = Client 1 = Server
9	Reserved	0b	Reserved.
8	OEM/Retail	0b	Legacy. Not currently used. 0 = Retail 1 = OEM
7:0	Reserved	0x00	Reserved.

38.20.1.15 SW Compatibility Word 2 (0x0011)

Five words in the NVM image are reserved for compatibility information. New bits within these fields are defined as the need arises for determining software compatibility between various hardware revisions.

Bits	Field Name	NVM Image Value	Description
15:0	Reserved	0xFFFF	Reserved.

38.20.1.16 SW Compatibility Word 3 (0x0012)

Five words in the NVM image are reserved for compatibility information. New bits within these fields are defined as the need arises for determining software compatibility between various hardware revisions.

Bits	Field Name	NVM Image Value	Description
15:0	Reserved	0xFFFF	Reserved.

38.20.1.17 SW Compatibility Word 4 (0x0013)

Five words in the NVM image are reserved for compatibility information. New bits within these fields are defined as the need arises for determining software compatibility between various hardware revisions.

Bits	Field Name	NVM Image Value	Description
15:0	Reserved	0xFFFF	Reserved.

38.20.1.18 SW Compatibility Word 5 (0x0014)

Five words in the NVM image are reserved for compatibility information. New bits within these fields are defined as the need arises for determining software compatibility between various hardware revisions.

Bits	Field Name	NVM Image Value	Description
15:0	Reserved	0xFFFF	Reserved.

38.20.1.19 PBA Flags (0x0015)

Bits	Field Name	NVM Image Value	Description
15:0	PBA Flags	0xFAFA	A flag value of 0xFAFA indicates that the PBA is stored in a separate PBA block.

38.20.1.20 PBA Block Pointer (0x0016)

Bits	Field Name	NVM Image Value	Description
15:0	PBA Block Pointer		

38.20.1.21 Boot Configuration Start Address (0x0017)

Address of iSCSI Boot configuration module. This is a word pointer. The block length is embedded in the module.

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	0b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	iSCSI Boot Configuration Start Address	0x7FFF	This module is 1504 bytes long and must be mapped in the first valid 4 KB sector of the Flash. Points to the Boot Configuration Block section. For more detail on the Boot Configuration Block inner structure, see Section 38.20.12.

38.20.1.22 Software Reserved Word 1 - Dev Starter Version (0x0018)

Dev_Starter map version used to produce this image. This word must be filled manually.

Bits	Field Name	NVM Image Value	Description
15:12	Major		NVM major version.
11:8	Decimal Point	0x0	Decimal Point. Used by automatic NVM reading tools. Must be always set to 0x0.
7:0	Minor		NVM minor version.

38.20.1.23 Software Reserved Word 2 (0x0019)

Bits	Field Name	NVM Image Value	Description
15:4	Reserved	0xFFF	Reserved.
3	WoL Control Port 3	0b	Wake On LAN feature for port 3. 0 = Enabled — Supported and enabled 1 = Disabled — Disabled or not supported
2	WoL Control Port 2	0b	Wake On LAN feature for port 2. 0 = Enabled — Supported and enabled 1 = Disabled — Disabled or not supported
1	WoL Control Port 1	0b	Wake On LAN feature for port 1. 0 = Enabled — Supported and enabled 1 = Disabled — Disabled or not supported
0	WoL Control Port 0	0b	Wake On LAN feature for port 0. 0 = Enabled — Supported and enabled 1 = Disabled — Disabled or not supported

38.20.1.24 Software Reserved Word 3 (0x001A)

Bits	Field Name	NVM Image Value	Description
15:0	Reserved	0xFFFF	Reserved.

38.20.1.25 Software Reserved Word 4 (0x001B)

Bits	Field Name	NVM Image Value	Description
15:0	Reserved	0xFFFF	Reserved.

38.20.1.26 Software Reserved Word 5 (0x001C)

Bits	Field Name	NVM Image Value	Description
15:0	Reserved	0xFFFF	Reserved.

38.20.1.27 Software Reserved Word 6 (0x001D)

Bits	Field Name	NVM Image Value	Description
15:0	Reserved	0xFFFF	Reserved.

38.20.1.28 Software Reserved Word 7 (0x001E)

Bits	Field Name	NVM Image Value	Description
15:0	Reserved	0xFFFF	Reserved.

38.20.1.29 Software Reserved Word 8 (0x001F)

Bits	Field Name	NVM Image Value	Description
15:0	Reserved	0xFFFF	Reserved.

38.20.1.30 Software Reserved Word 9 (0x0020)

Bits	Field Name	NVM Image Value	Description
15:0	Reserved	0xFFFF	Reserved.

38.20.1.31 Software Reserved Word 10 (0x0021)

Bits	Field Name	NVM Image Value	Description
15:0	Reserved	0xFFFF	Reserved.

38.20.1.32 Software Reserved Word 11 (0x0022)

Bits	Field Name	NVM Image Value	Description
15:0	Reserved	0xFFFF	Reserved.

38.20.1.33 Software Reserved Word 12 (0x0023)

Bits	Field Name	NVM Image Value	Description
15:0	Reserved	0xFFFF	Reserved.

38.20.1.34 Software Reserved Word 13 (0x0024)

Bits	Field Name	NVM Image Value	Description
15:0	Reserved	0xFFFF	Reserved.

38.20.1.35 Software Reserved Word 14 (0x0025)

Bits	Field Name	NVM Image Value	Description
15:0	Reserved	0xFFFF	Reserved.

38.20.1.36 Software Reserved Word 15 (0x0026)

Bits	Field Name	NVM Image Value	Description
15:0	Reserved	0xFFFF	Reserved.

38.20.1.37 Software Reserved Word 16 (0x0027)

Bits	Field Name	NVM Image Value	Description
15:0	Reserved	0xFFFF	Reserved.

38.20.1.38 Software Reserved Word 18 - Map Version (0x0029)

Automatically generated by the NVM update tool.

Bits	Field Name	NVM Image Value	Description
15:0	Map Version		

38.20.1.39 Software Reserved Word 19 - NVM_Image_Version (0x002A)

Automatically generated by the NVM update tool.

Bits	Field Name	NVM Image Value	Description
15:0	NVM Image Version		

38.20.1.40 Software Reserved Word 20 - NVM Structure Version (0x002B)

Bits	Field Name	NVM Image Value	Description
15:12	Major	0x1	NVM major version.
11:8	Decimal Point	0x0	Decimal Point. Used by automatic NVM reading tools. Must be always set to 0x0.
7:0	Minor	0x3	NVM structure minor version.

38.20.1.41 Software Reserved Word 22 - EETRACK ID 1 (0x002D)

This word is for the first word of the eTrack_ID number written by EEPROM Manager Tool.

Bits	Field Name	NVM Image Value	Description
15:0	eTrack_ID Word 1	0xFFFF	EEPROM Manager Tool writes a unique 32-bit eTrack_ID number in two sequential NVM words. The eTrack_ID is written when EEPROM Manager Tool creates an image on the Intel network. The eTrack_ID DB tracks NVM images back to a specific SCM build.

38.20.1.42 Software Reserved Word 23 - EETRACK ID 2 (0x002E)

This word is for the second word of the eTrack_ID number written by EEPROM Manager Tool.

Bits	Field Name	NVM Image Value	Description
15:0	eTrack_ID Word 2		

38.20.1.43 VPD Module Pointer (0x002F)

Word pointer to Vital Product Data module. Block length is embedded in the module.

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	0b	Pointer Type:. 0 = Word units. 1 = 4 KB sector units.
14:0	VPD Pointer	0x7FFF	Vital Product Data Pointer. Points to the VPD Module section. For more detail on the VPD Module inner structure, see Section 38.20.6. 0x7FFF is the default unless VPD relative section is specified. The VPD section size is usually 64 words and is initialized to 0 or 0x7FFF. During run time this module is accessible through the VPD capability in the PCI configuration space. This module must be mapped in the first valid 4KB sector of the Flash.

38.20.1.44 PXE Setup Options Pointer (0x0030)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	0b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	PXE Setup Options Pointer	0x7FFF	Points to the PXE Setup Options section. For more detail on the PXE Setup Options inner structure, see Section 38.20.10.

38.20.1.45 PXE Configuration Customization Options Pointer (0x0031)

Word 0x31 of the NVM contains settings that can be programmed by an OEM or network administrator to customize the operation of the software. These settings cannot be changed from within the Control- S setup menu. The lower byte contains settings that would typically be configured by a network administrator using an external utility; these settings generally control which setup menu options are changeable. The upper byte is generally settings that would be used by an OEM to control the operation of the agent in a LOM environment, although there is nothing in the agent to prevent their use on a NIC implementation. The default value for this word is 0x4000.

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	0b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	PXE Configuration Customization Options Pointer	0x7FFF	Points to the PXE Configuration Customizations Options section. For more detail on the PXE Configuration Customizations Options inner structure, see Section 38.20.11.

38.20.1.46 PXE Version (0x0032)

Word 0x32 of the NVM is used to store the version of the boot agent that is stored in the Flash image. When the Boot Agent loads, it can check this value to determine if any first-time configuration needs to be performed. The agent then updates this word with its version. Some diagnostic tools to report the version of the Boot Agent in the Flash also read this word.

Bits	Field Name	NVM Image Value	Description
15:12	Major Version	0x0	PXE Boot Agent Major Version. Default value is 0.
11:8	Minor Version	0x0	PXE Boot Agent Minor Version. Default value is 0.
7:0	Build Number	0x00	PXE Boot Agent Build Number. Default value is 0.

38.20.1.47 IBA Capabilities (0x0033)

Word 0x33 of the NVM is used to enumerate the boot technologies that have been programmed into the Flash. This is updated by Flash configuration tools and is not updated or read by IBA.

Bits	Field Name	NVM Image Value	Description
15:14	Signature	01b	Signature. Must be set to 01b to indicate that this word has been programmed by the agent or other configuration software.
13:6	Reserved	0x00	Reserved. Must be 0.
5	Reserved	1b	Reserved
4	iSCSI boot	0b	iSCSI is present if set to 1b. 0 = Not Present 1 = Present
3	efi ebd driver	1b	EFI UNDI driver is present if set to 1b. 0 = Not Present 1 = Present
2	RPL	0b	RPL module is present if set to 1b. Reserved bit for devices. 0 = Not Present 1 = Present
1	PXE/UNDI Driver	1b	PXE UNDI driver is present if set to 1b. 0 = Not Present 1 = Present
0	PXE Base Code	1b	PXE Base Code is present if set to 1b. 0 = Not Present 1 = Present

38.20.1.48 Software Reserved Word 24 - Original EETRACK ID 1 (0x0034)

Bits	Field Name	NVM Image Value	Description
15:0	Original EETRACK ID 1	0x0000	

38.20.1.49 Software Reserved Word 25 - Original EETRACK ID 2 (0x0035)

Bits	Field Name	NVM Image Value	Description
15:0	Original EETRACK ID 2	0x0000	

38.20.1.50 iSCSI Option ROM Version (0x0036)

Word 0x36 of the NVM is used to store the version of iSCSI Option ROM updated. The value must be above 0x2000 and the value below (word 0x1FFF = 16 KB NVM size) is reserved for future expansion for a pointer to combo option ROM component version structure. iSCSIUtl, FLAUtil, DMiX update iSCSI Option ROM version if the value is above 0x2000, 0x0000, or 0xFFFF. The pointer (0x0040 - 0x1FFF) should be kept and not be overwritten.

Bits	Field Name	NVM Image Value	Description
15:0	Reserved	0xFFFF	Reserved.

38.20.1.51 VLAN Configuration Block Pointer (0x0037)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	0b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	VLAN Configuration Block Pointer	0x7FFF	Points to the VLAN Configuration Block section. For more detail on the VLAN Configuration Block inner structure, see Section 38.20.13.

38.20.1.52 POR Registers Auto-Load Module Pointer (0x0038)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	0b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	POR Registers Auto-Load Module Pointer	0x7FFF	Points to the POR Registers Auto-Load Module section. For more detail on the POR Registers Auto-Load Module inner structure, see Section 38.20.18.

38.20.1.53 EMPR Registers Auto-Load Pointer (0x003A)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	0b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	EMPR Registers Auto-Load Module Pointer	0x7FFF	Points to the EMPR Auto-Load section. For more detail on the EMPR Auto-Load inner structure, see Section 38.20.4.

38.20.1.54 GLOBR Registers Auto-Load Pointer (0x003B)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	0b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	GLOBR Registers Auto-Load Module Pointer	0x7FFF	Points to the GLOBR Registers Auto-Load Module section. For more detail on the GLOBR Registers Auto-Load Module inner structure, see Section 38.20.18.

38.20.1.55 CORER Registers Auto-Load Pointer (0x003C)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	0b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	CORER Registers Auto-Load Module Pointer	0x7FFF	Points to the CORER Registers Auto-Load Module section. For more detail on the CORER Registers Auto-Load Module inner structure, see Section 38.20.18.

38.20.1.56 Software Checksum (0x003F)

Bits	Field Name	NVM Image Value	Description
15:0	Checksum		The SW Checksum field covers the whole 64 KB shadow RAM contents (reserved words included), excepted to the VPD area and to the PCIe ALT Auto-Load module which are skipped. Its value is computed such that after adding all the covered words, including the SW Checksum word itself, the sum is 0xBABA.The checksum word is used to ensure that the base NVM image is a valid image. The initial value in the 16-bit summing register should be 0x0000 and the carry bit should be ignored after each addition. This word is used strictly by the SW. The HW does not calculate nor check its content but rather checks the NVM validity field in the NVM Control Word 1.

38.20.1.57 1st Free Provisioning Area Pointer (0x0040)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	1b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	1st Free Provisioning Area Pointer	0x7FFF	Points to the 1st Free Provisioning Area section.

38.20.1.58 1st Free Provisioning Area Size (0x0041)

Bits	Field Name	NVM Image Value	Description
15:10	Reserved	0x00	Reserved.

Bits	Field Name	NVM Image Value	Description
9:0	1st Free Provisioning Area Size	0x138	Size expressed in 4 KB sector units.

38.20.1.59 Reserved for 4th Free Provisioning Area Pointer (0x0042)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	1b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	2nd Free Provisioning Area Pointer	0x7FFF	Points to the 4th Free Provisioning Area section.

38.20.1.60 Reserved for 4th Free Provisioning Area Size (0x0043)

Bits	Field Name	NVM Image Value	Description
15:10	Reserved	0x00	Reserved.
9:0	2nd Free Provisioning Area Size	0x010	Size expressed in 4 KB sector units

38.20.1.61 Reserved for 3rd Free Provisioning Area Pointer (0x0044)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	1b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	3rd Free Provisioning Area Pointer	0x7FFF	Points to the 3rd Free Provisioning Area section.

38.20.1.62 Reserved for 3rd Free Provisioning Area Size (0x0045)

Bits	Field Name	NVM Image Value	Description
15:10	Reserved	0x00	Reserved.
9:0	3rd Free Provisioning Area Size	0x020	Size expressed in 4 KB sector units

38.20.1.63 2nd Free Provisioning Area Pointer (0x0046)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	1b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	2nd Free Provisioning Area Pointer	0x7FFF	Points to the 2nd Free Provisioning Area section.

38.20.1.64 2nd Free Provisioning Area Size (0x0047)

Bits	Field Name	NVM Image Value	Description
15:10	Reserved	0x00	Reserved.
9:0	2nd Free Provisioning Area Size	0x002	Size expressed in 4 KB sector units

38.20.1.65 EMP SR Settings Pointer (0x0048)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	0b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	EMP SR Settings Pointer	0x7FFF	Points to the EMP SR Settings Module Header section. For more detail on the EMP SR Settings Module Header inner structure, see Section 38.20.19.

38.20.1.66 Feature Configuration Pointer (0x0049)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	0b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	Feature Configuration Pointer	0x7FFF	Points to the Feature Configuration section. For more detail on the Feature Configuration inner structure, see Section 38.20.19.

38.20.1.67 PE CORER Registers Auto-load Pointer (0x004A)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	0b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	PE CORER Config Pointer	0x7FFF	Points to the PE CORER Registers section. For more detail on the PE CORER Registers inner structure, see Section 38.20.18.

38.20.1.68 Configuration Metadata Pointer (0x004D)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	1b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	Configuration Metadata Pointer	0x7FFF	Points to the Configuration Metadata section. For more detail on the Configuration Metadata inner structure, see Section 38.20.18.

38.20.1.69 Immediate Values Pointer (0x004E)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	0b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	Immediate Values Pointer	0x0	Points to the Immediate Values section. For more detail on the Immediate Values inner structure, see Section 38.20.26

38.20.1.70 Core Mem Config Pointer (0x004F)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	0b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	Core Mem Config Pointer	0x7FFF	Points to the Core Mem Config Section. For more detail on the Core Mem Config inner structure, see Section 38.20.5.

38.20.1.71 FW Scratch Pad Area Pointer (0x0050)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	1b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	FW Scratch Pad Area Pointer	0x7FFF	Points to the FW Scratch Pad Area Section. For more detail on the FW Scratch Pad Area inner structure, see Section 38.20.

38.20.1.72 FW Scratch Pad Area Size (0x0051)

Bits	Field Name	NVM Image Value	Description
15:10	Reserved	0x00	Reserved.
9:0	FW Scratch Pad Area Size	0x040	Size expressed in 4 KB sector units.

38.20.1.73 POR CORER Registers Auto-Load Pointer (0x0052)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	1b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	POR CORER Config Pointer	0x7FFF	Points to the POR CORER Registers Section. For more detail on the POR CORER Registers inner structure, see Section 38.20.18.

38.20.1.74 Internal PHY 0 Configuration Module Pointer (0x0054)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	1b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	Internal PHY Configuration Pointer	0x0	Points to the Internal PHY 0 Configuration 0 Section. For more detail on the Internal PHY 0 Configuration 0 inner structure, see Section 38.20.28.

38.20.1.75 Soft SKUs(0x0055)

Bits	Field Name	NVM Image Value	Description
15	Reserved	0x0	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	Soft SKUs	0x4	Number of ports enable to operate at 10 GbE. Valid values are 0, 2, and 4.

38.20.1.76 Internal PHY 1 PLL Configuration Module Pointer (0x0056)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	1b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	Internal PHY Configuration Pointer	0x0	Points to the Internal PHY 1 PLL Configuration 0 Section. For more detail on the Internal PHY 1 PLL Configuration 0 inner structure, see Section 38.20.42.

38.20.1.77 Internal PHY 1 Configuration Module Pointer (0x0057)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	1b	Pointer Type. 0 = Word units. 1 = 4 KB sector units.
14:0	Internal PHY Configuration Pointer	0x0	Points to the Internal PHY 1 Configuration 0 Section. For more detail on the Internal PHY 1 Configuration 0 inner structure, see Section 38.20.28.

38.20.1.78 Internal PHY Configuration Override Module Pointer (0x0058)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	0x0	Pointer Type. 0b = Word units. 1b = 4 KB sector units.
14:0	Internal PHY Configuration Module Pointer	0x0	Points to the Internal PHY Configuration Module Pointer section. For more detail on the Internal PHY Configuration Override inner structure, see Section 38.20.28

38.20.1.79 SW-Data-Recovery (0x0059)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	0x0	Pointer Type. 0b = Word units. 1b = 4 KB sector units.
14:0	SW-Data-Recovery Pointer	0x7FFF	Points to the SW-Data-Recovery section. For more detail on the SW-Data-Recovery inner structure, see Section 38.20.7

38.20.1.80 PCIR-Data-Recovery (0x005A)

Bits	Field Name	NVM Image Value	Description
15	Pointer Type	0x0	Pointer Type. 0b = Word units. 1b = 4 KB sector units.
14:0	PCIR-Data-Recovery Pointer	0x7FFF	Points to the PCIR-Data-Recovery Pointer section. For more detail on the PCIR-Data-Recovery Pointer inner structure, see Section 38.20.8

38.20.1.81 Spare NVM Header Words[n] (0x005B + 1*n, n=0...164)

Bits	Field Name	NVM Image Value	Description
15:0	RESERVED	0x7FFF	

38.20.2 RO PCIR Registers Auto-load Module Section Summary Table

Contains read-only parameters that configure the PCIe transaction layer.

Word Offset	Description	Page
0x0000	Module Length	2522
0x0001	LO PCIR Data	2523

38.20.2.1 Module Length (0x0000)

Bits	Field Name	NVM Image Value	Description
15:0	Module Length		

38.20.2.2 LO PCIR Data (0x0001)

Raw data module length: variable.

This word must be disabled at the image level.

38.20.3 CSR Protected List Section Summary Table

Defines the list of the protected CSRs and their default settings. These registers are made RO to the host as they contains settings which are critical for the host to flash access when in blank flash programming mode.

Word Offset	Description	Page
0x0000	Module Length	2523
0x0001	Reserved (must be disabled at the image level)	
0x0002 - 0x0005	NVM contents for GLGEN_STAT	2523
0x0006 - 0x0009	NVM contents for GLNVM_ALTIMERS	2524
0x000A - 0x000D	NVM contents for GLGEN_PE_ENA	2524
0x001D - 0x0020	NVM contents for GLPCI_LBARCTRL	2524
0x0021 - 0x0028	Reserved	
0x0029 - 0x00A3	NVM contents for GLNVM_PROTCSR	2524

38.20.3.1 Module Length (0x0000)

Bits	Field Name	NVM Image Value	Description
15:0	Module Length		Length in: 2-bytes unit - 1. First Section -> Word: CSR Protected List -> Module Length Last Section -> Word: CSR Protected List -> Starting Address Low at GLNVM_PROTCSR[0]

38.20.3.2 GLGEN_STAT (0x0002 - 0x0005)

38.20.3.2.1 Address Low at GLGEN_STAT (0x0002)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLGEN_STAT	0x12C	
3:0	Type	0x1	

38.20.3.2.2 Address High at GLGEN_STAT (0x0003)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLGEN_STAT	0x00B6	

38.20.3.2.3 Data Low of GLGEN_STAT (0x0004)

38.20.3.2.4 Data High of GLGEN_STAT (0x0005)

38.20.3.3 GLNVM_ALTIMERS (0x0006 - 0x0009)

38.20.3.3.1 Address Low at GLNVM_ALTIMERS (0x0006)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLNVM_ALTIMERS	0x140	
3:0	Type	0x1	

38.20.3.3.2 Address High at GLNVM_ALTIMERS - 0x0007

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLNVM_ALTIMERS		

38.20.3.3.3 Data Low of GLNVM_ALTIMERS (0x0008)

38.20.3.3.4 Data High of GLNVM_ALTIMERS (0x009)

38.20.3.4 GLPCI_LBARCTRL (0x001D - 0x0020)

38.20.3.4.1 Address Low at GLPCI_LBARCTRL (0x001D)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLPCI_LBARCTRL	0xBE484	
3:0	Type	0x1	

38.20.3.4.2 Address High at GLPCI_LBARCTRL (0x001E)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLPCI_LBARCTRL		

38.20.3.4.3 Data Low of GLPCI_LBARCTRL (0x001F)

38.20.3.4.4 Data High of GLPCI_LBARCTRL (0x0020)

38.20.3.5 GLNVM_PROTCSR (0x0029 - 0x00A3)

38.20.3.5.1 Starting Address Low at GLNVM_PROTCSR[n] (0x0029)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLNVM_PROTCSR	0xB6010	
3:0	Type	0x2	

38.20.3.5.2 Starting Address High at GLNVM_PROTCSR[n] (0x002A)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLNVM_PROTCSR		

38.20.3.5.3 Attributes at GLNVM_PROTCSR[n] (0x002B)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x3C	
4:3	Skip	00b	
2:0	Width	000b	

38.20.3.5.4 Data Low of GLNVM_PROTCSR[n] (0x002C + 2*n, n=0...59)

38.20.3.5.5 Data High of GLNVM_PROTCSR[n] (0x002D + 2*n, n=0...59)

38.20.4 EMPR Auto-Load Section Summary Table

Read margin settings to the device's core memories which are loaded by HW auto-load only on the first EMPR event that occurs after POR.

The contents of this module is checked against the CSR Protected List.

Word Offset	Description	Page
0x0000	Module Length	2525
0x0001	EMPR Autoload Data	2525

38.20.4.1 Module Length (0x0000)

Bits	Field Name	NVM Image Value	Description
15:0	Module Length		Length in: 2-bytes unit - 1. First Section -> Word: EMPR Auto-Load -> Module Length Last Section -> Word: EMPR Auto-Load -> EMPR Auto-Load Data

38.20.4.2 EMPR Autoload Data (0x0001)

Raw data module length: variable

Contains patches

38.20.5 Core Mem Config Section Summary Table

Read margin settings to the device’s core memories, which are loaded by hardware auto-load on every CORER event. The contents of this module is checked against the CSR Protected List.

Word Offset	Description	Page
0x0000	Module Length	2526
0x0001	Core Mem Config Data	2526

38.20.5.1 Module Length (0x0000)

Bits	Field Name	NVM Image Value	Description
15:0	Module Length		Length in: 2 bytes unit - 1. First section -> Word: Core Mem Config -> Module Length Last section -> Word: Core Mem Config -> Core Mem Config Data

38.20.5.2 Core Mem Config Data (0x0001)

Raw data module length: variable

38.20.6 VPD Module Section Summary Table

VDP loaded by OEM. It contains RO and RW info about the NIC/LOM.

Word Offset	Description	Page
0x0000	[New Word]	See 38.20.6.1

38.20.6.1 [New Word] (0x0000)

Raw data module length: 512 words

38.20.7 SW-Data-Recovery Section Summary Table

Word Offset	Description	Reference
0x0000	Section Length	See section 38.20.7.1
0x0001	Section Header	See section 38.20.7.2
0x0002	Original EE-Track-ID Word 1	See section 38.20.7.3
0x0003	Original EE-Track-ID Word 2	See section 38.20.7.4
0x0004 + 1*n, n=0...4	PBA	See section 38.20.7.5
0x0009	MAC Addresses Length	See section 38.20.7.6
0x000A + 3*n, n=0...3	MAC Addresses Word 0	See section 38.20.7.7
0x000B + 3*n, n=0...3	MAC Addresses Word 1	See section 38.20.7.8
0x000C + 3*n, n=0...3	MAC Addresses Word 2	See section 38.20.7.9

38.20.7.1 Section Length (0x0000)

Bits	Field Name	NVM Image Value	Description
15:0	Section Length		Length in: 2 Bytes unit - 1 First Section -> Word: SW-Data-Recovery Section -> Section Length Last Section -> Word: SW-Data-Recovery Section -> MAC Addresses word 2

38.20.7.2 Section Header (0x0001)

Bits	Field Name	NVM Image Value	Description
15	Recovery Mode	0x0	
14:4	Reserved	0x0	Define the current status of NVM-Recovery-Mode. Set by firmware when entering recovery mode to keep the status POR-persistent. Firmware must clear this bit after an NVM update is done by software.
3	MAC Addresses Valid	0x0	
2	PBA Valid	0x0	
1	EETRACK-ID Valid	0x0	
0	PCIR Section Valid	0x0	Define the validity of the PCIR-Data-Recovery section. 0b = Data wasn't programmed yet by firmware. 1b = Data programmed and valid.

38.20.7.3 Original EE-Track-ID word 1 (0x0002)

The original EE-Track-ID of the device.

Taken from words 0x34-0x35 in NVM.

If this field is empty (all 0xFF's), firmware must copy the current ETID from words 0x2D-0x2E to words 0x34-0x35 and use this field.

Bits	Field Name	NVM Image Value	Description
15:0	ETID	0x0000	

38.20.7.4 Original EE-Track-ID word 2 (0x0003)

The original EE-Track-ID of the device.

Taken from words 0x34-0x35 in NVM.

If this field is empty (all 0xFF's), firmware must copy the current ETID from words 0x2D-0x2E to words 0x34-0x35 and use this field.

Bits	Field Name	NVM Image Value	Description
15:0	ETID	0x0000	

38.20.7.5 PBA[n] (0x0004 + 1*n, n=0...4)

PBA section.

Bits	Field Name	NVM Image Value	Description
15:0	PBA	0x0000	See respective bits in the PRTPM_SAH register.

38.20.7.6 MAC Addresses Length (0x0009)

Length of the MAC addresses field [words].

Bits	Field Name	NVM Image Value	Description
15:0	MAC Addresses Length	0x004	

38.20.7.7 MAC Addresses Word 0[n] (0x000A + 3*n, n=0...3)

The PF MAC address table in the NVM that is loaded by firmware to PRTPM_SAL/H. The full table is kept to revert all MAC addresses in SFP mode.

Bits	Field Name	NVM Image Value	Description
15:0	MAC Addresses Word 0	0x0	

38.20.7.8 MAC Addresses Word 1[n] (0x000B + 3*n, n=0...3)

The PF MAC address table in the NVM that is loaded by firmware to PRTPM_SAL/H. The full table is kept to revert all MAC addresses in SFP mode.

Bits	Field Name	NVM Image Value	Description
15:0	MAC Addresses Word 1	0x0	

38.20.7.9 MAC Addresses word 2[n] (0x000C + 3*n, n=0...3)

The PF MAC address table in the NVM that is loaded by firmware to PRTPM_SAL/H. The full table is kept to revert all MAC addresses in SFP mode.

Bits	Field Name	NVM Image Value	Description
15:0	MAC Addresses Word 2	0x0	

38.20.8 PCIR-Data-Recovery Section Summary Table

Word Offset	Description	Reference
0x0000	Section Length	See section 38.20.8.1
0x0001 + 1*n, n=0...23	PCIR Word	See section 38.20.8.2

38.20.8.1 Section Length (0x0000)

Bits	Field Name	NVM Image Value	Description
15:0	Section Length		Length in: 2 Bytes unit - 1 First Section -> Word: PCIR-Data-Recovery Section -> Section Length Last Section -> Word: PCIR-Data-Recovery Section -> PCIR Word

38.20.8.2 PCIR Word[n] (0x0001 + 1*n, n=0...23)

Bits	Field Name	NVM Image Value	Description
15:0	PCIR Field	0x0	

38.20.9 PBA Block Section Summary Table

The PBA block contains the complete PBA number including the dash and the first digit of the 3-digit suffix.

Word Offset	Description	Page
0x0000	PBA Section Length	2529
0x0001	Word1	2529
0x0002	Word2	2530
0x0003	Word3	2530
0x0004	Word4	2530
0x0005	Word5	2530

38.20.9.1 PBA Section Length (0x0000)

Bits	Field Name	NVM Image Value	Description
15:0	PBA Section Length Field	0x0006	Length in words of the PBA Block.

38.20.9.2 Word1 (0x0001)

Bits	Field Name	NVM Image Value	Description
15:0	Word1 Field		PBA Number stored in hexadecimal ASCII values.

38.20.9.3 Word2 (0x0002)

Bits	Field Name	NVM Image Value	Description
15:0	Word2 Field		PBA Number stored in hexadecimal ASCII values.

38.20.9.4 Word3 (0x0003)

Bits	Field Name	NVM Image Value	Description
15:0	Word3 Field		PBA Number stored in hexadecimal ASCII values.

38.20.9.5 Word4 (0x0004)

Bits	Field Name	NVM Image Value	Description
15:0	Word4 Field		PBA Number stored in hexadecimal ASCII values.

38.20.9.6 Word5 (0x0005)

Bits	Field Name	NVM Image Value	Description
15:0	Word5 Field		PBA Number stored in hexadecimal ASCII values.

38.20.10 PXE Setup Options Section Summary Table

Setup options for the PXE driver. It is defined per PCIe function.

Word Offset	Description	Page
0x0000	Section Length	2532
0x0001 + 1*n, n=0...15	Setup Options PCI Function	2530

38.20.10.1 Section Length (0x0000)

The length of the section in words. Note that section length does not include a count for the section length word.

Bits	Field Name	NVM Image Value	Description
15:0	Section Length		

38.20.10.2 Setup Options PCI Function[n] (0x0001 + 1*n, n=0...15)

The main setup options for PF n are stored in this word. These options are those that can be changed by the user with the Control-S setup menu.

Bits	Field Name	NVM Image Value	Description
15:13	Reserved	000b	Reserved. Must be 000b.

Bits	Field Name	NVM Image Value	Description
12:10	FSD	000b	Control forcing speed and duplex during driver operation. 000b = Auto-negotiate 001b = Reserved 010b = Reserved 011b = Not valid (treated as 000b) 100b = Reserved 101b = Reserved 110b = Reserved 111b = 1000 Mb/s full duplex Only applicable for copper-based adapters. Not applicable to 10 GbE. Default value is 000b.
9	Reserved	0b	Reserved
8	DSM	1b	Display Setup Message. If the bit is set to 1b, the "Press Control-S" message displays after the title message. Default value is 1b.
7:6	PT	00b	Prompt Time. These bits control how long the "Press Control-S" setup prompt message is displayed, if enabled by DIM. 00b = 2 seconds (default) 01b = 3 seconds 10b = 5 seconds 11b = 0 seconds ***Note:*** The "Press Ctrl-S" message is not displayed if 0 seconds prompt time is selected.
5	iSCSI Boot disabled	0b	When this bit is set and adapter port is neither iSCSI primary nor secondary, Setup code must not be loaded. Otherwise iSCSI banner and Setup menu should be accessible as in current design. 0 = Enabled 1 = Disabled This bit must be changed at factory level and not be altered by any end-customer tools. ***Note:*** For regular NICs and LOM design this bit should be always cleared in NVM image. Otherwise, iSCSI setup is accessible for the user.
4:3	DBS	00b	Default Boot Selection. These bits select which device is the default boot device. They are only used if the agent detects that the BIOS does not support boot order selection, or if the *MODE* field of word 31h is set to MODE_LEGACY. 00b = Network boot, then local boot (default). 01b = Local boot, then network boot. 10b = Network boot only. 11b = Local boot only.
2:0	PS	000b	Protocol Select. These bits select the active boot protocol. 000b = PXE (default value). 001b = RPL (only if RPL is in the Flash). 010b = iSCSI Boot primary port (only if iSCSI Boot is using this adapter). 011b = iSCSI Boot secondary port (only if iSCSI Boot is using this adapter). 100b = Reserved All other values are reserved. Only the default value of 00b should be initially programmed into the adapter. Other values should only be set by configuration utilities.

38.20.11 PXE Configuration Customizations Options Section Summary Table

Configuration customization options for the PXE driver. It is defined per PCIe function.

Word Offset	Description	Page
0x0000	Section Length	2532
0x0001 + 1*n, n=0...15	Configuration Customization Options PCI Function	2532

38.20.11.1 Section Length (0x0000)

The length of the section in words. Note that section length does not include a count for the section length word.

Bits	Field Name	NVM Image Value	Description
15:0	Section Length		

38.20.11.2 Configuration Customization Options PCI Function[n] (0x0001 + 1*n, n=0...15)

Contains settings that can be programmed by an OEM or network administrator to customize the operation of the software. These settings cannot be changed from within the Control- S setup menu. The lower byte contains settings that would typically be configured by a network administrator using an external utility; these settings generally control which setup menu options are changeable. The upper byte is generally settings that would be used by an OEM to control the operation of the agent in a LOM environment, although there is nothing in the agent to prevent their use on a NIC implementation. The default value for this word is 0x4000.

Bits	Field Name	NVM Image Value	Description
15:14	Signature	01b	Signature. Must be set to 01b to indicate that this word has been programmed by the agent or other configuration software.
13:12	Reserved	00b	Reserved. Must be 0.
11	Continuous Retry	0b	Selects Continuous Retry operation. If this bit is set, IBA does NOT transfer control back to the BIOS if it fails to boot due to a network error (such as failure to receive DHCP replies). Instead, it restarts the PXE boot process again. 0 = Disable 1 = Enable If this bit is set, the only way to cancel PXE boot is for the user to press **ESC** on the keyboard. Retry is not attempted due to hardware conditions such as an invalid NVM checksum or failing to establish link. Default value is 0b.

Bits	Field Name	NVM Image Value	Description
10:8	Operating mode	000b	Selects the agent's boot order setup mode. This field changes the agent default behavior to make it compatible with systems that do not completely support the BBS and PnP Expansion ROM standards. 000b = Normal mode — The agent attempts to detect BBS and PnP Expansion ROM support as it normally does. 001b = Force Legacy mode. The agent does not attempt to detect BBS or PnP Expansion ROM supports in the BIOS and assumes the BIOS is not compliant. The user can change the BIOS boot order in the Setup Menu. 010b = Force BBS mode. The agent assumes the BIOS is BBS compliant, even though it might not be detected as such by the agent's detection code. The user CANNOT change the BIOS boot order in the Setup Menu. 011b = Force PnP Int18 mode. The agent assumes the BIOS allows boot order setup for PnP Expansion ROMs and hooks interrupt 0x18 (to inform the BIOS that the agent is a bootable device) in addition to registering as a BBS IPL device. The user CANNOT change the BIOS boot order in the Setup Menu. 100b = Force PnP Int19 mode. The agent assumes the BIOS allows boot order setup for PnP Expansion ROMs and hook interrupt 0x19 (to inform the BIOS that the agent is a bootable device) in addition to registering as a BBS IPL device. The user CANNOT change the BIOS boot order in the Setup Menu. 101b = Reserved for future use. If specified, is treated as a value of 000b. 110b = Reserved for future use. If specified, is treated as a value of 000b. 111b = Reserved for future use. If specified, is treated as a value of 000b.
7:6	Reserved	00b	Reserved. Must be 0.
5	Disable Flash Update	0b	Disable Flash Update. If this bit is set to 1b, the user is not allowed to update the flash image using PROSet. Default value is 0b. 0 = Enable flash update. 1 = Disable flash update.
4	Disable Legacy OS Wakeup Menu	0b	Disable Legacy Wakeup Support. If this bit is set to 1b, the user is not allowed to change the Legacy OS Wakeup Support menu option. Default value is 0b. 0 = Enable legacy wakeup support. 1 = Disable legacy wakeup support.
3	Disable Boot Selection Menu	0b	Disable Boot Selection. If this bit is set to 1b, the user is not allowed to change the boot order menu option. Default value is 0b. 0 = Enable 1 = Disable
2	Disable Protocol Selection Menu	0b	Disable Protocol Select. If set to 1b, the user is not allowed to change the boot protocol. Default value is 0b. 0 = Enable 1 = Disable
1	Disable Title Message Display	0b	Disable Title Message. If this bit is set to 1b, the title message displaying the version of the Boot Agent is suppressed; the Control-S message is also suppressed. This is for OEMs who do not wish the boot agent to display any messages at system boot. Default value is 0b. 0 = Enable 1 = Disable
0	Setup Menu	0b	Disable Setup Menu. If this bit is set to 1b, the user is not allowed to invoke the setup menu by pressing Control-S. In this case, the NVM may only be changed via an external program. Default value is 0b. 0 = Enable 1 = Disable

38.20.12 Boot Configuration Block Section Summary Table

Contains the required setup to be used for the boot operations.

Word Offset	Description	Page
0x0000	Boot Signature	2537
0x0001	Block Size	2537
0x0002	Structure Version	2537
0x0003	Reserved	
0x0083 + 1*n, n=0...16	Reserved	
0x0094	iSCSI Flags	2537
0x0095 + 1*n, n=0...1	iSCSI Initiator IP	2538
0x0097 + 1*n, n=0...1	Subnet Mask	2538
0x0099 + 1*n, n=0...1	Gateway IP	2538
0x009B	iSCSI Boot LUN	2538
0x009C + 1*n, n=0...1	iSCSI Target IP	2539
0x009E	iSCSI Target Port	2539
0x009F	Reserved	
0x011F	Reserved	
0x0128	Reserved	
0x0168	VLAN ID	2539
0x0169	Reserved	
0x0173 + 1*n, n=0...2	Reserved	
0x0176	Reserved	
0x017A	Boot LUN	2539
0x017B	VLAN ID	2539
0x017C	Target Boot Order	2539
0x017D	Reserved	
0x017E	Reserved	2540
0x0182	Boot LUN	2540
0x0183	VLAN ID	2540
0x0184	Target Boot Order	2540
0x0185	Reserved	
0x0186	Reserved	
0x018A	Boot LUN	2540
0x018B	VLAN ID	2540
0x018C	Target Boot Order	2540
0x018D	Reserved	2540
0x018E	Reserved	
0x0192	Boot LUN	2540
0x0193	VLAN ID	2541
0x0194	Target Boot Order	2541
0x0195 + 1*n, n=0...44	Reserved	
0x01C2	iSCSI Flags	2541
0x01C3 + 1*n, n=0...1	iSCSI Initiator IP	2541

Word Offset	Description	Page
0x01C5 + 1*n, n=0...1	Subnet Mask	2541
0x01C7 + 1*n, n=0...1	Gateway IP	2541
0x01C9	iSCSI Boot LUN	2541
0x01CA + 1*n, n=0...1	iSCSI Target IP	2541
0x01CC	iSCSI Target Port	2541
0x01CD	Reserved	
0x024D	Reserved	
0x0256	Reserved	
0x0296	VLAN ID	2541
0x0297	Mutual CHAP Password	2541
0x02A1 + 1*n, n=0...2	Reserved	
0x02A4	Reserved	
0x02A8	Boot LUN	2541
0x02A9	VLAN ID	2541
0x02AA	Target Boot Order	2542
0x02AB	Reserved	
0x02AC	Reserved	
0x02B0	Boot LUN	2542
0x02B1	VLAN ID	2542
0x02B2	Target Boot Order	2542
0x02B3	Reserved	
0x02B4	Reserved	
0x02B8	Boot LUN	2542
0x02B9	VLAN ID	2542
0x02BA	Target Boot Order	2542
0x02BB	Reserved	
0x02BC	Reserved	
0x02C0	Boot LUN	2542
0x02C1	VLAN ID	2542
0x02C2	Target Boot Order	2542
0x02C3 + 1*n, n=0...44	Reserved	
0x02F0	iSCSI Flags	2542
0x02F1 + 1*n, n=0...1	iSCSI Initiator IP	2542
0x02F3 + 1*n, n=0...1	Subnet Mask	2542
0x02F5 + 1*n, n=0...1	Gateway IP	2542
0x02F7	iSCSI Boot LUN	2542
0x02F8 + 1*n, n=0...1	iSCSI Target IP	2543
0x02FA	iSCSI Target Port	2543
0x02FB	Reserved	
0x037B	Reserved	
0x0384	Reserved	
0x03C4	VLAN ID	2543
0x03C5	Mutual CHAP Password	

Word Offset	Description	Page
0x03CF + 1*n, n=0...2	Reserved	
0x03D2	Reserved	
0x03D6	Boot LUN	2543
0x03D7	VLAN ID	2543
0x03D8	Target Boot Order	2543
0x03D9	Reserved	
0x03DA	Reserved	
0x03DE	Boot LUN	2543
0x03DF	VLAN ID	2543
0x03E0	Target Boot Order	2543
0x03E1	Reserved	
0x03E2	Reserved	
0x03E6	Boot LUN	2543
0x03E7	VLAN ID	2543
0x03E8	Target Boot Order	2543
0x03E9	Reserved	
0x03EA	Reserved	
0x03EE	Boot LUN	2543
0x03EF	VLAN ID	2543
0x03F0	Target Boot Order	2543
0x03F1 + 1*n, n=0...44	Reserved	
0x041E	iSCSI Flags	2544
0x041F + 1*n, n=0...1	iSCSI Initiator IP	2544
0x0421 + 1*n, n=0...1	Subnet Mask	2544
0x0423 + 1*n, n=0...1	Gateway IP	2544
0x0425	iSCSI Boot LUN	2544
0x0426 + 1*n, n=0...1	iSCSI Target IP	2544
0x0428	iSCSI Target Port	2544
0x0429	Reserved	
0x04A9	Reserved	
0x04B2	Reserved	
0x04F2	VLAN ID	2544
0x04F3	Reserved	
0x04FD + 1*n, n=0...2	Reserved	
0x0500	Reserved	
0x0504	Boot LUN	2544
0x0505	VLAN ID	2544
0x0506	Target Boot Order	2544
0x0507	Reserved	
0x0508	Reserved	
0x050C	Boot LUN	2544
0x050D	VLAN ID	2544
0x050E	Target Boot Order	2544

Word Offset	Description	Page
0x050F	Reserved	
0x0510	Reserved	
0x0514	Boot LUN	2544
0x0515	VLAN ID	2545
0x0516	Target Boot Order	2545
0x0517	Reserved	
0x0518	Reserved	
0x051C	Boot LUN	2545
0x051D	VLAN ID	2545
0x051E	Target Boot Order	2545
0x051F + 1*n, n=0...44	Reserved	

38.20.12.1 Boot Signature (0x0000)

Bits	Field Name	NVM Image Value	Description
15:0	Boot Signature	0x5369	Boot Signature: 'i', 'S' (0x3569)

38.20.12.2 Block Size (0x0001)

Bits	Field Name	NVM Image Value	Description
15:0	Block Size		Length in: 1-byte unit. First Section -> Word: Boot Configuration Block -> Boot Signature Last Section -> Word: Boot Configuration Block -> Reserved

38.20.12.3 Structure Version (0x0002)

Bits	Field Name	NVM Image Value	Description
15:8	Reserved	0x00	Reserved.
7:0	Structure Version	0x01	Version of this structure. Should be set to 0x1.

38.20.12.4 iSCSI Flags (0x0094)

Bits	Field Name	NVM Image Value	Description
15:10	ARP Timeout	0x0F	Timeout value for each try.
9:8	ARP Retries	01b	Retry value.
7:6	Reserved	00b	Reserved.
5:4	Ctrl-D Entry	00b	Ctrl-D Entry. 00b = Enabled 01b = Reserved 10b = Reserved 11b = Disabled — Skip Ctrl-D entry.

Bits	Field Name	NVM Image Value	Description
3:2	Authentication Type	00b	Authentication Type. 00b = None 01b = One Way CHAP 10b = Mutual CHAP 11b - Reserved
1	Enable DHCP for iSCSI Target	1b	Enable DHCP for getting iSCSI target information. 0 = Disabled — Use static target configuration. 1 = Enabled — Use DHCP to get target information by the Option 17 Root Path.
0	Enable DHCP	1b	Enable DHC. 0 = Disabled — Use static configurations from this structure. 1 = Enabled — Overrides configurations retrieved from DHCP.

38.20.12.5 iSCSI Initiator IP[n] (0x0095 + 1*n, n=0...1)

Bits	Field Name	NVM Image Value	Description
15:0	iSCSI Initiator IP	0x0000	Initiator DHCP flag. Not set = This field should contain the initiator IP address. Set = This field is ignored.

38.20.12.6 Subnet Mask[n] (0x0097 + 1*n, n=0...1)

Bits	Field Name	NVM Image Value	Description
15:0	iSCSI Initiator IP	0x0000	Initiator DHCP flag. Not set = This field should contain the subnet mask. Set = This field is ignored.

38.20.12.7 Gateway IP[n] (0x0099 + 1*n, n=0...1)

Bits	Field Name	NVM Image Value	Description
15:0	iSCSI Initiator IP	0x0000	Initiator DHCP flag. Not set = This field should contain the gateway IP address. Set = This field is ignored.

38.20.12.8 iSCSI Boot LUN (0x009B)

Bits	Field Name	NVM Image Value	Description
15:0	iSCSI Boot LUN	0x0000	Target DHCP flag. Not set = iSCSI target LUN number should be specified. Set = This field is ignored.

38.20.12.9 iSCSI Target IP[n] (0x009C + 1*n, n=0...1)

Bits	Field Name	NVM Image Value	Description
15:0	iSCSI Initiator IP	0x0000	Target DHCP flag. Not set = IP address of iSCSI target. Set = This field is ignored.

38.20.12.10iSCSI Target Port (0x009E)

Bits	Field Name	NVM Image Value	Description
15:0	iSCSI Target Port	0x0CBC	Target DHCP flag Not set = TCP port used by iSCSI target. Default is 3260. Set = This field is ignored.

38.20.12.11VLAN ID (0x0168)

Bits	Field Name	NVM Image Value	Description
15:0	VLAN ID	0x0001	Reserved area, since the function is disabled due to Microsoft restrictions. VLAN ID to include the tag in iSCSI boot frames. A valid VLAN ID is between 1 and 4094. Zero means no VLAN tag support.

38.20.12.12Boot LUN (0x017A)

Bits	Field Name	NVM Image Value	Description
15:0	Target LUN	0x0000	Target LUN.

38.20.12.13VLAN ID (0x017B)

Bits	Field Name	NVM Image Value	Description
15:0	VLAN ID	0x0000	VLAN ID for the Port. Default is 0x0.

38.20.12.14Target Boot Order (0x017C)

Bits	Field Name	NVM Image Value	Description
15:8	Reserved	0x00	Reserved for future use. Should be set to 0x0.
7:0	Target Boot Order	0x00	Target Boot Order. Valid range is 0-4, with 0 meaning no boot order.

38.20.12.15Boot LUN (0x0182)

Bits	Field Name	NVM Image Value	Description
15:0	Target LUN	0x0000	Target LUN.

38.20.12.16VLAN ID (0x0183)

Bits	Field Name	NVM Image Value	Description
15:0	VLAN ID	0x0000	VLAN ID for the Port. Default is 0x0.

38.20.12.17Target Boot Order (0x0184)

Bits	Field Name	NVM Image Value	Description
15:8	Reserved	0x00	Reserved for future use. Should be set to 0x0.
7:0	Target Boot Order	0x00	Target Boot Order. Valid range is 0-4, with 0 meaning no boot order.

38.20.12.18Boot LUN (0x018A)

Bits	Field Name	NVM Image Value	Description
15:0	Target LUN	0x0000	Target LUN.

38.20.12.19VLAN ID (0x018B)

Bits	Field Name	NVM Image Value	Description
15:0	VLAN ID	0x0000	VLAN ID for the Port. Default is 0x0.

38.20.12.20Target Boot Order (0x018C)

Bits	Field Name	NVM Image Value	Description
15:8	Reserved	0x00	Reserved for future use. Should be set to 0x0.
7:0	Target Boot Order	0x00	Target Boot Order. Valid range is 0-4, with 0 meaning no boot order.

38.20.12.21Boot LUN (0x0192)

Bits	Field Name	NVM Image Value	Description
15:0	Target LUN	0x0000	Target LUN.

38.20.12.22VLAN ID (0x0193)

Bits	Field Name	NVM Image Value	Description
15:0	VLAN ID	0x0000	VLAN ID for the Port. Default is 0x0.

38.20.12.23Target Boot Order (0x0194)

Bits	Field Name	NVM Image Value	Description
15:8	Reserved	0x00	Reserved for future use. Should be set to 0x0.
7:0	Target Boot Order	0x00	Target Boot Order. Valid range is 0-4, with 0 meaning no boot order.

38.20.12.24iSCSI Flags (0x01C2)

For detail on the inner structure, see Section 38.20.12.4.

38.20.12.25iSCSI Initiator IP[n] (0x01C3 + 1*n, n=0...1)

For detail on the inner structure, see Section 38.20.12.5.

38.20.12.26Subnet Mask[n] (0x01C5 + 1*n, n=0...1)

For detail on the inner structure, see Section 38.20.12.6.

38.20.12.27Gateway IP[n] (0x01C7 + 1*n, n=0...1)

For detail on the inner structure, see Section 38.20.12.7.

38.20.12.28iSCSI Boot LUN (0x01C9)

For detail on the inner structure, see Section 38.20.12.8.

38.20.12.29iSCSI Target IP[n] (0x01CA + 1*n, n=0...1)

For detail on the inner structure, see Section 38.20.12.9.

38.20.12.30iSCSI Target Port (0x01CC)

For detail on the inner structure, see Section 38.20.12.10.

38.20.12.31VLAN ID (0x0296)

For detail on the inner structure, see Section 38.20.12.11.

38.20.12.32Boot LUN (0x02A8)

For detail on the inner structure, see Section 38.20.12.12.

38.20.12.33VLAN ID (0x02A9)

For detail on the inner structure, see Section 38.20.12.13.

38.20.12.34Target Boot Order (0x02AA)

For detail on the inner structure, see Section 38.20.12.14.

38.20.12.35Boot LUN (0x02B0)

For detail on the inner structure, see Section 38.20.12.15.

38.20.12.36VLAN ID (0x02B1)

For detail on the inner structure, see Section 38.20.12.16.

38.20.12.37Target Boot Order (0x02B2)

For detail on the inner structure, see Section 38.20.12.17.

38.20.12.38Boot LUN (0x02B8)

For detail on the inner structure, see Section 38.20.12.18.

38.20.12.39VLAN ID (0x02B9)

For detail on the inner structure, see Section 38.20.12.19.

38.20.12.40Target Boot Order (0x02BA)

For detail on the inner structure, see Section 38.20.12.20.

38.20.12.41Boot LUN (0x02C0)

For detail on the inner structure, see Section 38.20.12.21.

38.20.12.42VLAN ID (0x02C1)

For detail on the inner structure, see Section 38.20.12.22.

38.20.12.43Target Boot Order (0x02C2)

For detail on the inner structure, see Section 38.20.12.23.

38.20.12.44iSCSI Flags (0x02F0)

For detail on the inner structure, see Section 38.20.12.4.

38.20.12.45iSCSI Initiator IP[n] (0x02F1 + 1*n, n=0...1)

For detail on the inner structure, see Section 38.20.12.5.

38.20.12.46Subnet Mask[n] (0x02F3 + 1*n, n=0...1)

For detail on the inner structure, see Section 38.20.12.6.

38.20.12.47Gateway IP[n] (0x02F5 + 1*n, n=0...1)

For detail on the inner structure, see Section 38.20.12.7.

38.20.12.48iSCSI Boot LUN (0x02F7)

For detail on the inner structure, see Section 38.20.12.8.

38.20.12.49iSCSI Target IP[n] (0x02F8 + 1*n, n=0...1)

For detail on the inner structure, see Section 38.20.12.9.

38.20.12.50iSCSI Target Port (0x02FA)

For detail on the inner structure, see Section 38.20.12.10.

38.20.12.51VLAN ID (0x03C4)

For detail on the inner structure, see Section 38.20.12.11.

38.20.12.52Boot LUN (0x03D6)

For detail on the inner structure, see Section 38.20.12.12.

38.20.12.53VLAN ID (0x03D7)

For detail on the inner structure, see Section 38.20.12.13.

38.20.12.54Target Boot Order (0x03D8)

For detail on the inner structure, see Section 38.20.12.14.

38.20.12.55Boot LUN (0x03DE)

For detail on the inner structure, see Section 38.20.12.15.

38.20.12.56VLAN ID (0x03DF)

For detail on the inner structure, see Section 38.20.12.16.

38.20.12.57Target Boot Order (0x03E0)

For detail on the inner structure, see Section 38.20.12.17.

38.20.12.58Boot LUN (0x03E6)

For detail on the inner structure, see Section 38.20.12.18.

38.20.12.59VLAN ID (0x03E7)

For detail on the inner structure, see Section 38.20.12.19.

38.20.12.60Target Boot Order (0x03E8)

For detail on the inner structure, see Section 38.20.12.20.

38.20.12.61Boot LUN (0x03EE)

For detail on the inner structure, see Section 38.20.12.21.

38.20.12.62VLAN ID (0x03EF)

For detail on the inner structure, see Section 38.20.12.22.

38.20.12.63Target Boot Order (0x03F0)

For detail on the inner structure, see Section 38.20.12.23.

38.20.12.64iSCSI Flags (0x041E)

For detail on the inner structure, see Section 38.20.12.4.

38.20.12.65iSCSI Initiator IP[n] (0x041F + 1*n, n=0...1)

For detail on the inner structure, see Section 38.20.12.5.

38.20.12.66Subnet Mask[n] (0x0421 + 1*n, n=0...1)

For detail on the inner structure, see Section 38.20.12.6.

38.20.12.67Gateway IP[n] (0x0423 + 1*n, n=0...1)

For detail on the inner structure, see Section 38.20.12.7.

38.20.12.68iSCSI Boot LUN (0x0425)

For detail on the inner structure, see Section 38.20.12.8.

38.20.12.69iSCSI Target IP[n] (0x0426 + 1*n, n=0...1)

For detail on the inner structure, see Section 38.20.12.9.

38.20.12.70iSCSI Target Port (0x0428)

For detail on the inner structure, see Section 38.20.12.10.

38.20.12.71VLAN ID (0x04F2)

For detail on the inner structure, see Section 38.20.12.11.

38.20.12.72Boot LUN (0x0504)

For detail on the inner structure, see Section 38.20.12.12.

38.20.12.73VLAN ID (0x0505)

For detail on the inner structure, see Section 38.20.12.13.

38.20.12.74Target Boot Order (0x0506)

For detail on the inner structure, see Section 38.20.12.14.

38.20.12.75Boot LUN (0x050C)

For detail on the inner structure, see Section 38.20.12.15.

38.20.12.76VLAN ID (0x050D)

For detail on the inner structure, see Section 38.20.12.16.

38.20.12.77Target Boot Order (0x050E)

For detail on the inner structure, see Section 38.20.12.17.

38.20.12.78Boot LUN (0x0514)

For detail on the inner structure, see Section 38.20.12.18.

38.20.12.79VLAN ID (0x0515)

For detail on the inner structure, see Section 38.20.12.19.

38.20.12.80Target Boot Order (0x0516)

For detail on the inner structure, see Section 38.20.12.20.

38.20.12.81Boot LUN (0x051C)

For detail on the inner structure, see Section 38.20.12.21.

38.20.12.82VLAN ID (0x051D)

For detail on the inner structure, see Section 38.20.12.22.

38.20.12.83Target Boot Order (0x051E)

For detail on the inner structure, see Section 38.20.12.23.

38.20.13 VLAN Configuration Block Section Summary Table

Word Offset	Description	Page
0x0000	Section Header	2545
0x0001	VLAN Block Signature	2545
0x0002	Structure Version and Size	2546
0x0003	Port 0 VLAN Tag	2546
0x0004	Port 1 VLAN Tag	2546
0x0005	Port 2 VLAN Tag	2546
0x0006	Port 3 VLAN Tag	2546

38.20.13.1 Section Header (0x0000)

Bits	Field Name	NVM Image Value	Description
15:0	Block Length		Length in: 2-bytes unit - 1. First Section -> Word: VLAN Configuration Block -> Section Header Last Section -> Word: VLAN Configuration Block -> Port 3 VLAN tag Section length in words.

38.20.13.2 VLAN Block Signature (0x0001)

Bits	Field Name	NVM Image Value	Description
15:0	VLAN Block Signature	0x4C56	'V' =0x56, 'L' = 0x4C.

38.20.13.3 Structure Version and Size (0x0002)

Bits	Field Name	NVM Image Value	Description
15:8	Structure Version and Size	0x0C	Total byte size of the configuration block. 0x06 = 1-port adapter. 0x08 = 2-port adapter. 0x0C = 4-port adapter default value). All other values are reserved.
7:0	Structure Version	0x01	The version of this structure. Should be set to 0x01.

38.20.13.4 Port 0 VLAN Tag (0x0003)

Bits	Field Name	NVM Image Value	Description
15:13	VLAN Priority	000b	The value of VLAN priority (0-7).
12	Reserved	0b	Reserved. Must be set to 0b.
11:0	VLAN Tag ID	0x000	The value of VLAN ID (1-4095). 0 means no VLAN configured for port.

38.20.13.5 Port 1 VLAN Tag (0x0004)

For detail on the inner structure, see Section 38.20.13.4.

38.20.13.6 Port 2 VLAN Tag (0x0005)

For detail on the inner structure, see Section 38.20.13.4.

38.20.13.7 Port 3 VLAN Tag (0x0006)

For detail on the inner structure, see Section 38.20.13.4.

38.20.14 POR Registers Auto-load Module Section Summary Table

Default setup to registers that load on POR events.

Word Offset	Description	Page
0x0000	Module Length	2547
0x0001 - 0x003F	NVM Contents for GLGEN_GPIO_CTL	2547
0x0040 - 0x0041	NVM Contents for GLGEN_LED_CTL	2548
0x0042 - 0x004C	NVM Contents for GLGEN_I2CPARAMS	2548
0x004D - 0x0057	NVM Contents for GLGEN_MDIO_I2C_SEL	2548
0x0058 - 0x005F	NVM Contents for GLGEN_MDIO_CTRL	2549
0x0060 - 0x0063	NVM Contents for GL_MNG_HWARB_CTRL	2549
0x0064 - 0x0086	NVM Contents for PFGEN_PORTNUM_CAR	2549
0x0087 - 0x00A9	NVM contents for PFPM_APM	2550
0x00AA - 0x00B4	NVM Contents for PRTGEN_CNF	2551
0x00B5 - 0x00BF	NVM Contents for PRTPM_GC	2551
0x00C0 - 0x00CA	NVM Contents for PRTGEN_CNF2	2552
0x00CB - 00CE	NVM Contents for GLGEN_CLKSTAT	2553

Word Offset	Description	Page
0x00D7 - 00F9	NVM Contents for PFPCI_FUNC	2553
0x00FA - 00FD	NVM Contents for GLPCI_CNF	2554

38.20.14.1 Module Length (0x0000)

Bits	Field Name	NVM Image Value	Description
15:0	Module Length		Length in: 2-bytes unit - 1. First Section -> Word: POR Registers Auto-Load Module -> Module Length Last Section -> Word: POR Registers Auto-Load Module -> Starting Address Low at TAR_ACC_APRVD_SAI_LIST[0]

38.20.14.2 GLGEN_GPIO_CTL (0x0001 - 0x003F)

38.20.14.2.1 Starting Address Low at GLGEN_GPIO_CTL[n] (0x0001)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLGEN_GPIO_CTL	0x100	
3:0	Type	0x2	

38.20.14.2.2 Starting Address High at GLGEN_GPIO_CTL[n] (0x0002)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLGEN_GPIO_CTL	0x0088	

38.20.14.2.3 Attributes at GLGEN_GPIO_CTL[n] (0x0003)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x01F	
4:3	Skip	00b	
2:0	Width	000b	

38.20.14.2.4 Data Low of GLGEN_GPIO_CTL[n] (0x0004 + 2*n, n=0...29)

38.20.14.2.5 Data High of GLGEN_GPIO_CTL[n] (0x0005 + 2*n, n=0...29)

38.20.14.3 GLGEN_LED_CTL (0x0040 - 0x0041)

38.20.14.3.1 Data Low of GLGEN_LED_CTL (0x0040)

38.20.14.3.2 Data High of GLGEN_LED_CTL (0x0041)

38.20.14.4 GLGEN_I2CPARAMS (0x0042 - 0x004C)

38.20.14.4.1 Starting Address Low at GLGEN_I2CPARAMS[n] (0x0042)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLGEN_I2CPARAMS	0x1AC	
3:0	Type	0x2	

38.20.14.4.2 Starting Address High at GLGEN_I2CPARAMS[n] (0x0043)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLGEN_I2CPARAMS	0x0088	

38.20.14.4.3 Attributes at GLGEN_I2CPARAMS[n] (0x0044)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x004	
4:3	Skip	00b	
2:0	Width	000b	

38.20.14.4.4 Data Low of GLGEN_I2CPARAMS[n] (0x0045 + 2*n, n=0...3)

38.20.14.4.5 Data High of GLGEN_I2CPARAMS[n] (0x0046 + 2*n, n=0...3)

38.20.14.5 GLGEN_MDIO_I2C_SEL (0x004D - 0x0057)

38.20.14.5.1 Starting Address Low at GLGEN_MDIO_I2C_SEL[n] (0x004D)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLGEN_MDIO_I2C_SEL	0x1C0	
3:0	Type	0x2	

38.20.14.5.2 Starting Address High at GLGEN_MDIO_I2C_SEL[n] (0x004E)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLGEN_MDIO_I2C_SEL	0x0088	

38.20.14.5.3 Attributes at GLGEN_MDIO_I2C_SEL[n] (0x004F)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x008	
4:3	Skip	00b	
2:0	Width	000b	

38.20.14.5.4 Data Low of GLGEN_MDIO_I2C_SEL[n] (0x0050 + 2*n, n=0...3)

38.20.14.5.5 Data High of GLGEN_MDIO_I2C_SEL[n] (0x0051 + 2*n, n=0...3)

38.20.14.6 GLGEN_MDIO_CTRL (0x0058 - 0x005F)

38.20.14.6.1 Data Low of GLGEN_MDIO_CTRL[n] (0x0058 + 2*n, n=0...3)

38.20.14.6.2 Data High of GLGEN_MDIO_CTRL[n] (0x0059 + 2*n, n=0...3)

38.20.14.7 GL_MNG_HWARB_CTRL (0x0060 - 0x0063)

38.20.14.7.1 Address Low at GL_MNG_HWARB_CTRL (0x0060)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GL_MNG_HWARB_CTRL	0x130	
3:0	Type	0x1	

38.20.14.7.2 Address High at GL_MNG_HWARB_CTRL (0x0061)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GL_MNG_HWARB_CTRL	0x00B6	

38.20.14.7.3 Data Low of GL_MNG_HWARB_CTRL (0x0062)

38.20.14.7.4 Data High of GL_MNG_HWARB_CTRL (0x0063)

38.20.14.8 PFGEN_PORTNUM_CAR (0x0064 - 0x0086)

38.20.14.8.1 Starting Address Low at PFGEN_PORTNUM_CAR[PF] (0x0064 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PFGEN_PORTNUM_CAR, for PF[0]	0x000	

Bits	Field Name	NVM Image Value	Description
3:0	Type	0x2	

38.20.14.8.2 Starting Address High at PFGEN_PORTNUM_CAR[PF] (0x0065 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PFGEN_PORTNUM_CAR, for PF[0]	0x00B8	

38.20.14.8.3 Attributes at PFGEN_PORTNUM_CAR[PF] (0x0066 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x010	
4:3	Skip	00b	
2:0	Width	000b	

38.20.14.8.4 Data Low of PFGEN_PORTNUM_CAR[PF] (0x0067 + 2*PF, PF=0...15)

38.20.14.8.5 Data High of PFGEN_PORTNUM_CAR[PF] (0x0068 + 2*PF, PF=0...15)

38.20.14.9 PFPM_APM (0x0087 - 0x00A9)

38.20.14.9.1 Starting Address Low at PFPM_APM[PF] (0x0087 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PFPM_APM, for PF[0]	0x080	
3:0	Type	0x2	

38.20.14.9.2 Starting Address High at PFPM_APM[PF] (0x0088 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PFPM_APM, for PF[0]	0x00B8	

38.20.14.9.3 Attributes at PFPM_APM[PF] (0x0089 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x010	
4:3	Skip	00b	
2:0	Width	000b	

38.20.14.9.4 Data Low of PFPM_APM[PF] (0x008A + 2*PF, PF=0...15)

38.20.14.9.5 Data High of PFPM_APM[PF] (0x008B + 2*PF, PF=0...15)

38.20.14.10 PRTGEN_CNF (0x00AA - 0x00B4)

38.20.14.10.1 Starting Address Low at PRTGEN_CNF[PRT] (0x00AA + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTGEN_CNF, for PRT[0]	0x120	
3:0	Type	0x2	

38.20.14.10.2 Starting Address High at PRTGEN_CNF[PRT] (0x00AB + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTGEN_CNF, for PRT[0]	0x00B8	

38.20.14.10.3 Attributes at PRTGEN_CNF[PRT] (0x00AC + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x004	
4:3	Skip	00b	
2:0	Width	000b	

38.20.14.10.4 Data Low of PRTGEN_CNF[PRT] (0x00AD + 2*PRT, PRT=0...3)

38.20.14.10.5 Data High of PRTGEN_CNF[PRT] (0x00AE + 2*PRT, PRT=0...3)

38.20.14.11 PRTPM_GC (0x00B5 - 0x00BF)

38.20.14.11.1 Starting Address Low at PRTPM_GC[PRT] (0x00B5 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTPM_GC, for PRT[0]	0x140	
3:0	Type	0x2	

38.20.14.11.2 Starting Address High at PRTPM_GC[PRT] (0x00B6 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTPM_GC, for PRT[0]	0x00B8	

38.20.14.11.3Attributes at PRTPM_GC[PRT] (0x00B7 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x004	
4:3	Skip	00b	
2:0	Width	000b	

38.20.14.11.4Data Low of PRTPM_GC[PRT] (0x00B8 + 2*PRT, PRT=0...3)

38.20.14.11.5Data High of PRTPM_GC[PRT] (0x00B9 + 2*PRT, PRT=0...3)

38.20.14.12PRTGEN_CNF2 (0x00C0 - 0x00CA)

38.20.14.12.1Starting Address Low at PRTGEN_CNF2[PRT] (0x00C0 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTGEN_CNF2, for PRT[0]	0x160	
3:0	Type	0x2	

38.20.14.12.2Starting Address High at PRTGEN_CNF2[PRT] (0x00C1 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTGEN_CNF2, for PRT[0]	0x00B8	

38.20.14.12.3Attributes at PRTGEN_CNF2[PRT] (0x00C2 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x004	
4:3	Skip	00b	
2:0	Width	000b	

38.20.14.12.4Data Low of PRTGEN_CNF2[PRT] (0x00C3 + 2*PRT, PRT=0...3)

38.20.14.12.5Data High of PRTGEN_CNF2[PRT] (0x00C4 + 2*PRT, PRT=0...3)

38.20.14.13GLGEN_CLKSTAT (0x00CB - 00CE)

38.20.14.13.1Address Low at GLGEN_CLKSTAT (0x00CB)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLGEN_CLKSTAT	0xB8184	
3:0	Type	0x1	

38.20.14.13.2Address High at GLGEN_CLKSTAT (0x00CC)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLGEN_CLKSTAT		

38.20.14.13.3Data Low of GLGEN_CLKSTAT (0x00CD)

38.20.14.13.4Data High of GLGEN_CLKSTAT (0x00CE)

38.20.14.14PFPCI_FUNC (0x00D7 - 00F9)

38.20.14.14.1Starting Address Low at PFPCI_FUNC[PF] (0x00D7 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PFPCI_FUNC, for PF[0]	0xBE200	
3:0	Type	0x2	

38.20.14.14.2Starting Address High at PFPCI_FUNC[PF] (0x00D8 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PFPCI_FUNC, for PF[0]		

38.20.14.14.3Attributes at PFPCI_FUNC[PF] (0x00D9 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x10	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.14.14.4Data Low of PFPCI_FUNC[PF] (0x00DA + 2*PF, PF=0...15)

38.20.14.14.5Data High of PFPCI_FUNC[PF] (0x00DB + 2*PF, PF=0...15)

38.20.14.15GLPCI_CNF (0x00FA - 00FD)

38.20.14.15.1Address Low at GLPCI_CNF (0x00FA)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLPCI_CNF	0xBE4C0	
3:0	Type	0x1	

38.20.14.15.2Address High at GLPCI_CNF (0x00FB)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLPCI_CNF		

38.20.14.15.3Data Low of GLPCI_CNF (0x00FC)

38.20.14.15.4Data High of GLPCI_CNF (0x00FD)

38.20.15 CORER Registers Auto-load Module Section Summary Table

Default setup to registers and internal memories that load on CORER events.

Word Offset	Description	Reference
0x0000	Module Length	2555
0x0001 - 0023	NVM contents for PFGEN_PORTMDIO_NUM	2556
0x002C - 0036	NVM contents for PRTDCB_TDPUC	2556
0x0037 - 0049	NVM contents for GL_SWT_L2TAGTXIB	2557
0x004A - 004B	NVM contents for GLLAN_TSOMSK_F	2557
0x004C - 004D	NVM contents for GLLAN_TSOMSK_M	2557
0x004E - 004F	NVM contents for GLLAN_TSOMSK_L	2557
0x0050 - 0054	NVM contents for GLLAN_TCTL_1	2557
0x0055 - 0056	NVM contents for GL_MDCK_TDAT	2558
0x0057 - 0069	NVM contents for GL_SWT_L2TAGRXEB	2558
0x006E - 0090	NVM contents for PFPM_WUC	2559
0x0091 - 009B	NVM contents for PRTDCB_GENC	2559
0x009C - 009F	NVM contents for GLDCB_GENC	2560
0x00A0 - 00AA	NVM contents for PRTDCB_TFLLPC	2560
0x00AB - 00B5	NVM contents for PRTDCB_TDPMC	2560
0x00B6 - 00C0	NVM contents for PRTDCB_TFPFCC	2561
0x00C5 - 00CF	NVM contents for PRTDCB_TCPMC	2562
0x00D0 - 00DA	NVM contents for PRTDCB_TCPFCPC	2562
0x00DB - 00E5	NVM contents for PRTDCB_TCPFCTCC	2563
0x00EE - 00F8	NVM contents for PRTRPB_SLW	2563
0x00F9 - 0103	NVM contents for PRTRPB_SPS	2564

Word Offset	Description	Reference
0x0104 - 0108	NVM contents for GLRPB_PHW	2564
0x0109 - 010A	NVM contents for GLRPB_PLW	2565
0x010B - 0115	NVM contents for PRTDCB_TCLLPC	2565
0x0122 - 0126	NVM contents for GLSCD_CREDITSPERQUANTA	2566
0x012F - 0151	NVM contents for GLHMC_SDPART	2566
0x0152 - 0174	NVM contents for GLHMC_PFPESDPART	2567
0x0179 - 017C	NVM contents for GLHMC_PEPFFIRSTSD	2567
0x017D - 0181	NVM contents for GLLAN_TCTL_0	2568
0x0182 - 0183	NVM contents for GL_MDCK_TCMD	2568
0x0184 - 0188	NVM contents for GLTLAN_MAX_TCBCMD	2568
0x0195 - 01EC	NVM contents for PRTDCB_RETSTCC	2569
0x01ED - 01F7	NVM contents for PRTDCB_RPPMC	2570
0x01F8 - 0202	NVM contents for PRTDCB_RETSC	2570
0x0203 - 0207	NVM contents for GLDCB_RSPMC	2571
0x020E - 0212	NVM contents for GLLAN_RCTL_1	2571
0x0215 - 0237	NVM contents for GLLAN_PF_RECIPE	2572
0x0238 - 025A	NVM contents for PFLAN_QALLOC	2572
0x025B - 027D	NVM contents for PFGEN_PORTNUM	2573
0x027E - 02A0	NVM contents for PF_VT_PFALLOC	2573
0x02A5 - 02C7	NVM contents for GLANL_L2ULP	2574
0x02DA - 02E4	NVM contents for PRT_L2TAGSEN	2574
0x02E5 - 0307	NVM contents for PFQF_FDALLOC	2575
0x0308 - 0312	NVM contents for PRT_SWT_SCBI	2575
0x0313 - 031D	NVM contents for PRTQF_CTL_0	2576
0x032E - 0331	NVM contents for GLQF_CTL	2577
0x034A - 0354	NVM contents for PRT_SWR_PM_THR	2577
0x0355 - 0358	NVM contents for GLQF_FDEVICTFLAG	2578

38.20.15.1 Module Length (0x0000)

Bits	Field Name	NVM Image Value	Description
15:0	Module Length		Length in: 2 Bytes unit - 1 First Section -> Word: CORER Registers Auto-load Module -> Module Length Last Section -> Word: CORER Registers Auto-load Module -> ANA_MERGE Data

38.20.15.2 PFGEN_PORTMDIO_NUM (0x0001 - 0023)

38.20.15.2.1 Starting Address Low at PFGEN_PORTMDIO_NUM[PF] (0x0001 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PFGEN_PORTMDIO_NUM, for PF[0]	0x3F100	
3:0	Type	0x2	

38.20.15.2.2 Starting Address High at PFGEN_PORTMDIO_NUM[PF] (0x0002 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PFGEN_PORTMDIO_NUM, for PF[0]		

38.20.15.2.3 Attributes at PFGEN_PORTMDIO_NUM[PF] (0x0003 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x10	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.2.4 Data Low of PFGEN_PORTMDIO_NUM[PF] (0x0004 + 2*PF, PF=0...15)

38.20.15.2.5 Data High of PFGEN_PORTMDIO_NUM[PF] (0x0005 + 2*PF, PF=0...15)

38.20.15.3 PRTDCB_TDPUC (0x002C - 0036)

38.20.15.3.1 Starting Address Low at PRTDCB_TDPUC[PRT] (0x002C + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTDCB_TDPUC, for PRT[0]	0x44100	
3:0	Type	0x2	

38.20.15.3.2 Starting Address High at PRTDCB_TDPUC[PRT] (0x002D + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTDCB_TDPUC, for PRT[0]		

38.20.15.3.3 Attributes at PRTDCB_TDPUC[PRT] (0x002E + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x4	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.4 GL_SWT_L2TAGTXIB (0x0037 - 0049)

38.20.15.4.1 Starting Address Low at GL_SWT_L2TAGTXIB[n] (0x0037)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GL_SWT_L2TAGTXIB	0x442B8	
3:0	Type	0x2	

38.20.15.4.2 Starting Address High at GL_SWT_L2TAGTXIB[n] (0x0038)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GL_SWT_L2TAGTXIB		

38.20.15.4.3 Attributes at GL_SWT_L2TAGTXIB[n] (0x0039)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0xB	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.5 GLLAN_TSOMSK_F (0x004A - 004B)

38.20.15.5.1 Data Low of GLLAN_TSOMSK_F (0x004A)

38.20.15.5.2 Data High of GLLAN_TSOMSK_F (0x004B)

38.20.15.6 GLLAN_TSOMSK_M (0x004C - 004D)

38.20.15.6.1 Data Low of GLLAN_TSOMSK_M - 0x004C

38.20.15.6.2 Data High of GLLAN_TSOMSK_M - 0x004D

38.20.15.7 GLLAN_TSOMSK_L (0x004E - 004F)

38.20.15.7.1 Data Low of GLLAN_TSOMSK_L (0x004E)

38.20.15.7.2 Data High of GLLAN_TSOMSK_L (0x004F)

38.20.15.8 GLLAN_TCTL_1 (0x0050 - 0054)

38.20.15.8.1 Starting Address Low at GLLAN_TCTL_1 (0x0050)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLLAN_TCTL_1	0x442F0	
3:0	Type	0x2	

38.20.15.8.2 Starting Address High at GLLAN_TCTL_1 (0x0051)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLLAN_TCTL_1		

38.20.15.8.3 Attributes at GLLAN_TCTL_1 (0x0052)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x2	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.9 GL_MDCK_TDAT - (0x0055 - 0056)

38.20.15.9.1 Data Low of GL_MDCK_TDAT - 0x0055

38.20.15.9.2 Data High of GL_MDCK_TDAT - 0x0056

38.20.15.10 GL_SWT_L2TAGRXEB (0x0057 - 0069)

38.20.15.10.1 Starting Address Low at GL_SWT_L2TAGRXEB[n] (0x0057)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GL_SWT_L2TAGRXEB	0x51000	
3:0	Type	0x2	

38.20.15.10.2 Starting Address High at GL_SWT_L2TAGRXEB[n] (0x0058)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GL_SWT_L2TAGRXEB		

38.20.15.10.3 Attributes at GL_SWT_L2TAGRXEB[n] (0x0059)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x8	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.11PFPM_WUC (0x006E - 0090)

38.20.15.11.1Starting Address Low at PFPM_WUC[PF] (0x006E + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PFPM_WUC, for PF[0]	0x6B200	
3:0	Type	0x2	

38.20.15.11.2Starting Address High at PFPM_WUC[PF] (0x006F + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PFPM_WUC, for PF[0]		

38.20.15.11.3Attributes at PFPM_WUC[PF] (0x0070 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x10	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.11.4Data Low of PFPM_WUC[PF] (0x0071 + 2*PF, PF=0...15)

38.20.15.11.5Data High of PFPM_WUC[PF] (0x0072 + 2*PF, PF=0...15)

38.20.15.12PRTDCB_GENC (0x0091 - 009B)

38.20.15.12.1Starting Address Low at PRTDCB_GENC[PRT] (0x0091 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTDCB_GENC, for PRT[0]	0x83000	
3:0	Type	0x2	

38.20.15.12.2Starting Address High at PRTDCB_GENC[PRT] (0x0092 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTDCB_GENC, for PRT[0]		

38.20.15.12.3Attributes at PRTDCB_GENC[PRT] (0x0093 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x4	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.12.4Data Low of PRTDCB_GENC[PRT] (0x0094 + 2*PRT, PRT=0...3)

38.20.15.12.5Data High of PRTDCB_GENC[PRT] (0x0095 + 2*PRT, PRT=0...3)

38.20.15.13GLDCB_GENC (0x009C - 009F)

38.20.15.13.1Address Low at GLDCB_GENC (0x009C)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLDCB_GENC	0x83044	
3:0	Type	0x1	

38.20.15.13.2Address High at GLDCB_GENC (0x009D)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLDCB_GENC		

38.20.15.13.3Data Low of GLDCB_GENC (0x009E)

38.20.15.13.4Data High of GLDCB_GENC (0x009F)

38.20.15.14PRTDCB_TFLLPC (0x00A0 - 00AA)

38.20.15.14.1Starting Address Low at PRTDCB_TFLLPC[PRT] (0x00A0 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTDCB_TFLLPC, for PRT[0]	0x98000	
3:0	Type	0x2	

38.20.15.14.2Starting Address High at PRTDCB_TFLLPC[PRT] (0x00A1 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTDCB_TFLLPC, for PRT[0]		

38.20.15.14.3Attributes at PRTDCB_TFLLPC[PRT] (0x00A2 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x4	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.15PRTDCB_TDPMC (0x00AB - 00B5)

38.20.15.15.1Starting Address Low at PRTDCB_TDPMC[PRT] (0x00AB + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTDCB_TDPMC, for PRT[0]	0xA0180	
3:0	Type	0x2	

38.20.15.15.2Starting Address High at PRTDCB_TDPMC[PRT] (0x00AC + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTDCB_TDPMC, for PRT[0]		

38.20.15.15.3Attributes at PRTDCB_TDPMC[PRT] (0x00AD + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x4	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.15.4Data Low of PRTDCB_TDPMC[PRT] (0x00AE + 2*PRT, PRT=0...3)

38.20.15.15.5Data High of PRTDCB_TDPMC[PRT] (0x00AF + 2*PRT, PRT=0...3)

38.20.15.16PRTDCB_TFPFCC (0x00B6 - 00C0)

38.20.15.16.1Starting Address Low at PRTDCB_TFPFCC[PRT] (0x00B6 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTDCB_TFPFCC, for PRT[0]	0xA01A0	
3:0	Type	0x2	

38.20.15.16.2Starting Address High at PRTDCB_TFPFCC[PRT] (0x00B7 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTDCB_TFPFCC, for PRT[0]		

38.20.15.16.3Attributes at PRTDCB_TFPFCC[PRT] (0x00B8 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x4	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.17PRTDCB_TCPMC (0x00C5 - 00CF)

38.20.15.17.1Starting Address Low at PRTDCB_TCPMC[PRT] (0x00C5 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTDCB_TCPMC, for PRT[0]	0xA21A0	
3:0	Type	0x2	

38.20.15.17.2Starting Address High at PRTDCB_TCPMC[PRT] (0x00C6 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTDCB_TCPMC, for PRT[0]		

38.20.15.17.3Attributes at PRTDCB_TCPMC[PRT] (0x00C7 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x4	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.17.4Data Low of PRTDCB_TCPMC[PRT] (0x00C8 + 2*PRT, PRT=0...3)

38.20.15.17.5Data High of PRTDCB_TCPMC[PRT] (0x00C9 + 2*PRT, PRT=0...3)

38.20.15.18PRTDCB_TCPFCPC (0x00D0 - 00DA)

38.20.15.18.1Starting Address Low at PRTDCB_TCPFCPC[PRT] (0x00D0 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTDCB_TCPFCPC, for PRT[0]	0xA21C0	
3:0	Type	0x2	

38.20.15.18.2Starting Address High at PRTDCB_TCPFCPC[PRT] (0x00D1 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTDCB_TCPFCPC, for PRT[0]		

38.20.15.18.3Attributes at PRTDCB_TCPFCPC[PRT] (0x00D2 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x4	
4:3	Skip	0x0	

Bits	Field Name	NVM Image Value	Description
2:0	Width	0x0	

38.20.15.19PRTDCB_TCPFCTCC (0x00DB - 00E5)

38.20.15.19.1Starting Address Low at PRTDCB_TCPFCTCC[PRT] (0x00DB + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTDCB_TCPFCTCC, for PRT[0]	0xA21E0	
3:0	Type	0x2	

38.20.15.19.2Starting Address High at PRTDCB_TCPFCTCC[PRT] (0x00DC + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTDCB_TCPFCTCC, for PRT[0]		

38.20.15.19.3Attributes at PRTDCB_TCPFCTCC[PRT] (0x00DD + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x4	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.20PRTRPB_SLW (0x00EE - 00F8)

38.20.15.20.1Starting Address Low at PRTRPB_SLW[PRT] (0x00EE + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTRPB_SLW, for PRT[0]	0xAC6A0	
3:0	Type	0x2	

38.20.15.20.2Starting Address High at PRTRPB_SLW[PRT] (0x00EF + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTRPB_SLW, for PRT[0]		

38.20.15.20.3Attributes at PRTRPB_SLW[PRT] (0x00F0 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x4	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.20.4Data Low of PRTRPB_SLW[PRT] (0x00F1 + 2*PRT, PRT=0...3)

38.20.15.20.5Data High of PRTRPB_SLW[PRT] (0x00F2 + 2*PRT, PRT=0...3)

38.20.15.21PRTRPB_SPS (0x00F9 - 0103)

38.20.15.21.1Starting Address Low at PRTRPB_SPS[PRT] (0x00F9 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTRPB_SPS, for PRT[0]	0xAC7C0	
3:0	Type	0x2	

38.20.15.21.2Starting Address High at PRTRPB_SPS[PRT] (0x00FA + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTRPB_SPS, for PRT[0]		

38.20.15.21.3Attributes at PRTRPB_SPS[PRT] (0x00FB + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x4	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.21.4Data Low of PRTRPB_SPS[PRT] (0x00FC + 2*PRT, PRT=0...3)

38.20.15.21.5Data High of PRTRPB_SPS[PRT] (0x00FD + 2*PRT, PRT=0...3)

38.20.15.22GLRPB_PHW (0x0104 - 0108)

38.20.15.22.1Starting Address Low at GLRPB_PHW (0x0104)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLRPB_PHW	0xAC844	
3:0	Type	0x2	

38.20.15.22.2Starting Address High at GLRPB_PHW (0x0105)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLRPB_PHW		

38.20.15.22.3Attributes at GLRPB_PHW (0x0106)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x2	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.22.4Data Low of GLRPB_PHW (0x0107)

38.20.15.22.5Data High of GLRPB_PHW (0x0108)

38.20.15.23GLRPB_PLW (0x0109 - 010A)

38.20.15.23.1Data Low of GLRPB_PLW - 0x0109

38.20.15.23.2Data High of GLRPB_PLW - 0x010A

38.20.15.24PRTDCB_TCLLPC (0x010B - 0115)

38.20.15.24.1Starting Address Low at PRTDCB_TCLLPC[PRT] (0x010B + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTDCB_TCLLPC, for PRT[0]	0xAE000	
3:0	Type	0x2	

38.20.15.24.2Starting Address High at PRTDCB_TCLLPC[PRT] (0x010C + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTDCB_TCLLPC, for PRT[0]		

38.20.15.24.3Attributes at PRTDCB_TCLLPC[PRT] (0x010D + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x4	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.25GLSCD_CREDITSPERQUANTA (0x0122 - 0126)

38.20.15.25.1Starting Address Low at GLSCD_CREDITSPERQUANTA (0x0122)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLSCD_CREDITSPERQUANTA	0xB2144	
3:0	Type	0x2	

38.20.15.25.2Starting Address High at GLSCD_CREDITSPERQUANTA (0x0123)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLSCD_CREDITSPERQUANTA		

38.20.15.25.3Attributes at GLSCD_CREDITSPERQUANTA (0x0124)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x3	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.26GLHMC_SDPART (0x012F - 0151)

38.20.15.26.1Starting Address Low at GLHMC_SDPART[n] (0x012F)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLHMC_SDPART	0xC0800	
3:0	Type	0x2	

38.20.15.26.2Starting Address High at GLHMC_SDPART[n] (0x0130)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLHMC_SDPART		

38.20.15.26.3Attributes at GLHMC_SDPART[n] (0x0131)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x10	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.26.4Data Low of GLHMC_SDPART[n] (0x0132 + 2*n, n=0...15)

38.20.15.26.5Data High of GLHMC_SDPART[n] (0x0133 + 2*n, n=0...15)

38.20.15.27GLHMC_PFPESDPART (0x0152 - 0174)

38.20.15.27.1Starting Address Low at GLHMC_PFPESDPART[n] (0x0152)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLHMC_PFPESDPART	0xC0880	
3:0	Type	0x2	

38.20.15.27.2Starting Address High at GLHMC_PFPESDPART[n] (0x0153)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLHMC_PFPESDPART		

38.20.15.27.3Attributes at GLHMC_PFPESDPART[n] (0x0154)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x10	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.27.4Data Low of GLHMC_PFPESDPART[n] (0x0155 + 2*n, n=0...15)

38.20.15.27.5Data High of GLHMC_PFPESDPART[n] (0x0156 + 2*n, n=0...15)

38.20.15.28GLHMC_PEPFFIRSTSD (0x0179 - 017C)

38.20.15.28.1Address Low at GLHMC_PEPFFIRSTSD (0x0179)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLHMC_PEPFFIRSTSD	0xC20E4	
3:0	Type	0x1	

38.20.15.28.2Address High at GLHMC_PEPFFIRSTSD (0x017A)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLHMC_PEPFFIRSTSD		

38.20.15.28.3Data Low of GLHMC_PEPFFIRSTSD (0x017B)

38.20.15.28.4Data High of GLHMC_PEPFFIRSTSD (0x017C)

38.20.15.29GLLAN_TCTL_0 (0x017D - 0181)

38.20.15.29.1Starting Address Low at GLLAN_TCTL_0 (0x017D)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLLAN_TCTL_0	0xE6488	
3:0	Type	0x2	

38.20.15.29.2Starting Address High at GLLAN_TCTL_0 (0x017E)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLLAN_TCTL_0		

38.20.15.29.3Attributes at GLLAN_TCTL_0 (0x017F)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x2	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.30GL_MDCK_TCMD - (0x0182 - 0183)

38.20.15.30.1Data Low of GL_MDCK_TCMD - 0x0182

38.20.15.30.2Data High of GL_MDCK_TCMD - 0x0183

38.20.15.31GLTLAN_MAX_TCBCMD (0x0184 - 0188)

38.20.15.31.1Starting Address Low at GLTLAN_MAX_TCBCMD (0x0184)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLTLAN_MAX_TCBCMD	0xE64D4	
3:0	Type	0x2	

38.20.15.31.2Starting Address High at GLTLAN_MAX_TCBCMD (0x0185)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLTLAN_MAX_TCBCMD		

38.20.15.31.3Attributes at GLTLAN_MAX_TCBCMD (0x0186)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x3	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.32PRTDCB_RETSTCC (0x0195 - 01EC)

38.20.15.32.1Starting Address Low at PRTDCB_RETSTCC[n,PRT] (0x0195 + 11*n, n=0...7)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTDCB_RETSTCC, for PRT[0]	0x122180	
3:0	Type	0x2	

38.20.15.32.2Starting Address High at PRTDCB_RETSTCC[n,PRT] (0x0196 + 11*n, n=0...7)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTDCB_RETSTCC, for PRT[0]		

38.20.15.32.3Attributes at PRTDCB_RETSTCC[n,PRT] (0x0197 + 11*n, n=0...7)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x4	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.32.4Data Low of PRTDCB_RETSTCC[n,PRT] (0x0198 + 11*n + 2*PRT, n=0...7, PRT=0...3)

38.20.15.32.5Data High of PRTDCB_RETSTCC[n,PRT] (0x0199 + 11*n + 2*PRT, n=0...7, PRT=0...3)

38.20.15.33PRTDCB_RPPMC (0x01ED - 01F7)

38.20.15.33.1Starting Address Low at PRTDCB_RPPMC[PRT] (0x01ED + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTDCB_RPPMC, for PRT[0]	0x1223A0	
3:0	Type	0x2	

38.20.15.33.2Starting Address High at PRTDCB_RPPMC[PRT] (0x01EE + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTDCB_RPPMC, for PRT[0]		

38.20.15.33.3Attributes at PRTDCB_RPPMC[PRT] (0x01EF + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x4	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.33.4Data High of PRTDCB_RPPMC[PRT] (0x01F1 + 2*PRT, PRT=0...3)

38.20.15.34PRTDCB_RETSC (0x01F8 - 0202)

38.20.15.34.1Starting Address Low at PRTDCB_RETSC[PRT] (0x01F8 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTDCB_RETSC, for PRT[0]	0x1223E0	
3:0	Type	0x2	

38.20.15.34.2Starting Address High at PRTDCB_RETSC[PRT] (0x01F9 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTDCB_RETSC, for PRT[0]		

38.20.15.34.3Attributes at PRTDCB_RETSC[PRT] (0x01FA + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x4	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.34.4Data Low of PRTDCB_RETSC[PRT] (0x01FB + 2*PRT, PRT=0...3)

38.20.15.34.5Data High of PRTDCB_RETSC[PRT] (0x01FC + 2*PRT, PRT=0...3)

38.20.15.35GLDCB_RSPMC (0x0203 - 0207)

38.20.15.35.1Starting Address Low at GLDCB_RSPMC (0x0203)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLDCB_RSPMC	0x122604	
3:0	Type	0x2	

38.20.15.35.2Starting Address High at GLDCB_RSPMC (0x0204)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLDCB_RSPMC		

38.20.15.35.3Attributes at GLDCB_RSPMC (0x0205)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x4	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.36GLLAN_RCTL_1 (0x020E - 0212)

38.20.15.36.1Starting Address Low at GLLAN_RCTL_1 (0x020E)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLLAN_RCTL_1	0x12A504	
3:0	Type	0x2	

38.20.15.36.2Starting Address High at GLLAN_RCTL_1 (0x020F)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLLAN_RCTL_1		

38.20.15.36.3Attributes at GLLAN_RCTL_1 (0x0210)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x2	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.37GLLAN_PF_RECIPE (0x0215 - 0237)

38.20.15.37.1Starting Address Low at GLLAN_PF_RECIPE[n] (0x0215)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLLAN_PF_RECIPE	0x12A5E0	
3:0	Type	0x2	

38.20.15.37.2Starting Address High at GLLAN_PF_RECIPE[n] (0x0216)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLLAN_PF_RECIPE		

38.20.15.37.3Attributes at GLLAN_PF_RECIPE[n] (0x0217)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x10	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.38PFLAN_QALLOC (0x0238 - 025A)

38.20.15.38.1Starting Address Low at PFLAN_QALLOC[PF] (0x0238 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PFLAN_QALLOC, for PF[0]	0x1C0400	
3:0	Type	0x2	

38.20.15.38.2Starting Address High at PFLAN_QALLOC[PF] (0x0239 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PFLAN_QALLOC, for PF[0]		

38.20.15.38.3Attributes at PFLAN_QALLOC[PF] (0x023A + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x10	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.38.4Data Low of PFLAN_QALLOC[PF] (0x023B + 2*PF, PF=0...15)

38.20.15.38.5Data High of PFLAN_QALLOC[PF] (0x023C + 2*PF, PF=0...15)

38.20.15.39PFGEN_PORTNUM (0x025B - 027D)

38.20.15.39.1Starting Address Low at PFGEN_PORTNUM[PF] (0x025B + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PFGEN_PORTNUM, for PF[0]	0x1C0480	
3:0	Type	0x2	

38.20.15.39.2Starting Address High at PFGEN_PORTNUM[PF] (0x025C + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PFGEN_PORTNUM, for PF[0]		

38.20.15.39.3Attributes at PFGEN_PORTNUM[PF] (0x025D + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x10	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.39.4Data Low of PFGEN_PORTNUM[PF] (0x025E + 2*PF, PF=0...15)

38.20.15.39.5Data High of PFGEN_PORTNUM[PF] (0x025F + 2*PF, PF=0...15)

38.20.15.40PF_VT_PFALLOC (0x027E - 02A0)

38.20.15.40.1Starting Address Low at PF_VT_PFALLOC[PF] (0x027E + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PF_VT_PFALLOC, for PF[0]	0x1C0500	
3:0	Type	0x2	

38.20.15.40.2Starting Address High at PF_VT_PFALLOC[PF] (0x027F + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PF_VT_PFALLOC, for PF[0]		

38.20.15.40.3Attributes at PF_VT_PFALLOC[PF] (0x0280 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x10	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.40.4Data Low of PF_VT_PFALLOC[PF] (0x0281 + 2*PF, PF=0...15)

38.20.15.40.5Data High of PF_VT_PFALLOC[PF] (0x0282 + 2*PF, PF=0...15)

38.20.15.41GLANL_L2ULP (0x02A5 - 02C7)

38.20.15.41.1Starting Address Low at GLANL_L2ULP[n] (0x02A5)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLANL_L2ULP	0x1C0A2C	
3:0	Type	0x2	

38.20.15.41.2Starting Address High at GLANL_L2ULP[n] (0x02A6)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLANL_L2ULP		

38.20.15.41.3Attributes at GLANL_L2ULP[n] (0x02A7)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x19	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.42PRT_L2TAGSEN (0x02DA - 02E4)

38.20.15.42.1Starting Address Low at PRT_L2TAGSEN[PRT] (0x02DA + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRT_L2TAGSEN, for PRT[0]	0x1C0B20	
3:0	Type	0x2	

38.20.15.42.2Starting Address High at PRT_L2TAGSEN[PRT] (0x02DB + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRT_L2TAGSEN, for PRT[0]		

38.20.15.42.3Attributes at PRT_L2TAGSEN[PRT] (0x02DC + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x4	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.42.4Data Low of PRT_L2TAGSEN[PRT] (0x02DD + 2*PRT, PRT=0...3)

38.20.15.42.5Data High of PRT_L2TAGSEN[PRT] (0x02DE + 2*PRT, PRT=0...3)

38.20.15.43PFQF_FDALLOC (0x02E5 - 0307)

38.20.15.43.1Starting Address Low at PFQF_FDALLOC[PF] (0x02E5 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PFQF_FDALLOC, for PF[0]	0x246280	
3:0	Type	0x2	

38.20.15.43.2Starting Address High at PFQF_FDALLOC[PF] (0x02E6 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PFQF_FDALLOC, for PF[0]		

38.20.15.43.3Attributes at PFQF_FDALLOC[PF] (0x02E7 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x10	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.43.4Data Low of PFQF_FDALLOC[PF] (0x02E8 + 2*PF, PF=0...15)

38.20.15.43.5Data High of PFQF_FDALLOC[PF] (0x02E9 + 2*PF, PF=0...15)

38.20.15.44PRT_SWT_SCBI (0x0308 - 0312)

38.20.15.44.1Starting Address Low at PRT_SWT_SCBI[PRT] (0x0308 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRT_SWT_SCBI, for PRT[0]	0x256D60	
3:0	Type	0x2	

38.20.15.44.2Starting Address High at PRT_SWT_SCBI[PRT] (0x0309 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRT_SWT_SCBI, for PRT[0]		

38.20.15.44.3Attributes at PRT_SWT_SCBI[PRT] (0x030A + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x4	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.45PRTQF_CTL_0 (0x0313 - 031D)

38.20.15.45.1Starting Address Low at PRTQF_CTL_0[PRT] (0x0313 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTQF_CTL_0, for PRT[0]	0x256E60	
3:0	Type	0x2	

38.20.15.45.2Starting Address High at PRTQF_CTL_0[PRT] (0x0314 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTQF_CTL_0, for PRT[0]		

38.20.15.45.3Attributes at PRTQF_CTL_0[PRT] (0x0315 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x4	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.45.4Data Low of PRTQF_CTL_0[PRT] (0x0316 + 2*PRT, PRT=0...3)

38.20.15.45.5Data High of PRTQF_CTL_0[PRT] (0x0317 + 2*PRT, PRT=0...3)

38.20.15.46GLQF_CTL (0x032E - 0331)

38.20.15.46.1Address Low at GLQF_CTL (0x032E)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLQF_CTL	0x269BA4	
3:0	Type	0x1	

38.20.15.46.2Address High at GLQF_CTL (0x032F)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLQF_CTL		

38.20.15.46.3Data Low of GLQF_CTL 0x0330)

38.20.15.46.4Data High of GLQF_CTL (0x0331)

38.20.15.47PRT_SWR_PM_THR (0x034A - 0354)

38.20.15.47.1Starting Address Low at PRT_SWR_PM_THR[PRT] (0x034A + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRT_SWR_PM_THR, for PRT[0]	0x26CD00	
3:0	Type	0x2	

38.20.15.47.2Starting Address High at PRT_SWR_PM_THR[PRT] (0x034B + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRT_SWR_PM_THR, for PRT[0]		

38.20.15.47.3Attributes at PRT_SWR_PM_THR[PRT] (0x034C + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x4	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.15.48GLQF_FDEVICTFLAG (0x0355 - 0358)

38.20.15.48.1Address Low at GLQF_FDEVICTFLAG (0x0355)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLQF_FDEVICTFLAG	0x270380	
3:0	Type	0x1	

38.20.15.48.2Address High at GLQF_FDEVICTFLAG (0x0356)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLQF_FDEVICTFLAG		

38.20.15.48.3Data Low of GLQF_FDEVICTFLAG (0x0357)

38.20.15.48.4Data High of GLQF_FDEVICTFLAG (0x0358)

38.20.16 GLOBR Registers Auto-Load Module Section Summary Table

Default setup to registers that load on GLOBR events.

Word Offset	Description	Page
0x0000	Module Length	2578
0x000C - 0x0016	NVM Contents for PRTGL_SAL	2579
0x0017 - 0x0021	NVM Contents for PRTGL_SAH	2579
0x0022 - 0x002C	NVM Contents for PRTDCB_MFLCN	2580
0x002D - 0x0037	NVM Contents for PRTTSYN_CTL0	2580
0x0038 - 0x003C	Reserved	
0x003D - 0x003E	Reserved	
0x003F - 0x0049	Reserved	
0x004A - 0x0054	Reserved	
0x0055 - 0x0080	NVM Contents for PRTDCB_FCTTVN	2581
0x0081 - 0x008B	NVM Contents for PRTDCB_FCRTV	2582
0x008C - 0x0096	NVM Contents for PRTDCB_FCCFG	2582

38.20.16.1 Module Length (0x0000)

Bits	Field Name	NVM Image Value	Description
15:0	Module Length		Length in: 2 Bytes unit - 1. First Section -> Word: GLOBR Registers Auto-Load Module -> Module Length Last Section -> Word: GLOBR Registers Auto-Load Module -> Starting Address Low at PRTDCB_FCCFG

38.20.16.2 PRTGL_SAL (0x000C - 0x0016)

38.20.16.2.1 Starting Address Low at PRTGL_SAL[PRT] (0x000C + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTGL_SAL, for PRT[0]	0x120	
3:0	Type	0x2	

38.20.16.2.2 Starting Address High at PRTGL_SAL[PRT] (0x000D + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTGL_SAL, for PRT[0]	0x01E2	

38.20.16.2.3 Attributes at PRTGL_SAL[PRT] (0x000E + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x004	
4:3	Skip	00b	
2:0	Width	000b	

38.20.16.2.4 Data Low of PRTGL_SAL[PRT] (0x000F + 2*PRT, PRT=0...3)

38.20.16.2.5 Data High of PRTGL_SAL[PRT] (0x0010 + 2*PRT, PRT=0...3)

38.20.16.3 PRTGL_SAH (0x0017 - 0x0021)

38.20.16.3.1 Starting Address Low at PRTGL_SAH[PRT] (0x0017 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTGL_SAH, for PRT[0]	0x140	
3:0	Type	0x2	

38.20.16.3.2 Starting Address High at PRTGL_SAH[PRT] (0x0018 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTGL_SAH, for PRT[0]	0x01E2	

38.20.16.3.3 Attributes at PRTGL_SAH[PRT] (0x0019 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x004	
4:3	Skip	00b	

Bits	Field Name	NVM Image Value	Description
2:0	Width	000b	

38.20.16.3.4 Data Low of PRTGL_SAH[PRT] (0x001A + 2*PRT, PRT=0...3)

38.20.16.3.5 Data High of PRTGL_SAH[PRT] (0x001B + 2*PRT, PRT=0...3)

38.20.16.4 PRTDCB_MFLCN (0x0022 - 0x002C)

38.20.16.4.1 Starting Address Low at PRTDCB_MFLCN[PRT] (0x0022 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTDCB_MFLCN, for PRT[0]	0x400	
3:0	Type	0x2	

38.20.16.4.2 Starting Address High at PRTDCB_MFLCN[PRT] (0x0023 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTDCB_MFLCN, for PRT[0]	0x01E2	

38.20.16.4.3 Attributes at PRTDCB_MFLCN[PRT] (0x0024 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x004	
4:3	Skip	00b	
2:0	Width	000b	

38.20.16.4.4 Data Low of PRTDCB_MFLCN[PRT] (0x0025 + 2*PRT, PRT=0...3)

38.20.16.4.5 Data High of PRTDCB_MFLCN[PRT] (0x0026 + 2*PRT, PRT=0...3)

38.20.16.5 PRTTSYN_CTL0 (0x002D - 0x0037)

38.20.16.5.1 Starting Address Low at PRTTSYN_CTL0[PRT] (0x002D + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTTSYN_CTL0 for PRT[0]	0x1E4200	
3:0	Type	0x2	

38.20.16.5.2 Starting Address High at PRTTSYN_CTL0[PRT] (0x002E+ 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTTSYN_CTL0 for PRT[0]		

38.20.16.5.3 Attributes at PRTTSYN_CTL0[PRT] (0x002F + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x4	
4:3	Skip	0x0	
2:0	Width	0x0	

38.20.16.5.4 Data Low of PRTTSYN_CTL0[PRT] (0x0030 + 2*PRT, PRT=0...3)

38.20.16.5.5 Data High of PRTTSYN_CTL0[PRT] (0x0031 + 2*PRT, PRT=0...3)

38.20.16.6 PRTDCB_FCTTVN (0x0055 - 0x0080)

38.20.16.6.1 Starting Address Low at PRTDCB_FCTTVN[n,PRT] (0x0055 + 11*n, n=0...3)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTDCB_FCTTVN, for PRT[0]	0x1E4580	
3:0	Type	0x2	

38.20.16.6.2 Starting Address High at PRTDCB_FCTTVN[n,PRT] (0x0056 + 11*n, n=0...3)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTDCB_FCTTVN, for PRT[0]		

38.20.16.6.3 Attributes at PRTDCB_FCTTVN[n,PRT] (0x0057 + 11*n, n=0...3)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x004	
4:3	Skip	00b	
2:0	Width	000b	

38.20.16.6.4 Data Low of PRTDCB_FCTTVN[n,PRT] (0x0058 + 11*n + 2*PRT, n=0...3, PRT=0...3)

38.20.16.6.5 Data High of PRTDCB_FCTTVN[n,PRT] (0x0059 + 11*n + 2*PRT, n=0...3, PRT=0...3)

38.20.16.7 PRTDCB_FCRTV (0x0081 - 0x008B)

38.20.16.7.1 Starting Address Low at PRTDCB_FCRTV[PRT] (0x0081 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	PRTDCB_FCRTV, for PRT[0]	0x1E4600	
3:0	Type	0x2	

38.20.16.7.2 Starting Address High at PRTDCB_FCRTV[PRT] (0x0082 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTDCB_FCRTV, for PRT[0]		

38.20.16.7.3 Attributes at PRTDCB_FCRTV[PRT] (0x0083 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x004	
4:3	Skip	00b	
2:0	Width	000b	

38.20.16.7.4 Data Low of PRTDCB_FCRTV[PRT] (0x0084 + 2*PRT, PRT=0...3)

38.20.16.7.5 Data High of PRTDCB_FCRTV[PRT] (0x0085 + 2*PRT, PRT=0...3)

38.20.16.8 PRTDCB_FCCFG (0x008C - 0096)

38.20.16.8.1 Starting Address Low at PRTDCB_FCCFG[PRT] (0x008C + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTDCB_FCCFG, for PRT[0]	0x1E4640	
3:0	Type	0x2	

38.20.16.8.2 Starting Address High at PRTDCB_FCCFG[PRT] (0x008D + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTDCB_FCCFG, for PRT[0]		

38.20.16.8.3 Attributes at PRTDCB_FCCFG[PRT] (0x008E + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x004	
4:3	Skip	00b	
2:0	Width	000b	

38.20.16.8.4 Data Low of PRTDCB_FCCFG[PRT] (0x008F + 2*PRT, PRT=0...3)

38.20.16.8.5 Data High of PRTDCB_FCCFG[PRT] (0x0090 + 2*PRT, PRT=0...3)

38.20.17 PE CORER Registers Section Summary Table

Default setup to registers and internal memories that load on CORER events for PE.

Word Offset	Description	Page
0x0000	Module Length	2583
0x0011 - 0x0015	NVM Contents for GLPE_RUPM_GCTL	2584
0x0016 - 0x0017	NVM Contents for GLPE_DUAL40_RUPM	
0x0018 - 0x0022	NVM Contents for PRTPE_RUPM_THRES	2584
0x0023 - 0x002D	NVM Contents for PRTPE_RUPM_CTL	2585
0x002E - 0x0038	NVM Contents for PRTPE_RUPM_PFCPC	2586
0x0039 - 0x0043	NVM Contents for PRTPE_RUPM_PFCTCC	2586
0x0044 - 0x0048	NVM Contents for GLPE_RUPM_PUSHPOOL	2587
0x0049 - 0x004A	NVM Contents for GLPE_RUPM_FLRPOOL	2587
0x004B - 0x004C	NVM Contents for GLPE_RUPM_CQPPOOL	2587
0x004D - 0x004E	NVM Contents for GLPE_RUPM_PTXPOLL	2588
0x004F - 0x0059	NVM Contents for PRTDCB_TCPMC_RLPM	2588
0x005A - 0x0064	NVM Contents for PRTDCB_TCPFCPC_RLPM	2588
0x0065 - 0x006F	NVM Contents for PRTDCB_TCPFCTCC_RLPM	2589
0x0070 - 0x0074	NVM Contents for GLDCB_TGENC_RLPM	2590

38.20.17.1 Module Length (0x0000)

Bits	Field Name	NVM Image Value	Description
15:0	Module Length		Length in: 2 Bytes unit - 1. First Section -> Word: PE CORER Registers -> Module Length Last Section -> Word: PE CORER Registers -> Data Low of GLGEN_DUAL40_RLPM

38.20.17.2 GLPE_RUPM_GCTL (0x0011 - 0x0015)

38.20.17.2.1 Starting Address Low at GLPE_RUPM_GCTL (0x0011)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLPE_RUPM_GCTL	0xDA00	
3:0	Type	0x2	

38.20.17.2.2 Starting Address High at GLPE_RUPM_GCTL (0x0012)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLPE_RUPM_GCTL		

38.20.17.2.3 Attributes at GLPE_RUPM_GCTL (0x0013)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x002	
4:3	Skip	00b	
2:0	Width	000b	

38.20.17.2.4 Data Low of GLPE_RUPM_GCTL (0x0014)

38.20.17.2.5 Data High of GLPE_RUPM_GCTL (0x0015)

38.20.17.3 PRTPE_RUPM_THRES (0x0018 - 0022)

38.20.17.3.1 Starting Address Low at PRTPE_RUPM_THRES[PRT] (0x0018 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTPE_RUPM_THRES, for PRT[0]	0xDA20	
3:0	Type	0x2	

38.20.17.3.2 Starting Address High at PRTPE_RUPM_THRES[PRT] (0x0019 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTPE_RUPM_THRES, for PRT[0]		

38.20.17.3.3 Attributes at PRTPE_RUPM_THRES[PRT] (0x001A + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x004	
4:3	Skip	00b	
2:0	Width	000b	

38.20.17.3.4 Data Low of PRTPE_RUPM_THRES[PRT] (0x001B + 2*PRT, PRT=0...3)

38.20.17.3.5 Data High of PRTPE_RUPM_THRES[PRT] (0x001C + 2*PRT, PRT=0...3)

38.20.17.4 PRTPE_RUPM_CTL (0x0023 - 002D)

38.20.17.4.1 Starting Address Low at PRTPE_RUPM_CTL[PRT] (0x0023 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTPE_RUPM_CTL, for PRT[0]	0xDA40	
3:0	Type	0x2	

38.20.17.4.2 Starting Address High at PRTPE_RUPM_CTL[PRT] (0x0024 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTPE_RUPM_CTL, for PRT[0]		

38.20.17.4.3 Attributes at PRTPE_RUPM_CTL[PRT] (0x0025 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x004	
4:3	Skip	00b	
2:0	Width	000b	

38.20.17.4.4 Data Low of PRTPE_RUPM_CTL[PRT] (0x0026 + 2*PRT, PRT=0...3)

38.20.17.4.5 Data High of PRTPE_RUPM_CTL[PRT] (0x0027 + 2*PRT, PRT=0...3)

38.20.17.5 PRTPE_RUPM_PFCPC (0x002E - 0038)

38.20.17.5.1 Starting Address Low at PRTPE_RUPM_PFCPC[PRT] (0x002E + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTPE_RUPM_PFCPC, for PRT[0]	0xDA80	
3:0	Type	0x2	

38.20.17.5.2 Starting Address High at PRTPE_RUPM_PFCPC[PRT] (0x002F + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTPE_RUPM_PFCPC, for PRT[0]		

38.20.17.5.3 Attributes at PRTPE_RUPM_PFCPC[PRT] (0x0030 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x004	
4:3	Skip	00b	
2:0	Width	000b	

38.20.17.5.4 Data Low of PRTPE_RUPM_PFCPC[PRT] (0x0031 + 2*PRT, PRT=0...3)

38.20.17.5.5 Data High of PRTPE_RUPM_PFCPC[PRT] (0x0032 + 2*PRT, PRT=0...3)

38.20.17.6 PRTPE_RUPM_PFCTCC (0x0039 - 0043)

38.20.17.6.1 Starting Address Low at PRTPE_RUPM_PFCTCC[PRT] (0x0039 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTPE_RUPM_PFCTCC, for PRT[0]	0xDA80	
3:0	Type	0x2	

38.20.17.6.2 Starting Address High at PRTPE_RUPM_PFCTCC[PRT] (0x003A + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTPE_RUPM_PFCTCC, for PRT[0]		

38.20.17.6.3 Attributes at PRTPE_RUPM_PFCTCC[PRT] (0x003B + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x004	
4:3	Skip	00b	
2:0	Width	000b	

38.20.17.6.4 Data Low of PRTPE_RUPM_PFCTCC[PRT] (0x003C + 2*PRT, PRT=0...3)

38.20.17.6.5 Data High of PRTPE_RUPM_PFCTCC[PRT] (0x003D + 2*PRT, PRT=0...3)

38.20.17.7 GLPE_RUPM_PUSHPOOL (0x0044 - 0048)

38.20.17.7.1 Starting Address Low at GLPE_RUPM_PUSHPOOL (0x0044)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLPE_RUPM_PUSHPOOL	0xDAC0	
3:0	Type	0x2	

38.20.17.7.2 Starting Address High at GLPE_RUPM_PUSHPOOL (0x0045)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLPE_RUPM_PUSHPOOL		

38.20.17.7.3 Attributes at GLPE_RUPM_PUSHPOOL (0x0046)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x004	
4:3	Skip	00b	
2:0	Width	000b	

38.20.17.7.4 Data Low of GLPE_RUPM_PUSHPOOL (0x0047)

38.20.17.7.5 Data High of GLPE_RUPM_PUSHPOOL (0x0048)

38.20.17.8 GLPE_RUPM_FLRPOOL (0x0049 - 004A)

38.20.17.8.1 Data Low of GLPE_RUPM_FLRPOOL (0x0049)

38.20.17.8.2 Data High of GLPE_RUPM_FLRPOOL (0x004A)

38.20.17.9 GLPE_RUPM_PTXPOOL (0x004B - 004C)

38.20.17.9.1 Data Low of GLPE_RUPM_PTXPOOL (0x004B)

38.20.17.9.2 Data High of GLPE_RUPM_PTXPOOL (0x004C)

38.20.17.10GLPE_RUPM_CQPPOOL (0x004D - 004E)

38.20.17.10.1Data Low of GLPE_RUPM_CQPPOOL (0x004D)

38.20.17.10.2Data High of GLPE_RUPM_CQPPOOL (0x004F)

38.20.17.11PRTDCB_TCPMC_RLPM (0x004F - 0x0059)

38.20.17.11.1Starting Address Low at PRTDCB_TCPMC_RLPM[PRT] (0x004F + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTDCB_TCPMC_RLPM, for PRT[0]	0x1F1A0	
3:0	Type	0x2	

38.20.17.11.2Starting Address High at PRTDCB_TCPMC_RLPM[PRT] (0x00502 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTDCB_TCPMC_RLPM, for PRT[0]		

38.20.17.11.3Attributes at PRTDCB_TCPMC_RLPM[PRT] (0x0051 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x004	
4:3	Skip	00b	
2:0	Width	000b	

38.20.17.11.4Data Low of PRTDCB_TCPMC_RLPM[PRT] (0x0052 + 2*PRT, PRT=0...3)

38.20.17.11.5Data High of PRTDCB_TCPMC_RLPM[PRT] (0x0053 + 2*PRT, PRT=0...3)

38.20.17.12PRTDCB_TCPFCPC_RLPM (0x005A - 0x0064)

38.20.17.12.1Starting Address Low at PRTDCB_TCPFCPC_RLPM[PRT] (0x005A + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTDCB_TCPFCPC_RLPM, for PRT[0]	0x1F1C0	
3:0	Type	0x2	

38.20.17.12.2Starting Address High at PRTDCB_TCPFCPC_RLPM[PRT] (0x005B + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTDCB_TCPFCPC_RLPM, for PRT[0]	0x001F	

38.20.17.12.3Attributes at PRTDCB_TCPFCPC_RLPM[PRT] (0x005C + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x004	
4:3	Skip	00b	
2:0	Width	000b	

38.20.17.13PRTDCB_TCPFCTCC_RLPM (0x0065 - 0x006F)

38.20.17.13.1Starting Address Low at PRTDCB_TCPFCTCC_RLPM[PRT] (0x0065 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PRTDCB_TCPFCTCC_RLPM, for PRT[0]	0x1F1E0	
3:0	Type	0x2	

38.20.17.13.2Starting Address High at PRTDCB_TCPFCTCC_RLPM[PRT] (0x0066 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PRTDCB_TCPFCTCC_RLPM, for PRT[0]		

38.20.17.13.3Attributes at PRTDCB_TCPFCTCC_RLPM[PRT] (0x0067 + 2*PRT, PRT=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x004	
4:3	Skip	00b	
2:0	Width	000b	

38.20.17.14GLDCB_TGENC_RLPM (0x0070 - 0x0074)

38.20.17.14.1Starting Address Low at GLDCB_TGENC_RLPM (0x0070)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLDCB_TGENC_RLPM	0x1F200	
3:0	Type	0x2	

38.20.17.14.2Starting Address High at GLDCB_TGENC_RLPM (0x0071)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLDCB_TGENC_RLPM	0x001F	

38.20.17.14.3Attributes at GLDCB_TGENC_RLPM (0x0072)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x002	
4:3	Skip	00b	
2:0	Width	000b	

38.20.18 POR CORER Registers Section Summary Table

Default setup to registers and internal memories that load on CORER events that follow POR.

Word Offset	Description	Page
0x0000	Module Length	2590

38.20.18.1 Module Length (0x0000)

Bits	Field Name	NVM Image Value	Description
15:0	Module Length		Length in: 2-byte units - 1 First Section -> Word: POR CORER Registers -> Module Length Last Section -> Word: POR CORER Registers -> Module Length

38.20.19 EMP SR Settings Module Header Section Summary Table

This section contains the modes of operation of the EMP that must be stored in the shadow RAM.

Word Offset	Description	Page
0x0000	Section Header	2592
0x0001	SMBus Slave Addresses	2592
0x0002	SMBus Slave Addresses 2	2592
0x0003	OEM Capabilities	2592
0x0004	Reserved	
0x0005	SR PF Allocations Pointer	2592
0x0006	Max PF and VF Per Port	2593
0x0007	PF LAN Device ID	2593
0x0008	Reserved	
0x0009	PF iSCSI Device ID	2593
0x000A	PF RDMA Device ID	2593
0x000B	VF LAN Device ID	2593
0x000C	Reserved	
0x000D	VF iSCSI Device ID	2593
0x000E	VF RDMA Device Id	2593
0x000F	PF LAN Subsystem ID	2594
0x0010	Reserved	
0x0011	PF iSCSI Subsystem ID	2594
0x0012	PF RDMA Subsystem ID	2594
0x0013	VF LAN Subsystem ID	2594
0x0014	Reserved	
0x0015	VF iSCSI Subsystem ID	2594
0x0016	VF RDMA Subsystem ID	2594
0x0017	MNG MAC Address Pointer	2594
0x0018	PF MAC Address Pointer	2595
0x0019	PHY Capability LAN 0 Pointer	2595
0x001A	PHY Capability LAN 1 Pointer	2595
0x001B	PHY Capability LAN 2 Pointer	2595
0x001C	PHY Capability LAN 3 Pointer	2595
0x001D + 1*n, n=0...7	PFGEN_STATE	2595
0x0025	EXT to INT PHY Mapping LAN 0 Pointer	2596
0x0026	EXT to INT PHY Mapping LAN 1 Pointer	2596
0x0027	EXT to INT PHY Mapping LAN 2 Pointer	2596
0x0028	EXT to INT PHY Mapping LAN 3 Pointer	2596
0x0029	Reserved	
0x002A	Features Enable	2596
0x002B	OCP Configuration Word 0	2597
0x002C	OCP Configuration Word 1	2598
0x002D	OCP Configuration Word 2	2599

Word Offset	Description	Page
0x002E	OCP Signature Word 0	2599
0x002F	OCP Signature Word 1	2599
0x0030	OCP Signature Word 2	2599
0x0031	Current Settings Pointer	2599

38.20.19.1 Section Header (0x0000)

Bits	Field Name	NVM Image Value	Description
15:0	Block Length		Length in: 2 Bytes unit - 1. First Section -> Word: EMP SR Settings Module Header -> Section Header Last Section -> Word: EMP SR Settings Module Header -> Current Settings Pointer Section length in words.

38.20.19.2 SMBus Slave Addresses (0x0001)

Bits	Field Name	NVM Image Value	Description
15:9	SMBus 1 Slave Address	0x4A	Dual address mode only.
8	Reserved	0b	Reserved.
7:1	SMBus 0 Slave Address	0x49	
0	Reserved	0b	Reserved.

38.20.19.3 SMBus Slave Addresses 2 (0x0002)

Bits	Field Name	NVM Image Value	Description
15:9	SMBus 3 Slave Addresses	0x4C	
8	Reserved	0b	Reserved.
7:1	SMBus 2 Slave Addresses	0x4B	
0	Reserved	0b	Reserved.

38.20.19.4 OEM Capabilities (0x0003)

Defines the OEM technologies supported in this device.

Bits	Field Name	NVM Image Value	Description
15:0	Reserved	0x0000	Reserved.

38.20.19.5 SR PF Allocations Pointer (0x0005)

Bits	Field Name	NVM Image Value	Description
15:0	PF Allocations Pointer	0xFFFF	Points to the SR PF Allocations section. For more detail on the SR PF Allocations inner structure, see Section 38.20.21.

38.20.19.6 Max PF and VF per Port (0x0006)

Bits	Field Name	NVM Image Value	Description
15:8	Max VF Per PF	0x80	Max number of VFs allocated to a PF.
7:3	Reserved	0x00	Reserved.
2:0	Max PF Per Port	010b	Max PF Per Port. 000b = 1 PF per port. All other values are reserved.

38.20.19.7 PF LAN Device ID (0x0007)

Bits	Field Name	NVM Image Value	Description
15:0	PF LAN Device ID	0x37CC	

38.20.19.8 PF iSCSI Device ID (0x0009)

Bits	Field Name	NVM Image Value	Description
15:0	PF iSCSI	0x37CC	

38.20.19.9 PF RDMA Device ID (0x000A)

Bits	Field Name	NVM Image Value	Description
15:0	PF RDMADevice ID	0x37CC	

38.20.19.10VF LAN Device ID (0x000B)

Bits	Field Name	NVM Image Value	Description
15:0	VF LAN Device ID	0x37CD	

38.20.19.11VF iSCSI Device ID (0x000D)

Bits	Field Name	NVM Image Value	Description
15:0	VF iSCSI Device ID	0x37CD	

38.20.19.12VF RDMA Device ID (0x000E)

Bits	Field Name	NVM Image Value	Description
15:0	VF RDMA Device ID	0x37CD	

38.20.19.13PF LAN Subsystem ID (0x000F)

Bits	Field Name	NVM Image Value	Description
15:0	PF LAN Subsystem ID	0x0	

38.20.19.14PF iSCSI Subsystem ID (0x0011)

Bits	Field Name	NVM Image Value	Description
15:0	PF iSCSI Subsystem ID	0x0	

38.20.19.15PF RDMA Subsystem ID (0x0012)

Bits	Field Name	NVM Image Value	Description
15:0	PF RDMA Subsystem ID	0x0	

38.20.19.16VF LAN Subsystem ID (0x0013)

Bits	Field Name	NVM Image Value	Description
15:0	VF LAN Subsystem ID	0x0	

38.20.19.17VF iSCSI Subsystem ID (0x0015)

Bits	Field Name	NVM Image Value	Description
15:0	VF iSCSI Subsystem ID	0x0	

38.20.19.18VF RDMA Subsystem ID (0x0016)

Bits	Field Name	NVM Image Value	Description
15:0	VF RDMA Subsystem ID	0x0	

38.20.19.19VF MNG MAC Address Pointer (0x0017)

Bits	Field Name	NVM Image Value	Description
15:0	MNG MAC Address pointer	0xFFFF	Points to the MNG MAC Address section. For more detail on the MNG MAC Address inner structure, see Section 38.20.22.

38.20.19.20PF MAC Address Pointer (0x0018)

Bits	Field Name	NVM Image Value	Description
15:0	PF MAC Address pointer	0xFFFF	Points to the PF MAC Address section. For more detail on the PF MAC Address inner structure, see Section 38.20.23.

38.20.19.21PHY Capability LAN 0 Pointer (0x0019)

Bits	Field Name	NVM Image Value	Description
15:0	PHY	0xFFFF	Points to the PHY Capability Data Structure 0 section. For more detail on the PHY Capability Data Structure 0 inner structure, see Section 38.20.24.

38.20.19.22PHY Capability LAN 1 Pointer (0x001A)

Bits	Field Name	NVM Image Value	Description
15:0	PHY	0xFFFF	Points to the PHY Capability Data Structure 1 section. For more detail on the PHY Capability Data Structure 1 inner structure, see Section 38.20.24.

38.20.19.23PHY Capability LAN 2 Pointer (0x001B)

Bits	Field Name	NVM Image Value	Description
15:0	PHY	0xFFFF	Points to the PHY Capability Data Structure 2 section. For more detail on the PHY Capability Data Structure 2 inner structure, see Section 38.20.24.

38.20.19.24PHY Capability LAN 3 Pointer (0x001C)

Bits	Field Name	NVM Image Value	Description
15:0	PHY	0xFFFF	Points to the PHY Capability Data Structure 3 section. For more detail on the PHY Capability Data Structure 3 inner structure, see Section 38.20.24.

38.20.19.25PFGEN_STATE[n] (0x001D + 1*n, n=0...7)

Bits	Field Name	NVM Image Value	Description
15:12	Reserved	0x0	Reserved.
11	PFGEN_STATE[1 + 2*n].PFSCEN	0b	
10	PFGEN_STATE[1 + 2*n].PFLINKEN	1b	
9	PFGEN_STATE[1 + 2*n].PFFCEN	0b	
8	PFGEN_STATE[1 + 2*n].PFPEEN	0b	
7:4	Reserved	0x0	Reserved.
3	PFGEN_STATE[0 + 2*n].PFSCEN	0b	
2	PFGEN_STATE[0 + 2*n].PFLINKEN	1b	

Bits	Field Name	NVM Image Value	Description
1	PFGEN_STATE[0 + 2*n].PFFCEN	0b	
0	PFGEN_STATE[0 + 2*n].PFPEEN	0b	

38.20.19.26EXT to INT PHY Mapping LAN 0 Pointer (0x0025)

Bits	Field Name	NVM Image Value	Description
15:0	LAN 0 Pointer	0xFFFF	Points to the External to Internal PHY Mapping 0 section. For more detail on the External to Internal PHY Mapping 0 inner structure, see Section 38.20.24.

38.20.19.27EXT to INT PHY Mapping LAN 1 Pointer (0x0026)

Bits	Field Name	NVM Image Value	Description
15:0	LAN 1 Pointer	0xFFFF	Points to the External to Internal PHY Mapping 1 section. For more detail on the External to Internal PHY Mapping 1 inner structure, see Section 38.20.24.

38.20.19.28EXT to INT PHY Mapping LAN 2 Pointer (0x0027)

Bits	Field Name	NVM Image Value	Description
15:0	LAN 2 Pointer	0xFFFF	Points to the External to Internal PHY Mapping 2 section. For more detail on the External to Internal PHY Mapping 2 inner structure, see Section 38.20.24.

38.20.19.29EXT to INT PHY Mapping LAN 3 Pointer (0x0028)

Bits	Field Name	NVM Image Value	Description
15:0	LAN 3 Pointer	0xFFFF	Points to the External to Internal PHY Mapping 3 section. For more detail on the External to Internal PHY Mapping 3 inner structure, see Section 38.20.24.

38.20.19.30Features Enable (0x002A)

Bits	Field Name	NVM Image Value	Description
15:14	Reserved	0x0	Reserved
13	PXE Mode No-drop Policy Supported	0x0	The value indicates if the PXE mode no-drop policy is supported. 0b = No support. 1b = PXE mode no-drop is supported. This mode can be enabled by the Configure No-Drop Policy AQ command. ***Note:*** This field is preserved by Intel NVM update tool.
12	Channel Identifier	0x0	Defines the Ether-type of the channel identifier used to differentiate channels within a port extender. Valid values are: 0x0 = S-tag 0x1 = VLAN ***Note:*** This field is preserved by Intel NVM update tool.

Bits	Field Name	NVM Image Value	Description
11:9	Switching mode	0x0	Reserved
8	Proxy Enabled For Port 3	0x1	If cleared, the proxy functionality embedded in the EMP is disabled on port 3. It could be re-enabled on this port only by resetting this bit to 1b and performing an EMPR cycle. Valid values are: 0x0 = Disabled. 0x1 = Enabled. ***Note:*** This field is preserved by Intel NVM update tool.
7	Proxy Enabled For Port 2	0x1	If cleared, the proxy functionality embedded in the EMP is disabled on port 2. It could be re-enabled on this port only by resetting this bit to 1b and performing an EMPR cycle. Valid values are: 0x0 = Disabled. 0x1 = Enabled. ***Note:*** This field is preserved by Intel NVM update tool.
6	Proxy Enabled For Port 1	0x1	If cleared, the proxy functionality embedded in the EMP is disabled on port 1. It could be re-enabled on this port only by resetting this bit to 1b and performing an EMPR cycle. Valid values are: 0x0 = Disabled. 0x1 = Enabled. ***Note:*** This field is preserved by Intel NVM update tool.
5	Proxy Enabled For Port 0	0x1	If cleared, the proxy functionality embedded in the EMP is disabled on port 0. It could be re-enabled on this port only by resetting this bit to 1b and performing an EMPR cycle. Valid values are: 0x0 = Disabled. 0x1 = Enabled. ***Note:*** This field is preserved by Intel NVM update tool.
4	FW WoL Enable 10G	0x0	If set, enables firmware WoL for 10 GbE link, ***Note:*** This field is preserved by Intel NVM update tool.
3	FW WoL Enable 1G	0x0	If set, enables firmware WoL for 1 GbE link, ***Note:*** This field is preserved by Intel NVM update tool.
2	Reserved	0x1	Reserved
1	EVB Protocols Enabled	0x1	If set, filtering according to IEEE 802.1QBg specification is enabled. Valid values are: 0x0 = Disabled. 0x1 = Enabled. ***Note:*** This field is preserved by Intel NVM update tool.
0	VEB Statistics Disable	0x0	When cleared (default) VEB statistics are enabled. When set, VEB statistics are disabled. This field is preserved by Intel NVM update tool.

38.20.19.31OCP Configuration Word 0 (0x002B)

Bits	Field Name	NVM Image Value	Description
15	OCP Enable	0b	If set, the OCP connector is part of the board.
14:12	Number of Ports	0x4	Number of physical ports in the system.
11:9	Port #3 OCP Connection	0x3	0x0 = OCP port 0 is connected to port 3. 0x1 = OCP port 1 is connected to port 3. 0x2 = OCP port 2 is connected to port 3. 0x3 = OCP port 3 is connected to port 3. 0x7 = No OCP port is connected to port 3. Other values are reserved.

Bits	Field Name	NVM Image Value	Description
8:6	Port #2 OCP Connection	0x2	0x0 = OCP port 0 is connected to port 2. 0x1 = OCP port 1 is connected to port 2. 0x2 = OCP port 2 is connected to port 2. 0x3 = OCP port 3 is connected to port 2. 0x7 = No OCP port is connected to port 2. Other values are reserved.
5:3	Port #1 OCP Connection	0x1	0x0 = OCP port 0 is connected to port 1. 0x1 = OCP port 1 is connected to port 1. 0x2 = OCP port 2 is connected to port 1. 0x3 = OCP port 3 is connected to port 1. 0x7 = No OCP port is connected to port 1. Other values are reserved.
2:0	Port 0 OCP Connection	0x0	0x0 = OCP port 0 is connected to port 0. 0x1 = OCP port 1 is connected to port 0. 0x2 = OCP port 2 is connected to port 0. 0x3 = OCP port 3 is connected to port 0. 0x7 = No OCP port is connected to port 0. Other values are reserved.

38.20.19.32OCP Configuration Word 1 (0x002C)

Bits	Field Name	NVM Image Value	Description
15:14	Reserved	0x0	Reserved
13:11	I2C First Auto-Detect Port	0x1	Defines the first I^2C port used for auto detect of the OCP card type. 000b = I^2C 0. 001b = I^2C 1. 010b = I^2C 2. 011b = I^2C 3. Other values are reserved. ***Note:*** This port should be configured to work in I^2C in GLGEN_MDIO_I2C_SEL. If it is not, the auto-detect flow is aborted. Should be set to the last port of the OCP card.
10:8	I2C Second Auto-Detect Port	0x3	Defines the first I^2C port used for auto detect of the OCP card type. 000b = I^2C 0. 001b = I^2C 1. 010b = I^2C 2. 011b = I^2C 3. Other values are reserved. ***Note:*** This port should be configured to work in I^2C in GLGEN_MDIO_I2C_SEL. If it is not, the auto-detect flow is aborted. Should be set to the last port of the OCP card.
7:0	I2C Second Auto-Detect Address	0xA8	Defines the I^2C address used for auto detect of the OCP card type.

38.20.19.33OCP Configuration word 2 (0x002D)

Bits	Field Name	NVM Image Value	Description
15:12	Port 3 Function Super-feature ID	0x3	LSB nibble for super feature ID of function associated with port 3.
11:8	Port 2 Function Super-feature ID	0x2	LSB nibble for super feature ID of function associated with port 2.
7:4	Port 1 Function Super-feature ID	0x1	LSB nibble for super feature ID of function associated with port 1.
3:0	Port 0 Function Super-feature ID	0x0	LSB nibble for super feature ID of function associated with port 0.

38.20.19.34OCP Signature Word 0 (0x002E)

Bits	Field Name	NVM Image Value	Description
15:0	OCP Signature Word 0	0x4F43	

38.20.19.35OCP Signature Word 1 (0x002F)

Bits	Field Name	NVM Image Value	Description
15:0	OCP Signature Word 1	0x505F	

38.20.19.36OCP Signature Word 2 (0x0030)

Bits	Field Name	NVM Image Value	Description
15:0	OCP Signature Word 2	0x0000	

38.20.20 Current Settings Section

Defines status of the LLDP agent. Each LAN port has independent status.

Word Offset	Description	Reference
0x0000	Section Length	2600
0x0001	Current Factory LLDP Admin Status	2600

38.20.20.1 Section Length - 0x0000

Bits	Field Name	Default NVM Value	Description
15:0	Section Length		Length in words of the section covered by CRC.

38.20.20.2 Current Factory LLDP Admin Status - 0x0001

Bits	Field Name	Default NVM Value	Description
15:12	Port 3	0xF	Defines status of LLDP agent. Applies to LAN Port 3.
11:8	Port 2	0xF	Defines status of LLDP agent. Applies to LAN Port 2.
7:4	Port 1	0xF	Defines status of LLDP agent. Applies to LAN Port 1.
3:0	Port 0	0xF	Defines status of LLDP agent. Applies to LAN Port 0.

38.20.21 SR PF Allocations Section Summary Table

This section contains the allocation of resources per PF that must be stored in the shadow RAM.

Word Offset	Description	Page
0x0000	Section Header	2600
0x0001 + 11*n, n=0...15	PF Flags	2601
0x0002 + 11*n, n=0...15	PF BW	2601
0x0003 + 11*n, n=0...15	PF Allocations - VSI and VEB	2601
0x0004 + 11*n, n=0...15	PF Allocations - MACs	2601
0x0005 + 11*n, n=0...15	PF Allocations - Outer Tags	2602
0x0006 + 11*n, n=0...15	Reserved	
0x0007 + 11*n, n=0...15	PF Allocations - Inner MACs	2602
0x0008 + 11*n, n=0...15	PF Allocations - IPs	2602
0x0009 + 11*n, n=0...15	PF Allocations - Stats	2602
0x000A + 11*n, n=0...15	PF Allocations - Mirror Rules	2603
0x000B + 11*n, n=0...15	PF Allocations - Queue Sets	2603

38.20.21.1 Section Header (0x0000)

Bits	Field Name	NVM Image Value	Description
15:0	Block Length		Length in: 2 Bytes unit - 1. First Section -> Word: SR PF Allocations -> Section Header Last Section -> Word: SR PF Allocations -> PF Allocations - Queue Sets Section length in words.

38.20.21.2 PF Flags[n] (0x0001 + 11*n, n=0...15)

Bits	Field Name	NVM Image Value	Description
15:13	Reserved	0x0	Reserved.
2	Multi_Qset	0b	***Note:*** This field is preserved by the Intel NVM update tool.
1	Reserved	0b	Reserved
0	Load PF MAC Address	1b	Defines if the LAN MAC address of the PF should be added to the filtering table. If this bit is set, only packets that pass the MAC (and if needed, the S-tag) are forwarded to the PF.

38.20.21.3 PF BW[n] (0x0002 + 11*n, n=0...15)

Bits	Field Name	NVM Image Value	Description
15:8	PF Max BW	0x64	Contains the maximum TX bandwidth allocation of the specified partition expressed in percent of the maximum physical port link speed. The percent value ranges from 0 to 100.
7:0	PF Min BW	0x00	Contains the minimum TX bandwidth allocation of the specified partition expressed in percent of the maximum physical port link speed. The percent value ranges from 0 to 100.

38.20.21.4 PF Allocations - VSI and VEB[n] (0x0003 + 11*n, n=0...15)

Defines the allocation of VSIs and VEBs to each PF.

Bits	Field Name	NVM Image Value	Description
15:10	VEBs	0x00	Defines the number of VEBs guaranteed to this PF. A value of zero means no VEBs are guaranteed to this PF.
9:0	VSIs	0x000	Defines the number of VSIs guaranteed to this PF. A value of zero means no VSIs are guaranteed to this PF.

38.20.21.5 PF Allocations - MACs[n] (0x0004 + 11*n, n=0...15)

Defines the allocation of MACs to each PF.

Bits	Field Name	NVM Image Value	Description
15:11	Reserved	0x00	Reserved.
10:0	MACs	0x000	Defines the number of MACs guaranteed to this PF. A value of zero means no MACs are guaranteed to this PF.

38.20.21.6 PF Allocations - Outer Tags[n] (0x0005 + 11*n, n=0...15)

Defines the allocation of S-tags/inner VLANs/outer VLANs to each PF. The type of resource allocated depends on the switching table configuration.

Bits	Field Name	NVM Image Value	Description
15:10	Reserved	0x00	Reserved.
9:0	S-tags/Outer VLANs	0x000	Defines the number of outer tags guaranteed to this PF. A value of zero means no tags are guaranteed to this PF. The tags allocated by this field depends on the image used: 1. For EVB image - indicates the number of S-tags allocated to the PF. 2. For Cloud images - N/A. 3. For CPM image - indicates the number of outer VLANs allocated to the PF.

38.20.21.7 PF Allocations - Inner MACs[n] (0x0007 + 11*n, n=0...15)

Defines the allocation of inner MACs to each PF. Relevant only for cloud images.

Bits	Field Name	NVM Image Value	Description
15:11	Reserved	0x00	Reserved.
10:0	Inner MACs	0x000	Defines the number of inner MACs guaranteed to this PF. A value of zero means no inner MACs are guaranteed to this PF.

38.20.21.8 PF Allocations - IPs[n] (0x0008 + 11*n, n=0...15)

Defines the allocation of IP to each PF. Relevant only for cloud images.

Bits	Field Name	NVM Image Value	Description
15:11	Reserved	0x00	Reserved.
10:0	IPs	0x000	Defines the number of IP addresses guaranteed to this PF. A value of zero means no IP addresses are guaranteed to this PF.

38.20.21.9 PF Allocations - Stats[n] (0x0009 + 11*n, n=0...15)

Defines the allocation of statistics pools to each PF.

Bits	Field Name	NVM Image Value	Description
15:14	Reserved	00b	Reserved.
13:7	SMONPRIOSTATS Pools	0x00	Defines the number of smonPrioStats statistics pools guaranteed to this PF. A value of zero means no pools are guaranteed to this PF.
6:0	SMONVLANSTATS Pools	0x00	Defines the number of smonVlanStats statistics pools guaranteed to this PF. A value of zero means no pools are guaranteed to this PF.

38.20.21.10PF Allocations - Mirror Rules[n] (0x000A + 11*n, n=0...15)

Defines the allocation of mirror rules to each PF.

Bits	Field Name	NVM Image Value	Description
15:7	Reserved	0x000	Reserved.
6:0	Mirror Rules	0x00	Defines the number of mirror rules pools guaranteed to this PF. A value of zero means no rules are guaranteed to this PF.

38.20.21.11PF Allocations - Queue Sets[n] (0x000B + 11*n, n=0...15)

Defines the allocation of Tx queue sets to each PF.

Bits	Field Name	NVM Image Value	Description
15:12	Reserved	0x0	Reserved.
11:0	Queue Sets	0x000	Defines the number of queue sets guaranteed to this PF. A value of zero means no queue sets are guaranteed to this PF.

38.20.22 MNG MAC Address Section Summary Table

Word Offset	Description	Page
0x0000	Section Header	2603
0x0001 + 3*n, n=0...15	LAN Ethernet MAC Address (LSB) MMAL	2604
0x0002 + 3*n, n=0...15	LAN Ethernet MAC Address (Mid) MMAL	2604
0x0003 + 3*n, n=0...15	LAN Ethernet MAC Address (MSB) MMAH	2604

38.20.22.1 Section Header (0x0000)

Bits	Field Name	NVM Image Value	Description
15:0	Block Length		Length in: 2 Bytes unit - 1. First Section -> Word: MNG MAC Address -> Section Header Last Section -> Word: MNG MAC Address -> LAN Ethernet MAC Address (MSB) MMAH Section length in words.

38.20.22.2 LAN Ethernet MAC Address (LSB) MMAL[n] (0x0001 + 3*n, n=0...15)

This word is loaded by firmware to the 16 LS bits of the MMAL[0-3] register of port n.

The index = 4*Port# + MAC#.

It is used as the MAC address when dedicated MAC address mode is used in legacy SMBus.

This is a per-device, per-port value allocated by manufacturing.

Bits	Field Name	NVM Image Value	Description
15:8	Ethernet MAC Address, Byte 1	0xFF	
7:0	Ethernet MAC Address, Byte 0	0xFF	

38.20.22.3 LAN Ethernet MAC Address (MID) MMAL[n] (0x0002 + 3*n, n=0...15)

This word is loaded by firmware to the 16 MS bits of the MMAL[0-3] register.

The index = 4*Port# + MAC#.

It is used as the MAC address when dedicated MAC address mode is used in legacy SMBus.

This is a per-device per-port value allocated by manufacturing.

Bits	Field Name	NVM Image Value	Description
15:8	Ethernet MAC Address, Byte 3	0xFF	
7:0	Ethernet MAC Address, Byte 2	0xFF	

38.20.22.4 LAN Ethernet MAC Address (MSB) MMAH[n] (0x0003 + 3*n, n=0...15)

This word is loaded by firmware to the MMAH[0-3] register.

The index = 4*Port# + MAC#.

It is used as the MAC address when dedicated MAC address mode is used in legacy SMBus.

This is a per-device per-port value allocated by manufacturing.

Bits	Field Name	NVM Image Value	Description
15:8	Ethernet MAC Address, Byte 5	0xFF	
7:0	Ethernet MAC Address, Byte 4	0xFF	

38.20.23 PF MAC Address Section Summary Table

Word Offset	Description	Page
0x0000	Section Header	2605
0x0001 + 4*n, n=0...15	PFPM_SAL0	2605
0x0002 + 4*n, n=0...15	PFPM_SAL1	2605
0x0003 + 4*n, n=0...15	PFPM_SAH0	2606
0x0004 + 4*n, n=0...15	PFPM_SAH1	2606

38.20.23.1 Section Header (0x0000)

Bits	Field Name	NVM Image Value	Description
15:0	Block Length		Length in: 2 Bytes unit - 1. First Section -> Word: PF MAC Address -> Section Header Last Section -> Word: PF MAC Address -> PFPM_SAH1 Section length in words.

38.20.23.2 PFPM_SAL0[n] (0x0001 + 4*n, n=0...15)

This word is loaded by firmware to the appropriate PRTPM_SAL register, bytes [1,0]. One MAC address per enabled PF. This is a per-device, per-enabled PF value allocated by manufacturing.

Bits	Field Name	NVM Image Value	Description
15:0	PFPM_SAL0	0x0000	See respective bits in the PRTPM_SAL register.

38.20.23.3 PFPM_SAL1[n] (0x0002 + 4*n, n=0...15)

This register is loaded by firmware to the appropriate PRTPM_SAL register, bytes [3:2]. One MAC address per enabled PF. This is a per-device, per-enabled PF value allocated by manufacturing.

Bits	Field Name	NVM Image Value	Description
15:0	PFPM_SAL1	0x0000	See respective bits in the PRTPM_SAL register.

38.20.23.4 PFPM_SAH0[n] (0x0003 + 4*n, n=0...15)

This register is loaded by firmware to the appropriate PRTPM_SAH register, bytes [1,0]. One MAC address per enabled PF. This is a per-device, per-enabled PF value allocated by manufacturing.

Bits	Field Name	NVM Image Value	Description
15:0	PFPM_SAH0	0x0000	See respective bits in the PRTPM_SAH register.

38.20.23.5 PFPM_SAH1[n] (0x0004 + 4*n, n=0...15)

This register is loaded by firmware to the appropriate PRTPM_SAH register, bytes [3:2]. One MAC address per enabled PF. This is a per-device, per-enabled PF value allocated by manufacturing.

Bits	Field Name	NVM Image Value	Description
15:0	PFPM_SAH1	0x0000	See respective bits in the PRTPM_SAH register.

38.20.24 PHY Capability Data Structure 0 Section Summary Table

The PHY capabilities data structure contains all the parameters to control the internal and external PHY operation. For example, interface type and mode, etc.

Note that the data structure is repeated per port.

Word Offset	Description	Page
0x0000	Section Length	2607
0x0001	INT-EXT PHY Select	2607
0x0002	Internal PHY Type	2607
0x0003	External PHY Type	2608
0x0004	PHY ID 0	2608
0x0005	PHY ID 1	2609
0x0006	Module Type 0	2609
0x0007	Module Type 1	2610
0x0008	PHY Capabilities Misc0	2610
0x0009	PHY Capabilities Misc1	2611
0x000A	Reserved	
0x000B	PHY capabilities Misc2	2611
0x000C	Re-timer and PCA Address	2612

38.20.24.1 Section Length (0x0000)

The length of the section in words. Note that section length does not include a count for the section length word.

Bits	Field Name	NVM Image Value	Description
15:0	Section Length		Length in: 2 Bytes unit - 1. First Section -> Word: PHY Capability Data Structure 0 -> Section Length Last Section -> Word: PHY Capability Data Structure 0 -> Re-timer and PCA Address

38.20.24.2 INT-EXT PHY Select (0x0001)

Bits	Field Name	NVM Image Value	Description
15:7	Reserved	0x0	Reserved.
6	Use Retimer	0x0	Valid values are: 0x0 = Disabled. 0x1 = Enabled.
5	Enable 1G LPLU	0x0	Valid values are: 0x0 = Disabled. 0x1 = Enabled.
4	Reserved	0x0	Reserved.
3:0	Int_Ext PHY	0x0	Valid values are: 0x0 = Internal PHY only. 0x1 = External 10GBASE-T PHY. 0x2 = External SFP+ module. 0x3 = External QSFP+ module. 0x4 = Reserved All other values are reserved.

38.20.24.3 Internal PHY Type (0x0002)

This parameter is used by firmware to determine the various internal PHY types to be supported on the port. The device might support multiple PHY types on the same port.

One of the PHY types might be enabled due to the result of auto-negotiation, or manually set by the firmware when auto-negotiation is disabled or not used. One bit per PHY type.

Note: 10GBASE-CR1 configuration is not an IEEE standard. However, this configuration could be enabled with switches supporting CR4 to obtain 4x10 Gb/s over QSFP+ module.

Bits	Field Name	NVM Image Value	Description
15:12	Reserved	0x0	Reserved.
11	10GBASE-CR1	0b	Used with 4x10 Gb/s QSFP+ breakout direct attach or optical cables.
10:8	Reserved	0b	
7	SFI	0b	
6	Reserved	0b	Reserved.

Bits	Field Name	NVM Image Value	Description
5	XAUI	0x0	
4	Reserved	0x0	Reserved
3	10GBASE-KR	1b	
2	Reserved	0b	Reserved.
1	1000BASE-KX	0b	
0	Reserved	0b	Reserved

38.20.24.4 External PHY Type (0x0003)

This parameter is used by firmware to determine the various external PHY types to be supported on the port. The device might support multiple PHY types on the same port.

One of the PHY types might be enabled due to the result of auto-negotiation or manually set by the firmware when auto-negotiation is disabled or not used. One bit per PHY type.

Bits	Field Name	NVM Image Value	Description
15:11	Reserved	0x0	Reserved.
10:8	Reserved	0b	Reserved.
7	10GBASE-CR1	0b	4x10 GbE QSFP+ CR over direct attach copper and breakout.
6	10GBASE-SFP+	0b	
5	10GBASE-LR	0b	
4	10GBASE-SR	0b	
3	10GBASE-T	0b	
2	1000BASE-T	0b	
1:0	Reserved	0b	Reserved.

38.20.24.5 PHY ID 0 (0x0004)

This parameter is used by firmware to verify the 10GBASE-T PHY-ID connected on the port.

The format of the PHY ID is defined in IEEE Std 802.3 and contains OUI, manufacturers model number, and revision number of the PHY.

Bits	Field Name	NVM Image Value	Description
15:8	PHY ID Byte1	0x00	
7:0	PHY ID Byte0	0x00	

38.20.24.6 PHY ID 1 (0x0005)

This parameter is used by firmware to verify the 10GBASE-T PHY-ID connected on the port.

The format of the PHY ID is defined in IEEE Std 802.3 and contains OUI, manufacturers model number, and revision number of the PHY.

Bits	Field Name	NVM Image Value	Description
15:8	PHY ID Byte3	0x00	
7:0	PHY ID Byte2	0x00	

38.20.24.7 Module Type 0 (0x0006)

This parameter is used by firmware to find out the module type on the port when connected to external modules. For example, the 10 GbE controller might be connected to an SFP+ or QSFP+ optical or direct attached copper modules. The format of the module type returns the ID and extended ID fields as defined in the SFP+ or QSFP+ specifications.

Bits	Field Name	NVM Image Value	Description
15:8	10G Compliance Code	0x00	SFF standard defines an 8-byte field used to indicate the electrical or optical support modes. These are bit-significant indicators that define the electronic or optical interfaces that are supported by the transceiver. At least one bit must be set in this field. SFP standard defines this field in bytes 3-10 of the standard address space. SFP standard defines this field in bytes 131-138 of the standard address space. For 10 GbE SFP+ modules this byte is decoded as follows: Bit[0] = SFP+ Cu passive. Bit[1] = SFP+ Cu active. Bits[3:2] = Reserved. Bits[7:4] = Decoded according to the SFP+ 10 GbE compliance codes (byte 3): Bit[4] = 10GBASE-SR. Bit[5] = 10GBASE-LR. Bit[6] = 10GBASE-LRM. Bit[7] = 10GBASE-ER.
7:0	Module Identifier	0x3	Defined by SFP+ (Addr 0xA0, Byte 0) or QSFP+ (Addr 128, page 0) specifications. 0x3 = SFP. 0xD = QSFP+. All other values are reserved.

38.20.24.8 Module Type 1 (0x0007)

This parameter is used by firmware to find out the module type on the port when connected to external modules. For example, the 10 GbE controller might be connected to an SFP+ or QSFP+ optical or direct attached copper modules. The format of the module type returns the ID and extended ID fields as defined in the SFP+ or QSFP+ specifications.

Bits	Field Name	NVM Image Value	Description
15:10	Reserved	0x00	Reserved.
9:8	Pause Ability Default	0x0	This parameter is used by the firmware to control the default setting for IEEE 802.3x PAUSE ability of the port. This default is also advertised if auto-negotiation is enabled. Bit 8: Set to 1b to enable IEEE 802.3x Tx link pause ability, or set to 0b to disable pause ability. Bit 9: Set to 1 to enable IEEE 802.3x Rx link pause ability, or set to 0b to disable pause ability. ***Note:*** This field is preserved by Intel NVM update tool.
7:0	1G Compliance Code	0x00	SFF standard defines an 8-byte field used to indicate the electrical or optical support modes. These are bit significant indicators that define the electronic or optical interfaces that are supported by the transceiver. At least one bit must be set in this field. SFP standard defines this field in byte 3-10 of the standard address space. SFP standard defines this field in byte 131-138 of the standard address space. 1 GbE compliance codes are in SFP byte 6. 1 GbE compliance codes are in QSFP byte 134.

38.20.24.9 PHY Capabilities Misc0 (0x0008)

Bits	Field Name	NVM Image Value	Description
15:14	Reserved	0x0	Reserved.
13	Enable 1G SFP 1GBASE-T	0b	When set to 1b, enables 1GBASE-T link through SFP.
12	Disable Link Management FW	0b	When set to 1b, link management firmware is disabled.
11	Enable Module Qualification	0b	When, set to 1b, module qualification process is enabled.
10	Low Power Ability	1b	Certain external PHYs connected to the port may have the ability to support low power mode. Setting the PHY in Low power mode affects normal operation. 0 = Otherwise. 1 = PHY supports low power ability.
9:8	Pause Ability	0x3	Used by the firmware to configure or advertise the IEEE 802.3x PAUSE ability of the port. Bit 0.0 = Set to 1b to enable IEEE 802.3x Tx link pause ability, or set to 0b to disable pause ability. Bit 0.1 = Set to 1b to enable IEEE 802.3x Rx link pause ability, or set to 0b to disable pause ability. ***Note:*** The Link Flow Control (LFC or Link Pause) is enabled through pause ability bits during PHY auto-negotiation and the PFC is enabled through DCBX protocol. The LFC and PFC features are mutually exclusive. So the management should disable link pause on links where PFC should be enabled.

Bits	Field Name	NVM Image Value	Description
7:0	Link Speed	0x8	Used by firmware to configure or advertise various link speeds supported on the port. The device may have the ability to support multiple link speeds on the same port. One of the link speeds may be enabled due to the result of auto-negotiation or manually set by FW when auto-negotiation is disabled or not used. Link Rate Supported on port: Bit 0.0 = Reserved Bit 0.1 = Reserved Bit 0.2 = 1 Gb/s Bit 0.3 = 10 Gb/s Bit 0.4 = Reserved Bit 0.5 = Reserved All other values are reserved. Must be zero.

38.20.24.10PHY Capabilities Misc1 (0x0009)

Bits	Field Name	NVM Image Value	Description
15:8	Multi-Speed Module Timeout	0x64	Configurable number of ms to wait for link establishment when running the multi-speed module flow. ***Note:*** This field is preserved by Intel NVM update tool.
7:0	Reserved	0x40	Reserved

38.20.24.11PHY Capabilities Misc2 (0x000B)

Bits	Field Name	NVM Image Value	Description
15:11	Reserved	0x0	Reserved
10	Reset Method	0x1	Should be set the same for all ports connected to the external PHY. Valid values are: 0x0 = Port expander. 0x1 = SDP.
9:8	Reserved	0x0	Reserved
7:6	External PHY Channel	0x0	The channel # used to configure this port. Valid only if INT-EXT PHY Select word bit 6 (user re-timer) is set for this port.
5:4	External PHY I^2C Connection	0x0	I^2C used to access the external PHY. Valid only if INT-EXT PHY Select word bit 6 (user re-timer) is set for this port.
3	External_-PHY_GPIO_po-larty_1G_LED	0x0	Valid values are: 0x0 = Active low. 0x1 = Active high.
2	External_-PHY_GPIO_en-able_1G_LED	0x0	Valid values are: 0x0 = Disabled. 0x1 = Enabled.
1	Reserved	0x1	Reserved

Bits	Field Name	NVM Image Value	Description
0	Default Low Power Ability	0x0	This is the default value that the system operates in. It has a different meaning for BASE-T and QSFP. For BASE-T, this is ignored if NVM loaded <Low Power Ability> = high power, further this field should be set to high power mode for link to come up at power up. For QSFP+, this is ignored if NVM loaded <Low Power Ability> = low power, further when this field is set to low power mode and a high power QSFP+ module is used then link would not come up until the software device driver changes power mode to high power. 1b = Low Power Mode 0b = High Power Mode ***Note:*** This field is preserved by the Intel NVM update tool.

38.20.24.12Re-Timer and PCA Address (0x000C)

Bits	Field Name	NVM Image Value	Description
15:8	PCA Address	0xDE	Should be set the same for all ports connected to the external PHY.
7:0	Re-timer Address	0xBE	

38.20.25 Feature Configuration Section Summary Table

NVM adaptive feature configuration.

Word Offset	Description	Reference

38.20.26 Immediate Values Section Summary Table

Required for NVM update with existing NUP.

Word Offset	Description	Reference

38.20.27 External-to-Internal PHY Mapping 0 Section Summary Table

This section provides the internal PHY mode to be selected for each external PHY/ module counter part.

The 10 GbE controller provides multiple PHY interface for connecting different types of external PHYs and optical or copper modules. The end user can plug-in different types of modules into the SFP+ or QSFP+ connectors and choose from several connectivity options when using external PHYs. For example, a 10GBASE-T PHY could use a KR connection to connect to the 10 GbE controller.

Firmware has to read the PHY or module ID and choose the appropriate interface to operate with that external module. Furthermore, when working with an external 10GBASE-T PHY, auto-negotiation might result in several speed modes.

Word Offset	Description	Page
0x0000	Section Length	2613
0x0001	Mapping Word0	2613
0x0002	Reserved	

38.20.27.1 Section Length (0x0000)

The length of the section in words. Note that section length does not include a count for the section length word.

Bits	Field Name	NVM Image Value	Description
15:0	Section Length		Length in: 2 Bytes unit - 1. First Section -> Word: External to Internal PHY Mapping 0 -> Section Length Last Section -> Word: External to Internal PHY Mapping 0 -> Mapping Word 0

38.20.27.2 Mapping Word0 (0x0001)

Bits	Field Name	NVM Image Value	Description
15:14	10GBASE-CR1	0x2	Internal interface for 10GBASE-CR1 external module. 00b = Reserved 01b = CR1 (This option may be used for 4x10Gb/s with QSFP+ modules) All other values are reserved.
13:12	10GBASE-SFP+Cu	00b	Internal interface for 10GBASE-SFP+Cu external module. 00b = Reserved 01b = SFI All other values are reserved.
11:10	10GBASE-LR	00b	Internal interface for 10GBASE-LR external module. 00b = Reserved (this option may be used with external SFI PHY) 01b = SFI All other values are reserved.
9:8	10GBASE-SR	00b	Internal interface for 10GBASE-SR external module. 00b = Reserved (this option may be used with external SFI PHY) 01b = SFI All other values are reserved.
7:6	10GBASE-T	0x1	Internal interface for 10GBASE-TX external PHY. 00b = Reserved 01b = 10GBASE-KR All other values are reserved.
5:4	1000BASE-T	00b	Internal interface for 1000BASE-T external PHY. All values are reserved.
3:0	Reserved	0x0	Reserved.

38.20.28 Internal PHY Configuration Override Section Summary Table

Includes static configurations of the internal PHY registers for PLL lock.

Word Offset	Description	Reference
0x0000	Section Length	2615
0x0001	Reserved	2615
0x0002	Port 0 Tx Invert - Addr Low	2615
0x0003	Port 0 Tx Invert - Addr High	2615
0x0004	Port 0 Tx Invert - Data Low	2615
0x0005	Port 0 Tx Invert - Data High	2615
0x0006	Port 1 Tx Invert - Addr Low	2615
0x0007	Port 1 Tx Invert - Addr High	2616
0x0008	Port 1 Tx Invert - Data Low	2616
0x0009	Port 1 Tx Invert - Data High	2616
0x000A	Port 2 Tx Invert - Addr Low	2616
0x000B	Port 2 Tx Invert - Addr High	2616
0x000C	Port 2 Tx Invert - Data Low	2616
0x000D	Port 2 Tx Invert - Data High	2616
0x000E	Port 3 Tx Invert - Addr Low	2616
0x000F	Port 3 Tx Invert - Addr High	2617
0x0010	Port 3 Tx Invert - Data Low	2617
0x0011	Port 3 Tx Invert - Data High	2617
0x0012	Port 0 Rx Invert - Addr Low	2617
0x0013	Port 0 Rx Invert - Addr High	2617
0x0014	Port 0 Rx Invert - Data Low	2617
0x0015	Port 0 Rx Invert - Data High	2617
0x0016	Port 1 Rx Invert - Addr Low	2617
0x0017	Port 1 Rx Invert - Addr High	2618
0x0018	Port 1 Rx Invert - Data Low	2618
0x0019	Port 1 Rx Invert - Data High	2618
0x001A	Port 2 Rx Invert - Addr Low	2618
0x001B	Port 2 Rx Invert - Addr High	2618
0x001C	Port 2 Rx Invert - Data Low	2618
0x001D	Port 2 Rx Invert - Data High	2618
0x001E	Port 3 Rx Invert - Addr Low	2618
0x001F	Port 3 Rx Invert - Addr High	2619
0x0020	Port 3 Rx Invert - Data Low	2619
0x0021	Port 3 Rx Invert - Data High	2619
0x0022	Recipe Raw Patch	2619

38.20.28.1 Section Length (0x0000)

Bits	Field Name	NVM Image Value	Description
15:0	Section Length		Length in: 2 Bytes unit - 1 First Section -> Word: Internal PHY Configuration Override -> Section Length Last Section -> Word: Internal PHY Configuration Override -> Recipe Raw Patch

38.20.28.2 Reserved (0x0001)

Bits	Field Name	NVM Image Value	Description
15:0	Recipe Type	0xFFFF	0x0 = SFI. 0x1 = KR. 0x2 =SGMII. 0xFFFF = Reserved.

38.20.28.3 Port 0 Tx Invert - Addr Low (0x0002)

Bits	Field Name	NVM Image Value	Description
15:0	Addr Low	0x5500	

38.20.28.4 Port 0 Tx Invert - Addr High (0x0003)

Bits	Field Name	NVM Image Value	Description
15:0	Addr High	0x0000	

38.20.28.5 Port 0 Tx Invert - Data Low (0x0004)

Bits	Field Name	NVM Image Value	Description
15:0	Data Low	0x0000	

38.20.28.6 Port 0 Tx Invert - Data High (0x0005)

Bits	Field Name	NVM Image Value	Description
15:0	Data High	0xE2AA	0xE2AA = Default. 0xFAAA = Inverted.

38.20.28.7 Port 1 Tx Invert - Addr Low (0x0006)

Bits	Field Name	NVM Image Value	Description
15:0	Addr Low	0x9500	

38.20.28.8 Port 1 Tx Invert - Addr High (0x0007)

Bits	Field Name	NVM Image Value	Description
15:0	Addr High	0x0000	

38.20.28.9 Port 1 Tx Invert - Data Low (0x0008)

Bits	Field Name	NVM Image Value	Description
15:0	Data Low	0x0000	

38.20.28.10Port 1 Tx Invert - Data High (0x0009)

Bits	Field Name	NVM Image Value	Description
15:0	Data High	0xE2AA	0xE2AA = Default. 0xFAAA = Inverted.

38.20.28.11Port 2 Tx Invert - Addr Low (0x000A)

Bits	Field Name	NVM Image Value	Description
15:0	Addr Low	0x5500	

38.20.28.12Port 2 Tx Invert - Addr High (0x000B)

Bits	Field Name	NVM Image Value	Description
15:0	Addr High	0x1000	

38.20.28.13Port 2 Tx Invert - Data Low (0x000C)

Bits	Field Name	NVM Image Value	Description
15:0	Data Low	0x0000	

38.20.28.14Port 2 Tx Invert - Data High (0x000D)

Bits	Field Name	NVM Image Value	Description
15:0	Data High	0xE2AA	0xE2AA = Default. 0xFAAA = Inverted.

38.20.28.15Port 3 Tx Invert - Addr Low (0x000E)

Bits	Field Name	NVM Image Value	Description
15:0	Addr Low	0x9500	

38.20.28.16Port 3 Tx Invert - Addr High (0x000F)

Bits	Field Name	NVM Image Value	Description
15:0	Addr High	0x1000	

38.20.28.17Port 3 Tx Invert - Data Low (0x0010)

Bits	Field Name	NVM Image Value	Description
15:0	Data Low	0x0000	

38.20.28.18Port 3 Tx Invert - Data High (0x0011)

Bits	Field Name	NVM Image Value	Description
15:0	Data High	0xE2AA	0xE2AA = Default. 0xFAAA = Inverted.

38.20.28.19Port 0 Rx Invert - Addr Low (0x0012)

Bits	Field Name	NVM Image Value	Description
15:0	Addr Low	0x4A00	

38.20.28.20Port 0 Rx Invert - Addr High (0x0013)

Bits	Field Name	NVM Image Value	Description
15:0	Addr High	0x0000	

38.20.28.21Port 0 Rx Invert - Data Low (0x0014)

Bits	Field Name	NVM Image Value	Description
15:0	Data Low	0x0000	0x0000 = Default. 0x0002 = Inverted.

38.20.28.22Port 0 Rx Invert - Data High (0x00150)

Bits	Field Name	NVM Image Value	Description
15:0	Data High	0x0000	

38.20.28.23Port 1 Rx Invert - Addr Low (0x00160)

Bits	Field Name	NVM Image Value	Description
15:0	Addr Low	0x8A00	

38.20.28.24Port 1 Rx Invert - Addr High (0x00170)

Bits	Field Name	NVM Image Value	Description
15:0	Addr High	0x0000	

38.20.28.25Port 1 Rx Invert - Data Low (0x00180)

Bits	Field Name	NVM Image Value	Description
15:0	Data Low	0x0000	0x0000 = Default. 0x0002 = Inverted.

38.20.28.26Port 1 Rx Invert - Data High (0x0019)

Bits	Field Name	NVM Image Value	Description
15:0	Data High	0x0000	

38.20.28.27Port 2 Rx Invert - Addr Low (0x001A)

Bits	Field Name	NVM Image Value	Description
15:0	Addr Low	0x4A00	

38.20.28.28Port 2 Rx Invert - Addr High (0x001B)

Bits	Field Name	NVM Image Value	Description
15:0	Addr High	0x1000	

38.20.28.29Port 2 Rx Invert - Data Low (0x001C)

Bits	Field Name	NVM Image Value	Description
15:0	Data Low	0x0000	0x0000 = Default. 0x0002 = Inverted.

38.20.28.30Port 2 Rx Invert - Data High (0x001D)

Bits	Field Name	NVM Image Value	Description
15:0	Data High	0x0000	

38.20.28.31Port 3 Rx Invert - Addr Low (0x001E)

Bits	Field Name	NVM Image Value	Description
15:0	Addr Low	0x8A00	

38.20.28.32Port 3 Rx Invert - Addr High (0x001F)

Bits	Field Name	NVM Image Value	Description
15:0	Addr High	0x1000	

38.20.28.33Port 3 Rx Invert - Data Low (0x0020)

Bits	Field Name	NVM Image Value	Description
15:0	Data Low	0x0000	0x0000 = Default. 0x0002 = Inverted.

38.20.28.34Port 3 Rx Invert - Data High (0x0021)

Bits	Field Name	NVM Image Value	Description
15:0	Data High	0x0000	

38.20.28.35Recipe Raw Patch (0x0022)

Raw data module length: variable.

38.20.29 PCIR Registers Auto-load Module Section Summary Table

Default setup to registers and internal memories that load on PCIR events.

Word Offset	Description	Page
0x0000	Module Length	2620
0x0001 - 0x0023	NVM Contents for PFPCI_CNF	2620
0x0024 - 0x0046	NVM Contents for PFPCI_DEVID	2621
0x0047 - 0x0069	NVM Contents for PFPCI_SUBSYSID	2621
0x006A - 0x008C	NVM Contents for PFPCI_FUNC2	2622
0x008D - 0x00AF	NVM Contents for PF_VT_PFALLOC_PCIE	2622
0x00B0 - 0x00D2	NVM Contents for PFPCI_CLASS	2623
0x00D3 - 0x00D6	Reserved	
0x00D7 - 0x00DB	NVM Contents for GLPCI_SUBVENID	2624
0x00DC - 0x00DD	NVM Contents for GLPCI_PWRDATA	2624
0x00DE - 0x00DF	NVM Contents for GLPCI_CNF2	2624
0x00E0 - 0x00E1	NVM Contents for GLPCI_SERL	2624
0x00E2 - 0x00E3	NVM Contents for GLPCI_SERH	2624
0x00E4 - 0x00E8	NVM Contents for GLPCI_CAPCTRL	2625
0x00E9 - 0x00EA	NVM Contents for GLPCI_CAPSUP	2625
0x00EB - 0x00EC	NVM Contents for GLPCI_LINKCAP	2625
0x00ED - 0x00EE	NVM Contents for GLPCI_PMSUP	2625
0x00EF - 0x00F0	NVM Contents for GLPCI_REVID	2625
0x00F1 - 0x00F2	NVM Contents for GLPCI_VFSUP	2626
0x00F3 - 0x00F6	Reserved	
0x00F7 - 0x00FA	Reserved	

Word Offset	Description	Page
0x00FB - 0x00FE	NVM Contents for GLPCI_VENDORID	2626
0x00FF - 0x0102	NVM Contents for GLGEN_PCIFCNCNT	2626
0x0103	Reserved	
0x0104	Reserved	
0x0105	LCB Attributes	2627
0x0106	LCB Data	

38.20.29.1 Module Length (0x0000)

Bits	Field Name	NVM Image Value	Description
15:0	Module Length		Length in: 2-bytes unit - 1. First Section -> Word: PCIR Registers Auto-Load Module -> Module Length Last Section -> Word: PCIR Registers Auto-Load Module -> LCB Data

38.20.29.2 PFPCI_CNF (0x0001 - 0x0023)

38.20.29.2.1 Starting Address Low at PFPCI_CNF[PF] (0x0001 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PFPCI_CNF, for PF[0]	0x000	
3:0	Type	0x2	

38.20.29.2.2 Starting Address High at PFPCI_CNF[PF] (0x0002 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PFPCI_CNF, for PF[0]	0x00BE	

38.20.29.2.3 Attributes at PFPCI_CNF[PF] (0x0003 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x010	
4:3	Skip	00b	
2:0	Width	000b	

38.20.29.2.4 Data Low of PFPCI_CNF[PF] (0x0004 + 2*PF, PF=0...15)

38.20.29.2.5 Data High of PFPCI_CNF[PF] (0x0005 + 2*PF, PF=0...15)

38.20.29.3 PFPCI_DEVID (0x0024 - 0x0046)

38.20.29.3.1 Starting Address Low at PFPCI_DEVID[PF] (0x0024 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PFPCI_DEVID, for PF[0]	0x080	
3:0	Type	0x2	

38.20.29.3.2 Starting Address High at PFPCI_DEVID[PF] (0x0025 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PFPCI_DEVID, for PF[0]	0x00BE	

38.20.29.3.3 Attributes at PFPCI_DEVID[PF] (0x0026 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x010	
4:3	Skip	00b	
2:0	Width	000b	

38.20.29.3.4 Data Low of PFPCI_DEVID[PF] (0x0027 + 2*PF, PF=0...15)

38.20.29.3.5 Data High of PFPCI_DEVID[PF] (0x0028 + 2*PF, PF=0...15)

38.20.29.4 PFPCI_SUBSYSID (0x0047 - 0x0069)

38.20.29.4.1 Starting Address Low at PFPCI_SUBSYSID[PF] (0x0047 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PFPCI_SUBSYSID, for PF[0]	0x100	
3:0	Type	0x2	

38.20.29.4.2 Starting Address High at PFPCI_SUBSYSID[PF] (0x0048 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PFPCI_SUBSYSID, for PF[0]	0x00BE	

38.20.29.4.3 Attributes at PFPCI_SUBSYSID[PF] (0x0049 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x010	
4:3	Skip	00b	
2:0	Width	000b	

38.20.29.4.4 Data Low of PFPCI_SUBSYSID[PF] (0x004A + 2*PF, PF=0...15)

38.20.29.4.5 Data High of PFPCI_SUBSYSID[PF] (0x004B + 2*PF, PF=0...15)

38.20.29.5 PFPCI_FUNC2 (0x006A - 0x008C)

38.20.29.5.1 Starting Address Low at PFPCI_FUNC2[PF] (0x006A + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PFPCI_FUNC2, for PF[0]	0x180	
3:0	Type	0x2	

38.20.29.5.2 Starting Address High at PFPCI_FUNC2[PF] (0x006B + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PFPCI_FUNC2, for PF[0]	0x00BE	

38.20.29.5.3 Attributes at PFPCI_FUNC2[PF] (0x006C + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x010	
4:3	Skip	00b	
2:0	Width	000b	

38.20.29.5.4 Data Low of PFPCI_FUNC2[PF] (0x006D + 2*PF, PF=0...15)

38.20.29.5.5 Data High of PFPCI_FUNC2[PF] (0x006E + 2*PF, PF=0...15)

38.20.29.6 PF_VT_PFALLOC_PCIE (0x008D - 0x00AF)

38.20.29.6.1 Starting Address Low at PF_VT_PFALLOC_PCIE[PF] (0x008D + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PF_VT_PFALLOC_PCIE, for PF[0]	0x380	
3:0	Type	0x2	

38.20.29.6.2 Starting Address High at PF_VT_PFALLOC_PCIE[PF] (0x008E + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PF_VT_PFALLOC_PCIE, for PF[0]	0x00BE	

38.20.29.6.3 Attributes at PF_VT_PFALLOC_PCIE[PF] (0x008F + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x010	
4:3	Skip	00b	
2:0	Width	000b	

38.20.29.6.4 Data Low of PF_VT_PFALLOC_PCIE[PF] (0x0090 + 2*PF, PF=0...15)

38.20.29.6.5 Data High of PF_VT_PFALLOC_PCIE[PF] (0x0091 + 2*PF, PF=0...15)

38.20.29.7 PFPCI_CLASS (0x00B0 - 0x00D2)

38.20.29.7.1 Starting Address Low at PFPCI_CLASS[PF] (0x00B0 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of PFPCI_CLASS, for PF[0]	0x400	
3:0	Type	0x2	

38.20.29.7.2 Starting Address High at PFPCI_CLASS[PF] (0x00B1 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of PFPCI_CLASS, for PF[0]	0x00BE	

38.20.29.7.3 Attributes at PFPCI_CLASS[PF] (0x00B2 + 2*PF, PF=0)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x010	
4:3	Skip	00b	
2:0	Width	000b	

38.20.29.7.4 Data Low of PFPCI_CLASS[PF] (0x00B3 + 2*PF, PF=0...15)

38.20.29.7.5 Data High of PFPCI_CLASS[PF] (0x00B4 + 2*PF, PF=0...15)

38.20.29.8 GLPCI_SUBVENID (0x00D7 - 0x00DB)

38.20.29.8.1 Starting Address Low at GLPCI_SUBVENID (0x00D7)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLPCI_SUBVENID	0x48C	
3:0	Type	0x2	

38.20.29.8.2 Starting Address High at GLPCI_SUBVENID (0x00D8)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLPCI_SUBVENID	0x00BE	

38.20.29.8.3 Attributes at GLPCI_SUBVENID (0x00D9)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x005	
4:3	Skip	00b	
2:0	Width	000b	

38.20.29.8.4 Data Low of GLPCI_SUBVENID (0x00DA)

38.20.29.8.5 Data High of GLPCI_SUBVENID (0x00DB)

38.20.29.9 GLPCI_PWRDATA (0x00DC - 0x00DD)

38.20.29.9.1 Data Low of GLPCI_PWRDATA (0x00DC)

38.20.29.9.2 Data High of GLPCI_PWRDATA (0x00DD)

38.20.29.10 GLPCI_CNF2 (0x00DE - 0x00DF)

38.20.29.10.1 Data Low of GLPCI_CNF2 (0x00DE)

38.20.29.10.2 Data High of GLPCI_CNF2 (0x00DF)

38.20.29.11 GLPCI_SERL (0x00E0 - 0x00E1)

38.20.29.11.1 Data Low of GLPCI_SERL (0x00E0)

38.20.29.11.2 Data High of GLPCI_SERL (0x00E1)

38.20.29.12 GLPCI_SERH (0x00E2 - 0x00E3)

38.20.29.12.1Data Low of GLPCI_SERH (0x00E2)

38.20.29.12.2Data High of GLPCI_SERH (0x00E3)

38.20.29.13GLPCI_CAPCTRL (0x00E4 - 0x00E8)

38.20.29.13.1Starting Address Low at GLPCI_CAPCTRL (0x00E4)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLPCI_CAPCTRL	0x4A4	
3:0	Type	0x2	

38.20.29.13.2Starting Address High at GLPCI_CAPCTRL (0x00E5)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLPCI_CAPCTRL	0x00BE	

38.20.29.13.3Attributes at GLPCI_CAPCTRL (0x00E6)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x006	
4:3	Skip	00b	
2:0	Width	000b	

38.20.29.13.4Data Low of GLPCI_CAPCTRL (0x00E7)

38.20.29.13.5Data High of GLPCI_CAPCTRL (0x00E8)

38.20.29.14GLPCI_CAPSUP (0x00E9 - 0x00EA)

38.20.29.14.1Data Low of GLPCI_CAPSUP (0x00E9)

38.20.29.14.2Data High of GLPCI_CAPSUP (0x00EA)

38.20.29.15GLPCI_LINKCAP (0x00EB - 0x00EC)

38.20.29.15.1Data Low of GLPCI_LINKCAP (0x00EB)

38.20.29.15.2Data High of GLPCI_LINKCAP (0x00EC)

38.20.29.16GLPCI_PMSUP (0x00ED - 0x00EE)

38.20.29.16.1Data Low of GLPCI_PMSUP (0x00ED)

38.20.29.16.2Data High of GLPCI_PMSUP (0x00EE)

38.20.29.17GLPCI_REVID (0x00EF - 0x00F0)

38.20.29.17.1Data Low of GLPCI_REVID (0x00EF)

38.20.29.17.2Data High of GLPCI_REVID (0x00F0)

38.20.29.18GLPCI_VFSUP (0x00F1 - 0x00F2)

38.20.29.18.1Data Low of GLPCI_VFSUP (0x00F1)

38.20.29.18.2Data High of GLPCI_VFSUP (0x00F2)

38.20.29.19GLPCI_VENDORID (0x00FB - 0x00FE)

38.20.29.19.1Address Low at GLPCI_VENDORID (0x00FB)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLPCI_VENDORID	0x518	
3:0	Type	0x1	

38.20.29.19.2Address High at GLPCI_VENDORID (0x00FC)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLPCI_VENDORID	0x00BE	

38.20.29.19.3Data Low of GLPCI_VENDORID (0x00FD)

38.20.29.19.4Data High of GLPCI_VENDORID (0x00FE)

38.20.29.20GLGEN_PCIFCNCNT (0x00FF - 0x0102)

38.20.29.20.1Address Low at GLGEN_PCIFCNCNT (0x00FF)

Bits	Field Name	NVM Image Value	Description
15:4	Low Address Bits of GLGEN_PCIFCNCNT	0xAB4	
3:0	Type	0x1	

38.20.29.20.2Address High at GLGEN_PCIFCNCNT (0x0100)

Bits	Field Name	NVM Image Value	Description
15:0	High Address Bits of GLGEN_PCIFCNCNT	0x01C0	

38.20.29.20.3Data Low of GLGEN_PCIFCNCNT (0x0101)

38.20.29.20.4Data High of GLGEN_PCIFCNCNT (0x0102)

38.20.29.21LCB Attributes (0x0105)

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x060	
4:3	Skip	00b	
2:0	Reserved	000b	Reserved.

38.20.29.22LCB Data (0x0106)

Raw data module length: 256 words.

Part of a Type 4 structure to load the PCIe LCB unit.

38.20.30 PCIR Registers Auto-load Module Section Summary Table

Word Offset	Description	Page
0x0002	LCB Attributes	2627
0x0003	LCB Data	2627

38.20.30.1 LCB Attributes (0x0002)

Part of a Type 4 structure to load the PCIe LCB unit.

Bits	Field Name	NVM Image Value	Description
15:5	Length	0x060	
4:3	Skip	00b	
2:0	Reserved	000b	Reserved.

38.20.30.2 LCB Data (0x0003)

Raw data module length: 256 words.

Part of a Type 4 structure to load the PCIe LCB unit.

38.20.31 Auto-generated Pointers Module Section Summary Table

Pointers to Type 1/2 words used by EMP and software.

Word Offset	Description	Page
0x7D80	Module Length	2629
0x7D81	Pointer to PFPM_APM Section	2629
0x7D82	Pointer to PFPM_APM Offset	2629
0x7D83	Pointer to PRTPM_GC Section	2629
0x7D84	Pointer to PRTPM_GC Offset	2630
0x7D85	Pointer to GLGEN_STAT Section	2630
0x7D86	Pointer to GLGEN_STAT Offset	2630

Word Offset	Description	Page
0x7D87	Pointer to GLPCI_SERL Section	2630
0x7D88	Pointer to GLPCI_SERL Offset	2630
0x7D89	Pointer to GLPCI_SERH Section	2630
0x7D8A	Pointer to GLPCI_SERH Offset	2630
0x7D8B	Pointer to PRTGL_SAL Section	2631
0x7D8C	Pointer to PRTGL_SAL Offset	2631
0x7D8D	Pointer to PRTGL_SAH Section	2631
0x7D8E	Pointer to PRTGL_SAH Offset	2631
0x7D8F	Pointer to GLPCI_CAPSUP Section	2631
0x7D90	Pointer to GLPCI_CAPSUP Offset	2631
0x7D91	Pointer to PRTDCB_MFLCN Section	2631
0x7D92	Pointer to PRTDCB_MFLCN Offset	2632
0x7D93	Pointer to PRTDCB_FCCFG Section	2632
0x7D94	Pointer to PRTDCB_FCCFG Offset	2632
0x7D95	Pointer to PFGEN_PORTNUM Section	2632
0x7D96	Pointer to PFGEN_PORTNUM Offset	2632
0x7D97	Pointer to PFPCI_FUNC2 Section	2632
0x7D98	Pointer to PFPCI_FUNC2 Offset	2632
0x7D99	Pointer to PFPCI_CLASS Section	2633
0x7D9A	Pointer to PFPCI_CLASS Offset	2633
0x7D9B	Pointer to PF_VT_PFALLOC_PCIE Section	2633
0x7D9C	Pointer to PF_VT_PFALLOC_PCIE Offset	2633
0x7D9D	Pointer to PF_VT_PFALLOC Section	2633
0x7D9E	Pointer to PF_VT_PFALLOC Offset	2633
0x7D9F	Pointer to GLGEN_PCIFCNCNT Section	2633
0x7DA0	Pointer to GLGEN_PCIFCNCNT Offset	2634
0x7DA1	Pointer to GLPCI_REVID Section	2634
0x7DA2	Pointer to GLPCI_REVID Offset	2634
0x7DA3	Pointer to PFPCI_DEVID Section	2634
0x7DA4	Pointer to PFPCI_DEVID Offset	2634
0x7DA5	Pointer to GLPCI_SUBVENID Section	2634
0x7DA6	Pointer to GLPCI_SUBVENID Offset	2634
0x7DA7	Pointer to PFPCI_SUBSYSID Section	2635
0x7DA8	Pointer to PFPCI_SUBSYSID Offset	2635
0x7DA9	Pointer to GLPCI_VENDORID Section	2635
0x7DAA	Pointer to GLPCI_VENDORID Offset	2635
0x7DAB	Pointer to GLPCI_CNF2 Section	2635
0x7DAC	Pointer to GLPCI_CNF2 Offset	2635
0x7DAD	Pointer to PFPCI_FUNC Section	2635
0x7DAE	Pointer to PFPCI_FUNC Offset	2636
0x7DAF	Pointer to PFPCI_CNF Section	2638
0x7DB0	Pointer to PFPCI_CNF Offset	2638
0x7DB1	Pointer to GLPCI_CAPCTRL Section	2638

Word Offset	Description	Page
0x7DB2	Pointer to GLPCI_CAPCTRL Offset	2639
0x7DB3	Pointer to GLGEN_GPIO_CTL Section	2639
0x7DB4	Pointer to GLGEN_GPIO_CTL Offset	2639
0x7DB5	Pointer to GLGEN_LED_CTL Section	2639
0x7DB6	Pointer to GLGEN_LED_CTL Offset	2639
0x7DB7	Pointer to PFGEN_PORTNUM_CAR Section	2639
0x7DB8	Pointer to PFGEN_PORTNUM_CAR Offset	2640
0x7DB9	Pointer to PFGEN_PORTMDIO_NUM Section	2640
0x7DBA	Pointer to PFGEN_PORTMDIO_NUM Offset	2640
0x7DBB	Pointer to PRTTSYN_CTL0 Section	2640
0x7DBC	Pointer to PRTTSYN_CTL0 Offset	2640
0x7DBD	Reserved	
0x7DBE	Reserved	
0x7DBF	Pointer to PRTGEN_CNF Section	2640
0x7DC0	Pointer to PRTGEN_CNF Offset	2641

38.20.31.1 Module Length (0x7D80)

Bits	Field Name	NVM Image Value	Description
15:0	Module Length		Length in: 2 Bytes unit - 1. First Section -> Word: Auto Generated Pointers Module -> Module Length Last Section -> Word: Auto Generated Pointers Module -> Pointer to PRTGEN_CNF offset

38.20.31.2 Pointer to PFPM_APM Section (0x7D81)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to PFPM_APM Section	0x0000	

38.20.31.3 Pointer to PFPM_APM Offset (0x7D82)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to PFPM_APM Offset	0x0089	

38.20.31.4 Pointer to PRTPM_GC Section (0x7D83)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to PRTPM_GC Section		

38.20.31.5 Pointer to PRTPM_GC Offset (0x7D84)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to PRTPM_GC Offset	0x00B7	

38.20.31.6 Pointer to GLGEN_STAT Section (0x7D85)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to GLGEN_STAT Section	0x0000	

38.20.31.7 Pointer to GLGEN_STAT Offset (0x7D86)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to GLGEN_STAT Offset	0x0002	

38.20.31.8 Pointer to GLPCI_SERL Section (0x7D87)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to GLPCI_SERL Section	0x0000	

38.20.31.9 Pointer to GLPCI_SERL Offset (0x7D88)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to GLPCI_SERL Offset	0x00DF	

38.20.31.10Pointer to GLPCI_SERH Section (0x7D89)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to GLPCI_SERH Section		

38.20.31.11Pointer to GLPCI_SERH Offset (0x7D8A)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to GLPCI_SERH Offset	0x00E1	

38.20.31.12Pointer to PRTGL_SAL Section (0x7D8B)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to PRTGL_SAL Section	0x0000	

38.20.31.13Pointer to PRTGL_SAL Offset (0x7D8C)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to PRTGL_SAL Offset	0x0057	

38.20.31.14Pointer to PRTGL_SAH Section (0x7D8D)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to PRTGL_SAH Section		

38.20.31.15Pointer to PRTGL_SAH Offset (0x7D8E)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to PRTGL_SAH Offset	0x0062	

38.20.31.16Pointer to GLPCI_CAPSUP Section (0x7D8F)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to GLPCI_CAPSUP Section		

38.20.31.17Pointer to GLPCI_CAPSUP Offset (0x7D90)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to GLPCI_CAPSUP Offset	0x00E8	

38.20.31.18Pointer to PRTDCB_MFLCN Section (0x7D91)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to PRTDCB_MFLCN Section		

38.20.31.19Pointer to PRTDCB_MFLCN Offset (0x7D92)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to PRTDCB_MFLCN Offset	0x006D	

38.20.31.20Pointer to PRTDCB_FCCFG Section (0x7D93)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to PRTDCB_FCCFG Section		

38.20.31.21Pointer to PRTDCB_FCCFG Offset (0x7D94)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to PRTDCB_FCCFG Offset	0x01D7	

38.20.31.22Pointer to PFGEN_PORTNUM Section (0x7D95)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to PFGEN_PORTNUM Section	0x0000	

38.20.31.23Pointer to PFGEN_PORTNUM Offset (0x7D96)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to PFGEN_PORTNUM Offset	0x0295	

38.20.31.24Pointer to PFPCI_FUNC2 Section (0x7D97)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to PFPCI_FUNC2 Section		

38.20.31.25Pointer to PFPCI_FUNC2 Offset (0x7D98)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to PFPCI_FUNC2 Offset	0x006C	

38.20.31.26Pointer to PFPCI_CLASS Section (0x7D99)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to PFPCI_CLASS Section		

38.20.31.27Pointer to PFPCI_CLASS Offset (0x7D9A)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to PFPCI_CLASS Offset	0x00B2	

38.20.31.28Pointer to PF_VT_PFALLOC_PCIE Section (0x7D9B)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to PF_VT_PFALLOC_PCIE Section		

38.20.31.29Pointer to PF_VT_PFALLOC_PCIE Offset (0x7D9C)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to PF_VT_PFALLOC_PCIE Offset	0x008F	

38.20.31.30Pointer to PF_VT_PFALLOC Section (0x7D9D)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to PF_VT_PFALLOC Section		

38.20.31.31Pointer to PF_VT_PFALLOC Offset (0x7D9E)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to PF_VT_PFALLOC Offset	0x02B8	

38.20.31.32Pointer to GLGEN_PCIFCNCNT Section (0x7D9F)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to GLGEN_PCIFCNCNT Section		

38.20.31.33Pointer to GLGEN_PCIFCNCNT Offset (0x7DA0)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to GLGEN_PCIFCNCNT Offset	0x0100	

38.20.31.34Pointer to GLPCI_REVID Section (0x7DA1)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to GLPCI_REVID Section		

38.20.31.35Pointer to GLPCI_REVID Offset (0x7DA2)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to GLPCI_REVID Offset	0x00EE	

38.20.31.36Pointer to PFPCI_DEVID Section (0x7DA3)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to PFPCI_DEVID Section		

38.20.31.37Pointer to PFPCI_DEVID Offset (0x7DA4)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to PFPCI_DEVID Offset	0x0026	

38.20.31.38Pointer to GLPCI_SUBVENID Section (0x7DA5)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to GLPCI_SUBVENID Section		

38.20.31.39Pointer to GLPCI_SUBVENID Offset (0x7DA6)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to GLPCI_SUBVENID Offset	0x00D9	

38.20.31.40Pointer to PFPCI_SUBSYSID Section (0x7DA7)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to PFPCI_SUBSYSID Section		

38.20.31.41Pointer to PFPCI_SUBSYSID Offset (0x7DA8)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to PFPCI_SUBSYSID Offset	0x0049	

38.20.31.42Pointer to GLPCI_VENDORID Section (0x7DA9)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to GLPCI_VENDORID Section		

38.20.31.43Pointer to GLPCI_VENDORID Offset (0x7DAA)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to GLPCI_VENDORID Offset	0x00FC	

38.20.31.44Pointer to GLPCI_CNF2 Section (0x7DAB)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to GLPCI_CNF2 Section		

38.20.31.45Pointer to GLPCI_CNF2 Offset (0x7DAC)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to GLPCI_CNF2 Offset	0x00DD	

38.20.31.46Pointer to PFPCI_FUNC Section (0x7DAD)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to PFPCI_FUNC Section		

38.20.31.47Pointer to PFPCI_FUNC Offset (0x7DAE)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to PFPCI_FUNC Offset	0x00E3	

38.20.31.48Pointer to GLGEN_PCIFCNCNT Section (0x7D9F)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to GLGEN_PCIFCNCNT section		

38.20.31.49Pointer to GLGEN_PCIFCNCNT Offset (0x7DA0)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to GLGEN_PCIFCNCNT offset	0x100	

38.20.31.50Pointer to GLPCI_REVID Section (0x7DA1)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to GLPCI_REVID section		

38.20.31.51Pointer to GLPCI_REVID Offset (0x7DA2)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to GLPCI_REVID offset	0xEE	

38.20.31.52Pointer to PFPCI_DEVID Section (0x7DA3)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to PFPCI_DEVID section		

38.20.31.53Pointer to PFPCI_DEVID Offset (0x7DA4)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to PFPCI_DEVID offset	0x26	

38.20.31.54Pointer to GLPCI_SUBVENID Section (0x7DA5)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to GLPCI_SUBVENID section		

38.20.31.55Pointer to GLPCI_SUBVENID Offset (0x7DA6)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to GLPCI_SUBVENID offset	0xD9	

38.20.31.56Pointer to PFPCI_SUBSYSID Section (0x7DA7)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to PFPCI_SUBSYSID section		

38.20.31.57Pointer to PFPCI_SUBSYSID Offset (0x7DA8)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to PFPCI_SUBSYSID offset	0x49	

38.20.31.58Pointer to GLPCI_VENDORID Section (0x7DA9)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to GLPCI_VENDORID section		

38.20.31.59Pointer to GLPCI_VENDORID Offset (0x7DAA)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to GLPCI_VENDORID offset	0xFC	

38.20.31.60Pointer to GLPCI_CNF2 Section (0x7DAB)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to GLPCI_CNF2 section		

38.20.31.61Pointer to GLPCI_CNF2 Offset (0x7DAC)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to GLPCI_CNF2 offset	0xDD	

38.20.31.62Pointer to PFPCI_FUNC Section (0x7DAD)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to PFPCI_FUNC section		

38.20.31.63Pointer to PFPCI_FUNC Offset (0x7DAE)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to PFPCI_FUNC offset	0xE0	

38.20.31.64Pointer to PFPCI_CNF Section (0x7DAF)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to PFPCI_CNF section		

38.20.31.65Pointer to PFPCI_CNF Offset (0x7DB0)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to PFPCI_CNF offset	0x3	

38.20.31.66Pointer to GLPCI_CAPCTRL Section (0x7DB1)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to GLPCI_CAPCTRL section		

38.20.31.67Pointer to GLPCI_CAPCTRL Offset (0x7DB2)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to GLPCI_CAPCTRL offset	0xE6	

38.20.31.68Pointer to GLGEN_GPIO_CTL Section (0x7DB3)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to GLGEN_GPIO_CTL section		

38.20.31.69Pointer to GLGEN_GPIO_CTL Offset (0x7DB4)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to GLGEN_GPIO_CTL offset	0x3	

38.20.31.70Pointer to GLGEN_LED_CTL Section (0x7DB5)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to GLGEN_LED_CTL section		

38.20.31.71Pointer to GLGEN_LED_CTL Offset (0x7DB6)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to GLGEN_LED_CTL offset	0x3F	

38.20.31.72Pointer to PFGEN_PORTNUM_CAR Section (0x7DB7)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to PFGEN_PORTNUM_CAR section		

38.20.31.73Pointer to PFGEN_PORTNUM_CAR Offset (0x7DB8)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to PFGEN_PORTNUM_CAR offset	0x66	

38.20.31.74Pointer to PFGEN_PORTMDIO_NUM Section (0x7DB9)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to PFGEN_PORTMDIO_NUM section		

38.20.31.75Pointer to PFGEN_PORTMDIO_NUM Offset (0x7DBA)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to PFGEN_PORTMDIO_NUM offset	0x3	

38.20.31.76Pointer to PRTTSYN_CTL0 Section (0x7DBB)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to PRTTSYN_CTL0 section		

38.20.31.77Pointer to PRTTSYN_CTL0 Offset (0x7DBC)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to PRTTSYN_CTL0 offset	0x2F	

38.20.31.78Pointer to PRTGEN_CNF Section (0x7DBF)

Bits	Field Name	NVM Image Value	Description
15:0	Cloned pointer to PRTGEN_CNF section		

38.20.31.79 Pointer to PRTGEN_CNF Offset (0x7DC0)

Bits	Field Name	NVM Image Value	Description
15:0	Pointer to PRTGEN_CNF Offset	0xAC	

38.20.32 EMP Global Module Section Summary Table

This section contains two sub-sections:

1. Vendor specific settings: Settings for external PHY or modules. This part has a proprietary format to enable advanced, vendor-specific PHY settings. These settings are loaded into the external devices via the MDIO/I^2C interface.
2. List of qualified modules: Parameter list of up to 16 modules. Per-module list holds OUI, revision, and version numbers.

Word Offset	Description	Page
0x13000	Section Length	2641
0x13001	Number of Qualified Modules	2642
0x13002 + 12*n, n=0...15	Module OUI Bytes 0-1	2642
0x13003 + 12*n, n=0...15	Module OUI Byte 2	2642
0x13004 + 12*n, n=0...15	Vendor Part Number Bytes 0-1	2642
0x13005 + 12*n, n=0...15	Vendor Part Number Bytes 2-3	2643
0x13006 + 12*n, n=0...15	Vendor Part Number Bytes 4-5	2643
0x13007 + 12*n, n=0...15	Vendor Part Number Bytes 6-7	2643
0x13008 + 12*n, n=0...15	Vendor Part Number Bytes 8-9	2643
0x13009 + 12*n, n=0...15	Vendor Part Number Bytes 10-11	2644
0x1300A + 12*n, n=0...15	Vendor Part Number Bytes 12-13	2644
0x1300B + 12*n, n=0...15	Vendor Part Number Bytes 14-15	2644
0x1300C + 12*n, n=0...15	Module Revision Number Bytes 0-1	2644
0x1300D + 12*n, n=0...15	Module Revision Number Bytes 2-3	2645
0x130C2	LESM Global Configurations	2645
0x130C3	LESM AN73 + PD State Configurations	2645
0x130C4	LESM 10G-KX4 State Configurations	2645
0x130C5	LESM 1G-KX State Configurations	2646
0x130C6	CRC8	2646

38.20.32.1 Section Length (0x13000)

The length of the section in words. Note that section length does not include a count for the section length word.

Bits	Field Name	NVM Image Value	Description
15:0	Section Length		Length in words of the section covered by CRC

38.20.32.2 Number of Qualified Modules (0x13001)

Number of valid entries (0 through 15) in the qualified modules' list.

Bits	Field Name	NVM Image Value	Description
15:0	Number of Qualified Modules	0x0010	

38.20.32.3 Module OUI Bytes 0-1[n] (0x13002 + 12*n, n=0...15)

OUI of qualified external modules.

SFP+: Address 0xA0, Byte 3-5

QSFP+: Address 133:131 page 0

Bits	Field Name	NVM Image Value	Description
15:8	Byte 1	0x00	
7:0	Byte 0	0x00	

38.20.32.4 Module OUI Byte 2[n] (0x13003 + 12*n, n=0...15)

OUI of qualified external modules.

SFP+: Address 0xA0, Byte 3-5

QSFP+: Address 133:131 page 0

Bits	Field Name	NVM Image Value	Description
15:8	Reserved	0x00	Reserved.
7:0	Byte 0	0x00	

38.20.32.5 Vendor Part Number Bytes 0-1[n] (0x13004 + 12*n, n=0...15)

Vendor Part Number of qualified external modules.

SFP+: Address 0xA0 Bytes 55:40

QSFP+: Address 183:168 page 0

Bits	Field Name	NVM Image Value	Description
15:8	Byte1	0x00	
7:0	Byte0	0x00	

38.20.32.6 Vendor Part Number Bytes 2-3[n] (0x13005 + 12*n, n=0...15)

Vendor Part Number of qualified external modules.

SFP+: Address 0xA0 Bytes 55:40

QSFP+: Address 183:168 page 0

Bits	Field Name	NVM Image Value	Description
15:8	Byte1	0x00	
7:0	Byte0	0x00	

38.20.32.7 Vendor Part Number Bytes 4-5[n] (0x13006 + 12*n, n=0...15)

Vendor Part Number of qualified external modules.

SFP+: Address 0xA0 Bytes 55:40

QSFP+: Address 183:168 page 0

Bits	Field Name	NVM Image Value	Description
15:8	Byte1	0x00	
7:0	Byte0	0x00	

38.20.32.8 Vendor Part Number Bytes 6-7[n] (0x13007 + 12*n, n=0...15)

Vendor Part Number of qualified external modules.

SFP+: Address 0xA0 Bytes 55:40

QSFP+: Address 183:168 page 0

Bits	Field Name	NVM Image Value	Description
15:8	Byte1	0x00	
7:0	Byte0	0x00	

38.20.32.9 Vendor Part Number Bytes 8-9[n] (0x13008 + 12*n, n=0...15)

Vendor Part Number of qualified external modules.

SFP+: Address 0xA0 Bytes 55:40

QSFP+: Address 183:168 page 0

Bits	Field Name	NVM Image Value	Description
15:8	Byte1	0x00	
7:0	Byte0	0x00	

38.20.32.10Vendor Part Number Bytes 10-11[n] (0x13009 + 12*n, n=0...15)

Vendor Part Number of qualified external modules.

SFP+: Address 0xA0 Bytes 55:40

QSFP+: Address 183:168 page 0

Bits	Field Name	NVM Image Value	Description
15:8	Byte1	0x00	
7:0	Byte0	0x00	

38.20.32.11Vendor Part Number Bytes 12-13[n] (0x1300A + 12*n, n=0...15)

Vendor Part Number of qualified external modules.

SFP+: Address 0xA0 Bytes 55:40

QSFP+: Address 183:168 page 0

Bits	Field Name	NVM Image Value	Description
15:8	Byte1	0x00	
7:0	Byte0	0x00	

38.20.32.12Vendor Part Number Bytes 14-15[n] (0x1300B + 12*n, n=0...15)

Vendor Part Number of qualified external modules.

SFP+: Address 0xA0 Bytes 55:40

QSFP+: Address 183:168 page 0

Bits	Field Name	NVM Image Value	Description
15:8	Byte1	0x00	
7:0	Byte0	0x00	

38.20.32.13Module Revision Number Bytes 0-1[n] (0x1300C + 12*n, n=0...15)

Revision Number of qualified external modules.

SFP+: Address 0xA0 Bytes 59:56

QSFP+: Address 185:184 page 0

Bits	Field Name	NVM Image Value	Description
15:8	Byte 1	0x00	
7:0	Byte 0	0x00	

38.20.32.14Module Revision Number Bytes 2-3[n] (0x1300D + 12*n, n=0...15)

Revision Number of qualified external modules.

SFP+: Address 0xA0 Bytes 59:56

QSFP+: Address 185:184 page 0

Bits	Field Name	NVM Image Value	Description
15:8	Byte 1	0x00	
7:0	Byte 0	0x00	

38.20.32.15LESM Global Configurations (0x130C2)

Bits	Field Name	NVM Image Value	Description
15:6	Reserved	0x0	
5:1	10G Only Counter	0x3	Number of attempts to make at 10 GbE before attempting to establish the link at 1 GbE.
0	LESM Enable	0x0	1 = LESM is enabled. 0 = LESM is disabled. 0x0 = Disabled. 0x1 = Enabled.

38.20.32.16LESM AN73 + PD State Configurations (0x130C3)

Bits	Field Name	NVM Image Value	Description
15:1	Timeout	0x32	Number of seconds to stay in this state before moving on to the next. Granularity is in 0.1 second.
0	State Enable	0x1	1 = State is enabled ***Note:*** This state must always be enabled when the LESM feature is enabled. 0x0 = Disabled. 0x1 = Enabled.

38.20.32.17LESM 10G-KX4 State Configurations (0x130C4)

Bits	Field Name	NVM Image Value	Description
15:1	Timeout	0x32	Number of seconds to stay in this state before moving on to the next. Granularity is in 0.1 second.
0	State Enable	0x1	1 = State is enabled. 0 = State is disabled. 0x0 = Disabled. 0x1 = Enabled.

38.20.32.18LESM 1G-KX State Configurations (0x130C5)

Bits	Field Name	NVM Image Value	Description
15:1	Timeout	0x32	Number of seconds to stay in this state before moving on to the next. Granularity is in 0.1 second.
0	State Enable	0x1	1 = State is enabled. ***Note:*** This state must always be enabled when the LESM feature is enabled. 0x0 = Disabled. 0x1 = Enabled.

38.20.32.19CRC8 (0x130C6)

Bits	Field Name	NVM Image Value	Description
15	CRC Field Used	1b	CRC Field Used. 0 = CRC not used. 1 = CRC used.
14:8	Reserved	0x00	Reserved.
7:0	CRC8		CRC-8-CCITT: Start Section -> Word: EMP Global Module -> Section Length End Section -> Word: EMP Global Module -> LESM 1G-KX State Configurations

38.20.33 Manageability Module Header Section Summary Table

This section contains parameters related to the manageability functionality such as connection type and others. It also points to subsections configuring the filters and the sideband interfaces.

Word Offset	Description	Page
0x14000	Section Length	2647
0x14001	Common Manageability Parameters	2647
0x14002	Common Manageability Parameters 2	2647
0x14003	Pass Through LAN 0 Configuration Pointer	2648
0x14004	Pass Through LAN 1 Configuration Pointer	2648
0x14005	Pass Through LAN 2 Configuration Pointer	2648
0x14006	Pass Through LAN 3 Configuration Pointer	2648
0x14007	Sideband Configuration Pointer	2648
0x14008	Flexible TCO Filter Configuration Pointer	2649
0x14009	Traffic Types Parameters	2649
0x1400A	OEM Structure Pointer	2650
0x1400B	CRC8	2650

38.20.33.1 Section Length (0x14000)

The length of the section in words. Note that section length does not include a count for the section length word.

Bits	Field Name	NVM Image Value	Description
15:0	Section Length		Length in words of the section covered by CRC.

38.20.33.2 Common manageability parameters (0x14001)

Bits	Field Name	NVM Image Value	Description
15:11	Reserved	0x00	Reserved.
10:8	Manageability Mode	010b	Manageability Mode. 000b = None 010b = Pass through (PT) mode
7:5	Reserved	000b	Reserved.
4	Force TCO Reset Disable	1b	If cleared, allows the BMC to do a Global reset of the device using the Force TCO SMBus or NC-SI commands 0 = Enable Force TCO reset. 1 = Disable Force TCO reset.
3	Reserved	0b	Reserved.
2	OS2BMC Capable	1b	OS2BMC Capable. 0 = Disabled 1 = Enabled
1:0	Reserved	00b	Reserved.

38.20.33.3 Common Manageability Parameters 2 (0x14002)

Bits	Field Name	NVM Image Value	Description
15:12	Reserved	0x0	Reserved.
11	Multi-Drop NC-SI	1b	Multi-Drop NC-SI. 0 = Point-to-point 1 = Multi-drop
10	Reserved	0b	Reserved.
9	EMP_LINK_ON	0b	Copy of the PRTPM_GC.EMP_LINK_ON bit for EMP use. 0 = Disable 1 = Enable
8:4	Reserved	0x00	Reserved.
3	LAN3_FTCO_ISOL_DIS	1b	Allow isolation of port 3 per BMC request. 0 = Enabled 1 = Disabled
2	LAN2_FTCO_ISOL_DIS	1b	Allow isolation of port 2 per BMC request. 0 = Enabled 1 = Disabled
1	LAN1_FTCO_ISOL_DIS	1b	Allow isolation of port 1 per BMC request. 0 = Enabled 1 = Disabled

Bits	Field Name	NVM Image Value	Description
0	LAN0_FTCO_ISOL_DIS	1b	Allow isolation of port 0 per BMC request. 0 = Enabled 1 = Disabled

38.20.33.4 Pass Through LAN 0 Configuration Pointer (0x14003)

Bits	Field Name	NVM Image Value	Description
15:0	Pass Through LAN 0 Configuration Pointer	0xFFFF	Points to the Pass Through Control Words Structure 0 section. For more detail on the Pass Through Control Words Structure 0 inner structure, see Section 38.21.

38.20.33.5 Pass Through LAN 1 Configuration Pointer (0x14004)

Bits	Field Name	NVM Image Value	Description
15:0	Pass Through LAN 1 Configuration Pointer	0xFFFF	Points to the Pass Through Control Words Structure 1 section. For more detail on the Pass Through Control Words Structure 1 inner structure, see Section 38.21.

38.20.33.6 Pass Through LAN 2 Configuration Pointer (0x14005)

Bits	Field Name	NVM Image Value	Description
15:0	Pass Through LAN 2 Configuration Pointer	0xFFFF	Points to the Pass Through Control Words Structure 2 section. For more detail on the Pass Through Control Words Structure 2 inner structure, see Section 38.21.

38.20.33.7 Pass Through LAN 3 Configuration Pointer (0x14006)

Bits	Field Name	NVM Image Value	Description
15:0	Pass Through LAN 3 Configuration Pointer	0xFFFF	Points to the Pass Through Control Words Structure 3 section. For more detail on the Pass Through Control Words Structure 3 inner structure, see Section 38.21.

38.20.33.8 Sideband Configuration Pointer (0x14007)

This module is 28 bytes long and must be mapped in the first valid 4 KB sector of the Flash.

Bits	Field Name	NVM Image Value	Description
15:0	Sideband Configuration Pointer	0xFFFF	Points to the Sideband Configuration Structure section. For more detail on the Sideband Configuration Structure inner structure, see Section 38.20.35.

38.20.33.9 Flexible TCO Filter Configuration Pointer (0x14008)

This section loads all of the flexible filters, The control + mask + filter data are repeatable as the number of filters. Section length in offset 0 is for all filters.

Bits	Field Name	NVM Image Value	Description
15:0	Flexible TCO Filter Configuration Pointer	0xFFFF	Points to the Flexible TCO Filter Configuration Structure section.

38.20.33.10Traffic Types Parameters (0x14009)

Bits	Field Name	NVM Image Value	Description
15:14	Reserved	00b	Reserved.
13:12	Port 3 traffic types	01b	Defines which type of traffic can flow to and from primary BMC connection on port 3. The traffic types defined by this field are enabled by the Manageability Mode field and the OS2BMC Capable bit in the Common Manageability Parameters 1 NVM word (see Section 38.20.33.2). 00b = Reserved. 01b = Network to BMC traffic only. 10b = OS2BMC traffic only. 11b = Both Network to BMC traffic and OS2BMC traffic.
11:10	Reserved	00b	Reserved.
9:8	Port 2 traffic types	01b	Defines which type of traffic can flow to and from primary BMC connection on port 2. The traffic types defined by this field are enabled by the Manageability Mode field and the OS2BMC Capable bit in the Common Manageability Parameters 1 NVM word (see Section 38.20.33.2). 00b = Reserved. 01b = Network to BMC traffic only. 10b = OS2BMC traffic only. 11b = Both Network to BMC traffic and OS2BMC traffic.
7:6	Reserved	00b	Reserved.
5:4	Port 1 traffic types	01b	Defines which type of traffic can flow to and from primary BMC connection on port 1 (see Section 38.20.33.2). The traffic types defined by this field are enabled by the Manageability Mode field and the OS2BMC Capable bit in the Common Manageability Parameters 1 NVM word. 00b = Reserved. 01b = Network to BMC traffic only. 10b = OS2BMC traffic only. 11b = Both Network to BMC traffic and OS2BMC traffic.
3:2	Reserved	00b	Reserved.
1:0	Port 0 traffic types	01b	Defines which type of traffic can flow to and from primary BMC connection on port 0. The traffic types defined by this field are enabled by the Manageability Mode field and the OS2BMC Capable bit in the Common Manageability Parameters 1 NVM word (see Section 38.20.33.2). 00b = Reserved. 01b = Network to BMC traffic only. 10b = OS2BMC traffic only. 11b = Both Network to BMC traffic and OS2BMC traffic.

38.20.33.11OEM Structure Pointer (0x1400A)

Bits	Field Name	NVM Image Value	Description
15:0	OEM Structure Pointer	0xFFFF	Points to the OEM section.

38.20.33.12CRC8 (0x1400B)

Bits	Field Name	NVM Image Value	Description
15	CRC Field Used	1b	CRC Field Used. 0 = CRC not used. 1 = CRC used.
14:8	Reserved	0x00	Reserved.
7:0	CRC8		CRC-8-CCITT: Start Section -> Word: Manageability Module Header -> Section Length End Section -> Word: Manageability Module Header -> OEM Structure Pointer

38.20.34 Flexible TCO Filter Configuration Structure Section Summary Table

This section contains the setting of the manageability flex filters for all the ports. Its format is described in the manageability section, in the flexible TCO filter configuration subsection.

Word Offset	Description	Page
0x0000	Section Length	2650
0x0001	Flexible Filter Data	2651
0x0002	CRC8	2651

38.20.34.1 Section Length (0x0000)

The length of the section in words. Note that section length does not include a count for the section length word.

Bits	Field Name	NVM Image Value	Description
15:0	Section Length		Length in: 2 Bytes unit - 1. First Section -> Word: Flexible TCO Filter Configuration Structure -> Section Length Last Section -> Word: Flexible TCO Filter Configuration Structure -> CRC8

38.20.34.2 Flexible Filter Data (0x0001)

Raw data module length: variable.

Flexible Filter Data.

38.20.34.3 CRC8 (0x0002)

Bits	Field Name	NVM Image Value	Description
15	CRC Field Used	1b	CRC Field Used 0 = CRC not used. 1 = CRC used.
14:8	Reserved	0x00	Reserved.
7:0	CRC8		CRC-8-CCITT: Start Section -> Word: Flexible TCO Filter Configuration Structure -> Section Length End Section -> Word: Flexible TCO Filter Configuration Structure -> Flexible Filter Data

38.20.35 Sideband Configuration Structure Section Summary Table

This section describes the setting of the different pass through interfaces (SMBus, NC-SI and MCTP).

Word Offset	Description	Page
0x0000	Section Length	2652
0x0001	SMBus Maximum Fragment Size	2652
0x0002	SMBus Notification Timeout and Flags	2652
0x0003	NC-SI Configuration 1	2653
0x0004	NC-SI Configuration 2	2653
0x0005	NC-SI Flow Control XOFF	2653
0x0006	NC-SI Flow Control XON	2654
0x0007	NC-SI HW Arbitration Configuration	2654
0x0008 - 0x000C	Reserved	
0x000D	OEM IANA	2654
0x000E	NC-SI over MCTP Message Types	2654
0x000F	NC-SI over MCTP Configuration	2654
0x0010	MCTP Rate Limiter Config 1	2655
0x0011	MCTP Rate Limiter Config 2	2655
0x0012	NC-SI Channel to Port Mapping	2655
0x0013	CRC8	2656

38.20.35.1 Section Length (0x0000)

The length of the section in words. Note that section length does not include a count for the section length word.

Bits	Field Name	NVM Image Value	Description
15:0	Section Length		Length in words of the section covered by CRC.

38.20.35.2 SMBus Maximum Fragment Size (0x0001)

Bits	Field Name	NVM Image Value	Description
15:0	Fragment size	0x0020	SMBus Maximum Fragment Size (bytes) Supported range is between 32 and 240 bytes. ***Note:*** In MCTP mode, this value should be set to 0x45 (64 bytes payload + 5 bytes of MCTP header).

38.20.35.3 SMBus Notification Timeout and Flags (0x0002)

Bits	Field Name	NVM Image Value	Description
15:8	SMBus Notification Timeout (ms)	0xFF	
7:6	SMBus Connection Speed	00b	SMBus Connection Speed 00b = Standard SMBus connection. 01b = 400 Kb/s fast I^2C. 10b = 1 Mb/s fast+ I^2C. 11b = Reserved.
5	SMBus Block Read command	0b	SMBus Block Read Command 0 = Block read command is 0xC0. 1 = Block read command is 0xD0.
4	Reserved	0b	Reserved.
3	Enable Fairness Arbitration	1b	MCTP Over SMBus Feature 0 = Disabled fairness arbitration. 1 = Enabled fairness arbitration.
2	Disable SMBus ARP Functionality	0b	Disable SMBus ARP Functionality 0 = SMBus ARP enabled. 1 = SMBus ARP disabled.
1	SMBus ARP PEC	1b	SMBus ARP PEC Should be set in MCTP modes. 0 = Disable SMBus ARP PEC. 1 = Enable SMBus ARP PEC.
0	SMBus Transaction PEC	0b	SMBus Transactions PEC Should be set in MCTP modes. 0 = Disable PEC — PEC is not added to master write or slave read transactions. A slave write transaction with PEC is dropped. 1 = Enable PEC — PEC is added for master SMBus write transactions. A PEC is added to slave read transactions and can be received in slave write transactions.

38.20.35.4 NC-SI Configuration 1 (0x0003)

Bits	Field Name	NVM Image Value	Description
15	Reserved	0b	Reserved.
14	Flow Control	0b	Flow Control 0 = NC-SI flow control disable. 1 = NC-SI flow control enable.
13:10	Reserved	0x0	Reserved.
9	NC-SI HW Arbitration Enable	0b	NC-SI Hardware Arbitration Enable 0 = Not supported. 1 = Supported.
8	Reserved	1b	Reserved.
7:5	Package ID	000b	Meaningful only when bit 15 of NC-SI Configuration 2 word (offset 0x07) is cleared.
4:0	Reserved	0x00	Reserved. Must be zero

38.20.35.5 NC-SI Configuration 2 (0x0004)

Bits	Field Name	NVM Image Value	Description
15	Read NC-SI Package ID	0b	Read NC-SI Package ID from: 0 = NVM, NC-SI Configuration 1 word bits 7:5 (offset 0x06). 1 = Reserved
14:10	Reserved	0x00	Reserved.
9:4	Reserved	0x01	Reserved.
3:0	Max XOFF Renewal	0x3	NC-SI Flow Control MAX XOFF Renewal (# of XOFF renewals allowed) 0x0 = Disabled. Unlimited number of XOFF frames can be sent. 0x1 = Up to 2 consecutive XOFFs frames can be sent by the device. 0x2 = Up to 3 consecutive XOFFs frames can be sent by the device. . . . 0xF = Up to 16 consecutive XOFFs frames can be sent by the device.

38.20.35.6 NC-SI Flow Control XOFF (0x0005)

Bits	Field Name	NVM Image Value	Description
15:0	XOFF Threshold	0x12B8	Tx buffer watermark for sending a XOFF NC-SI flow control packet in bytes. The XOFF Threshold value refers to the occupied space in the buffer. ***Notes:*** 1. Field relevant for NC-SI operation mode only. 2. To support a maximum packet size of 1.5 KB, the value programmed assuming a Tx buffer size of 8 KB value of field should be 0x12B8 (4,792 bytes).

38.20.35.7 NC-SI Flow Control XON (0x0006)

Bits	Field Name	NVM Image Value	Description
15:0	XON Threshold	0x1B48	Tx buffer water mark for sending a XON NC-SI flow control packet in bytes. The XON Threshold value refers to the available space in the TX buffer. ***Notes:*** 1. Field relevant for NC-SI operation mode only. 2. To support maximum packet size of 1.5 KB, the value programmed should be a positive value that equals: XOFF Threshold (refer to Section 6.5.5.7) + 1536 bytes. Assuming a TX Buffer size is 8 KB and the XOFF Threshold is 2,744 bytes value of field should be 0x1B48 (6,985 bytes).

38.20.35.8 NC-SI HW Arbitration Configuration (0x0007)

Bits	Field Name	NVM Image Value	Description
15:0	TOKEN Timeout	0xA000	NC-SI HW-Arbitration TOKEN Timeout (in NC-SI REF_CLK cycles - 20 ns). Setting value to 0 disables the timeout mechanism.

38.20.35.9 OEM IANA (0x000D)

Bits	Field Name	NVM Image Value	Description
15:0	OEM IANA	0x0000	If not zero and not 0x157, the device accepts additional OEM commands with this IANA number. These commands are accepted only if the OEM NVM Structure Pointer is valid. The set of commands accepted depends on the IANA value in this field. The regular Intel OEM commands are accepted only with IANA 0x157.

38.20.35.10NC-SI Over MCTP Message Types (0x000E)

Bits	Field Name	NVM Image Value	Description
15:8	NC-SI Control Message Type	0x02	Defines the MCTP message type used to identify NC-SI control packets.
7:0	NC-SI Pass Through Message Type	0x03	Defines the MCTP message type used to identify NC-SI pass through packets.

38.20.35.11NC-SI Over MCTP Configuration (0x000F)

Bits	Field Name	NVM Image Value	Description
15:7	Reserved	0x000	Reserved.
6	Simplified MCTP	0b	If set, only SOM and EOM bits are used for the reassembly process. Relevant only in SMBus mode
5	Disable ACLs	0b	If set, the ACLs on the PCIe VDMs are disabled.
4:0	Reserved	0x00	Reserved.

38.20.35.12MCTP Rate Limiter Config 1 (0x0010)

MCTP rate limiter configuration for first channel.

Bits	Field Name	NVM Image Value	Description
15:0	MCTP Rate	0x3E80	Defines the number of cycles between accesses of the MCTP send client to the memory arbiter. Current value assumes a clock of 125 MHz and a bus width of 128 bits. This value provides a bit rate of 1 Mb/s.

38.20.35.13MCTP Rate Limiter Config 2 (0x0011)

MCTP rate limiter configuration for first channel.

Bits	Field Name	NVM Image Value	Description
15	Decision Point	0b	Defines if, when credits are available, a full MCTP message is sent or a single VDM is sent. 0 = Per VDM. 1 = Per packet.
14:0	MCTP Max Credits	0x0005	Defines the maximum number of 16-byte credit that can be accumulated. These credits include the VDM header line (one line for each 64-byte VDM).

38.20.35.14NC-SI Channel to Port Mapping (0x0012)

Defines the mapping of NC-SI channels to Physical ports.

Note: The list of channel IDs must starts at zero and be consecutive.

Bits	Field Name	NVM Image Value	Description
15	Table Valid	0b	Table Valid 0 = Table invalid — Use default algorithm described in Channel ID mapping section. 1 = Table valid — Use the mapping defined in this word.
14:13	Port #3 channel ID	00b	
12	Port #3 Channel Enable	1b	Port #3 Channel Enable 0 = Disabled. 1 = Enabled.
11	Reserved	0b	Reserved.
10:9	Port #2 channel ID	00b	
8	Port #2 Channel Enable	1b	Port #2 Channel Enable 0 = Disabled. 1 = Enabled.
7	Reserved	0b	Reserved.
6:5	Port #1 channel ID	00b	
4	Port #1 Channel Enable	1b	Port #1 Channel Enable 0 = Disabled. 1 = Enabled.
3	Reserved	0b	Reserved.
2:1	Port #0 channel ID	00b	

Bits	Field Name	NVM Image Value	Description
0	Port #0 Channel Enable	1b	Port #0 Channel Enable 0 = Disabled. 1 = Enabled.

38.20.35.15CRC8 (0x0013)

Bits	Field Name	NVM Image Value	Description
15	CRC Field Used	1b	CRC Field Used 0 = CRC not used. 1 = CRC used.
14:8	Reserved	0x00	Reserved.
7:0	CRC8		CRC-8-CCITT: Start Section -> Word: Sideband Configuration Structure -> Section Length End Section -> Word: Sideband Configuration Structure -> NC-SI Channel to Port Mapping

38.20.36 OEM Section Summary Table

This section is the header of the OEM-specific data identifying the OEM for which this data is defined.

Word Offset	Description	Page
0x0000	Section Length	2656
0x0001	OEM Header	2656
0x0002 - 0x0005	Reserved	
0x0006	CRC8	2657

38.20.36.1 Section Length (0x0000)

The length of the section in words. Note that section length does not include a count for the section length word.

Bits	Field Name	NVM Image Value	Description
15:0	Section Length		Length in: 2 Bytes unit - 1. First Section -> Word: OEM Section -> Section Length Last Section -> Word: OEM Section -> CRC8

38.20.36.2 OEM Header (0x0001)

Bits	Field Name	NVM Image Value	Description
15:0	OEM Identifier	0x02A2	Identify the OEM for which this section is defined. Should be equal to the OEM IANA value.

38.20.36.3 CRC8 (0x0006)

Bits	Field Name	NVM Image Value	Description
15	CRC Field Used	1b	CRC Field Used. 0 = CRC not used. 1 = CRC used.
14:8	Reserved	0x00	Reserved.
7:0	CRC8		CRC-8-CCITT: Start Section -> Word: OEM Section -> Section Length End Section -> Word: OEM Section -> CEM Flags

38.20.37 EMP Settings Module Header Section Summary Table

This section contains the modes of operation of the EMP.

Word Offset	Description	Page
0x15000	Section Length	2657
0x15001	Common Firmware Parameters	2657
0x15002	Reserved	
0x15003	Features Enable	2658
0x15004	Reserved	
0x15005	Reserved	2658
0x15006	LLDP Configuration Pointer	2658
0x15007 - 0x1500A	Reserved	
0x1500B	LLDP TLVs Pointer	2658
0x1500C	CRC8	2658

38.20.37.1 Section Length (0x15000)

The length of the section in words. Note that section length does not include a count for the section length word.

Bits	Field Name	NVM Image Value	Description
15:0	Section Length		Length in words of the section covered by CRC.

38.20.37.2 Common Firmware Parameters (0x15001)

Bits	Field Name	NVM Image Value	Description
15:10	Reserved	0x5	Reserved.
9	PF reset on queue overflow	0b	PF Reset On Queue Overflow 0 = Disabled. 1 = Enabled.
8:0	Reserved	0x3	Reserved.

38.20.37.3 Features Enable (0x15003)

Describes support for various features.

Bits	Field Name	NVM Image Value	Description
15:12	Reserved	0x0	Reserved.
11:9	Switching Mode	000b	Switching mode in NVM. 000b = EVB switching 802.1Qbg/801.1BR switching. All other values are reserved.
8:1	Reserved	0x0	1 = reserved
4:1	Reserved	0x0	Reserved.
0	EVB Protocols Enabled	1b	If set, filtering according to IEEE 802.1QBg and 802.1BR specs is enabled. 0 = Disabled. 1 = Enabled.

38.20.37.4 LLDP Configuration Pointer (0x15006)

Bits	Field Name	NVM Image Value	Description
15:0	LLDP configuration pointer	0xFFFF	Points to the LLDP Configuration section.

38.20.37.5 LLDP TLVs Pointer (0x1500B)

Pointer to a set of fixed-structure TLVs used by the EMP.

Bits	Field Name	NVM Image Value	Description
15:0	LLDP TLVs Pointer	0xFFFF	Pointer to a set of fixed-structure TLVs used by the EMP. Points to the LLDP OEM TLVs section.

38.20.37.6 CRC8 (0x1500C)

Bits	Field Name	NVM Image Value	Description
15	CRC Field Used	1b	CRC Field Used 0 = CRC not used. 1 = CRC used.
14:8	Reserved	0x00	Reserved.
7:0	CRC8		CRC-8-CCITT: Start Section -> Word: EMP Settings Module Header -> Section Length End Section -> Word: EMP Settings Module Header -> LLDP TLVs pointer

38.20.38 LLDP Configuration Section Summary Table

Default settings to the embedded LLDP agent.

Word Offset	Description	Page
0x0000	Section Length	2659
0x0001	LLDP Admin Status (Factory)	2659
0x0002	msgFastTx	2660
0x0003	msgTxInterval	2660
0x0004	LLDP Tx Parameters	2660
0x0005	LLDP Initialization Timers	2660
0x0006	ENDLESS_XOFF_THRESH	2660
0x0007	DCBX Mode	2661
0x0008	Current Factory LLDP Admin Status	2661
0x0009	CRC8	2661

38.20.38.1 Section Length (0x0000)

The length of the section in words. Note that section length does not include a count for the section length word.

Bits	Field Name	NVM Image Value	Description
15:0	Section Length		Length in words of the section covered by CRC.

38.20.38.2 LLDP Admin Status-Factory (0x0001)

Defines status of LLDP agent. Each LAN port has independent status.

3: Both receive and transmit enabled.
2: LLDP is configured for transmits only.
1: LLDP is configured for receives only.
0: LLDP agent is disabled.

Bits	Field Name	NVM Image Value	Description
15:12	Port 3	0x3	Defines status of LLDP agent. Applies to LAN Port 3.
11:8	Port 2	0x3	Defines status of LLDP agent. Applies to LAN Port 2.
7:4	Port 1	0x3	Defines status of LLDP agent. Applies to LAN Port 1.
3:0	Port 0	0x3	Defines status of LLDP agent. Applies to LAN Port 0.

38.20.38.3 msgFastTx (0x0002)

Bits	Field Name	NVM Image Value	Description
15:0	msgFastTx	0x0001	Time interval in timer ticks (seconds) between PDU transmits during fast transmits period.

38.20.38.4 msgTxInterval (0x0003)

Bits	Field Name	NVM Image Value	Description
15:0	msgTxInterval	0x001E	Time in timer ticks (seconds) between transmissions during normal transmission.

38.20.38.5 LLDP Tx Parameters (0x0004)

Bits	Field Name	NVM Image Value	Description
15:8	txCreditMax	0x05	Determines maximum number of LLDPDUs that can be sent per second.
7:0	msgTxHold	0x04	Used as a multiplier of *msgTxInterval* to determine the txTTL that is carried in the LLDP frames.

38.20.38.6 LLDP Initialization Timers (0x0005)

Timers used by LLDP agent during initialization and when to reinitialize. All times are in seconds.

Bits	Field Name	NVM Image Value	Description
15:8	reinitDelay	0x02	This parameter indicates the amount of delay, in seconds, from when adminStatus becomes disabled until re-initialization is attempted.
7:0	txFastInit	0x04	This variable is used as the initial value for the txFast variable. This value determines the number of LLDPDUs that are transmitted during a fast transmission period.

38.20.38.7 ENDLESS_XOFF_THRESH (0x0006)

Bits	Field Name	NVM Image Value	Description
15:0	ENDLESS_XOFF_THRESH	0x0A	Defines a time limit the firmware waits for XOFF condition during PFR flow. The time is defined in 1 ms units. A possible default setting is 10 ms.

38.20.38.8 DCBX Mode (0x0007)

Defines per-PORT DCBX mode. Note that each port could be configured independently regardless of the other ports configurations.

0: No DCBX is supported.

1: IEEE DCBX only.

2: CEE DCBX only.

3: Auto-select, starting in IEEE (default).

Bits	Field Name	NVM Image Value	Description
15:12	DCBX Port[3]	0x3	Port[3] DCBX mode
11:8	DCBX Port[2]	0x3	Port[2] DCBX mode
7:4	DCBX Port[1]	0x3	Port[1] DCBX mode
3:0	DCBX Port[0]	0x3	Port[0] DCBX mode

38.20.38.9 Current Factory LLDP Admin Status - 0x0008

Defines the status of each LLDP agent. Each LAN port has independent status.

0xF = Value is invalid, use factory LLDP Admin Status.

0x3 = Both receive and transmit enabled.

0x2 = LLDP is configured for transmits only.

0x1 = LLDP is configured for receives only.

0x0 = LLDP agent is disabled.

Bits	Field Name	NVM Image Value	Description
15:12	Port 3	0xF	Defines status of the LLDP agent. Applies to LAN Port 3.
11:8	Port 2	0xF	Defines status of the LLDP agent. Applies to LAN Port 2.
7:4	Port 1	0xF	Defines status of the LLDP agent. Applies to LAN Port 1.
3:0	Port 0	0xF	Defines status of the LLDP agent. Applies to LAN Port 0.

38.20.38.10CRC8 (0x0009)

Bits	Field Name	NVM Image Value	Description
15	CRC Field Used	1b	CRC Field Used 0 = CRC not used. 1 = CRC used.
14:8	Reserved	0x00	Reserved.
7:0	CRC8		CRC-8-CCITT: Start Section -> Word: LLDP Configuration -> Section Length End Section -> Word: LLDP Configuration -> Current Factory LLDP Admin Status

38.20.39 PHY_0 pre-PLL Section Summary Table

Includes static configurations of the internal PHY registers for PLL lock.

Word Offset	Description	Reference
0x0000	Section Length	2663
0x0001	Recipe Type	2663
0x0002	Reg Write Indirect List	2663
0x0003	moduleTypeL	2663
0x0004	moduleTypeH	2663
0x0005	headerLenL	2664
0x0006	headerLenH	2664
0x0007	headerVersionL	2664
0x0008	headerVersionH	2664
0x0009	moduleIDL	2664
0x000A	moduleIDH	2664
0x000B	moduleVendorL	2664
0x000C	moduleVendorH	2664
0x000D	dateL	2665
0x000E	dateH	2665
0x000F	sizeL	2665
0x0010	sizeH	2665
0x0011	keySizeL	2665
0x0012	keySizeH	2665
0x0013	modulusSizeL	2665
0x0014	modulusSizeH	2666
0x0015	exponentSizeL	2666
0x0016	exponentSizeH	2666
0x0017	lad_srevL	2666
0x0018	lad_srevH	2666
0x0019	reserved1L	2666
0x001A	reserved1H	2666
0x001B	lad_fw_entry_offsetL	2666
0x001C	lad_fw_entry_offsetH	2667
0x001D	reserved2L	2667
0x001E	reserved2H	2667
0x001F	lad_image_unique_idL	2667
0x0020	lad_image_unique_idH	2667
0x0021	lad_module_idL	2667
0x0022	lad_module_idH	2667
0x0023 + 1*n, n=0...31	reserved	2668

Word Offset	Description	Reference
0x0043 + 1*n, n=0...127	RSA Public Key	2668
0x00C3	RSA ExponentL	2668
0x00C4	RSA ExponentH	2668
0x00C5 + 1*n, n=0...127	Encrypted SHA256 Hash	2668
0x0145	Device Blank NVM Device ID	2668
0x0146	Max Module AreaL	2668
0x0147	Max Module AreaH	2668
0x0148	Current Module AreaL	2669
0x0149	Current Module AreaH	2669
0x014A	Format Version + CRC	2669
0x014B	Code Revision	2669
0x014C	Reserved Spare Word	2669

38.20.39.1 Section Length (0x0000)

Bits	Field Name	NVM Image Value	Description
15:0	Section Length	0x545	

38.20.39.2 Recipe Type (0x0001)

Bits	Field Name	NVM Image Value	Description
15:0	Recipe Type	0xFFFF	0x0 = SFI. 0x1 = KR. 0x2 = SGMII. 0xFFFF Reserved.

38.20.39.3 Reg Write Indirect List (0x0002)

Raw data module length: variable

38.20.39.4 moduleTypeL (0x0003)

Bits	Field Name	NVM Image Value	Description
15:0	moduleType	0x6	

38.20.39.5 moduleTypeH (0x0004)

Bits	Field Name	NVM Image Value	Description
15:0	moduleTypeH		

38.20.39.6 headerLenL (0x0005)

Bits	Field Name	NVM Image Value	Description
15:0	headerLenL	0xA1	

38.20.39.7 headerLenH (0x0006)

Bits	Field Name	NVM Image Value	Description
15:0	headerLenH		

38.20.39.8 headerVersionL (0x0007)

Bits	Field Name	NVM Image Value	Description
15:0	headerVersionL	0x00010000	

38.20.39.9 headerVersionH (0x0008)

Bits	Field Name	NVM Image Value	Description
15:0	headerVersionH		

38.20.39.10moduleIDL (0x0009)

Bits	Field Name	NVM Image Value	Description
15:0	moduleIDL	0x0	

38.20.39.11moduleIDH (0x000A)

Bits	Field Name	NVM Image Value	Description
15	signMode	0x1	
14:0	moduleIDH		

38.20.39.12moduleVendorL (0x000B)

Bits	Field Name	NVM Image Value	Description
15:0	moduleVendorL	0x00008086	

38.20.39.13moduleVendorH (0x000C)

Bits	Field Name	NVM Image Value	Description
15:0	moduleVendorH		

38.20.39.14dateL (0x000D)

0xMMDD

Bits	Field Name	NVM Image Value	Description
15:0	DateL	0x20130530	0xMMDD

38.20.39.15dateH (0x000E)

0xYYYY

Bits	Field Name	NVM Image Value	Description
15:0	DateH		

38.20.39.16sizeL (0x000F)

Bits	Field Name	NVM Image Value	Description
15:0	sizeL	0x00000800	

38.20.39.17sizeH (0x0010)

Bits	Field Name	NVM Image Value	Description
15:0	sizeH		

38.20.39.18keySizeL (0x0011)

Bits	Field Name	NVM Image Value	Description
15:0	keySizeL	0x40	

38.20.39.19keySizeH (0x0012)

Bits	Field Name	NVM Image Value	Description
15:0	keySizeH		

38.20.39.20modulusSizeL (0x0013)

Bits	Field Name	NVM Image Value	Description
15:0	modulusSizeL	0x40	

38.20.39.21modulusSizeH (0x0014)

Bits	Field Name	NVM Image Value	Description
15:0	modulusSizeH		

38.20.39.22exponentSizeL (0x0015)

Bits	Field Name	NVM Image Value	Description
15:0	exponentSizeL	0x1	

38.20.39.23exponentSizeH (0x0016)

Bits	Field Name	NVM Image Value	Description
15:0	exponentSizeH		

38.20.39.24lad_srevL (0x0017)

Bits	Field Name	NVM Image Value	Description
15:0	lad_srevL	0x0	

38.20.39.25lad_srevH (0x0018)

Bits	Field Name	NVM Image Value	Description
15:0	lad_srevH		

38.20.39.26reserved1L (0x0019)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.39.27reserved1H (0x001A)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.39.28lad_fw_entry_offsetL (0x001B)

Bits	Field Name	NVM Image Value	Description
15:0	lad_fw_entry_offsetL	0x14C	

38.20.39.29lad_fw_entry_offsetH (0x001C)

Bits	Field Name	NVM Image Value	Description
15:0	lad_fw_entry_offsetH		

38.20.39.30reserved2L (0x001D)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.39.31reserved2H (0x001E)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.39.32lad_image_unique_idL (0x001F)

Bits	Field Name	NVM Image Value	Description
15:0	lad_image_unique_idL	0x0	

38.20.39.33lad_image_unique_idH (0x0020)

Bits	Field Name	NVM Image Value	Description
15:0	lad_image_unique_idH		

38.20.39.34lad_module_idL (0x0021)

Bits	Field Name	NVM Image Value	Description
15:0	lad_module_idL	0x6	0x1 = EMP image. 0x2 = PE image. 0x3 = PCIe analog. 0x4 = PHY analog. 0x5 = Option ROM. 0x6 = PHY 0 pre-PLL analog. 0x7 = PHY 0 post-PLL analog. 0x8 = PHY 1 pre-PLL analog. 0x9 = PHY 1 post-PLL analog.

38.20.39.35lad_module_idH (0x0022)

Bits	Field Name	NVM Image Value	Description
15:0	lad_module_idH		

38.20.39.36reserved[n] (0x0023 + 1*n, n=0...31)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.39.37RSA Public Key [n] (0x0043 + 1*n, n=0...127)

Bits	Field Name	NVM Image Value	Description
15:0	RSA Public Key	0x0	

38.20.39.38RSA ExponentL (0x00C3)

Bits	Field Name	NVM Image Value	Description
15:0	RSA ExponentL	0x0	

38.20.39.39RSA ExponentH (0x00C4)

Bits	Field Name	NVM Image Value	Description
15:0	RSA ExponentH	0x0	

38.20.39.40Encrypted SHA256 Hash[n] (0x00C5 + 1*n, n=0...127)

Bits	Field Name	NVM Image Value	Description
15:0	RSA Public Key	0x0	

38.20.39.41Device Blank NVM Device ID (0x0145)

Bits	Field Name	NVM Image Value	Description
15:0	Device Blank NVM Device ID	0x37CC	

38.20.39.42Max Module AreaL (0x0146)

Bits	Field Name	NVM Image Value	Description
15:0	Max Module AreaL	0x1000	

38.20.39.43Max Module AreaH (0x0147)

Bits	Field Name	NVM Image Value	Description
15:0	Max Module AreaH	0x0009	

38.20.39.44Current Module AreaL (0x0148)

Bits	Field Name	NVM Image Value	Description
15:0	Current Module AreaL	0x1000	

38.20.39.45Current Module AreaH (0x0149)

Bits	Field Name	NVM Image Value	Description
15:0	Current Module AreaH	0x0000	

38.20.39.46Format Version + CRC (0x014A)

Bits	Field Name	NVM Image Value	Description
15	CRC Field Used	0x0	0x0 = CRC not used. 0x1 = CRC used.
14:8	Format Version	0x2	
7:0	CRC8	0xFF	

38.20.39.47Code Revision (0x014B)

Bits	Field Name	NVM Image Value	Description
15:8	Major Revision	0x0	
7:0	Minor Revision	0x0	

38.20.39.48Reserved Spare Word (0x014C)

Bits	Field Name	NVM Image Value	Description
15:0	Reserved Spare Word	0x0	

38.20.40 PHY_0 PLL Section Summary Table

Includes static configurations of the internal PHY registers for link establishment.

Word Offset	Description	Reference
0x0000	Section Length	2671
0x0001	Recipe Type	2671
0x0002	Reg Write Indirect List	2671
0x0003	moduleTypeL	2671
0x0004	moduleTypeH	2671
0x0005	headerLenL	2671
0x0006	headerLenH	2671
0x0007	headerVersionL	2672
0x0008	headerVersionH	2672

Word Offset	Description	Reference
0x0009	moduleIDL	2672
0x000A	moduleIDH	2672
0x000B	moduleVendorL	2672
0x000C	moduleVendorH	2672
0x000D	dateL	2672
0x000E	dateH	2672
0x000F	sizeL	2673
0x0010	sizeH	2673
0x0011	keySizeL	2673
0x0012	keySizeH	2673
0x0013	modulusSizeL	2673
0x0014	modulusSizeH	2673
0x0015	exponentSizeL	2673
0x0016	exponentSizeH	2674
0x0017	lad_srevL	2674
0x0018	lad_srevH	2674
0x0019	reserved1L	2674
0x001A	reserved1H	2674
0x001B	lad_fw_entry_offsetL	2674
0x001C	lad_fw_entry_offsetH	2674
0x001D	reserved2L	2674
0x001E	reserved2H	2675
0x001F	lad_image_unique_idL	2675
0x0020	lad_image_unique_idH	2675
0x0021	lad_module_idL	2675
0x0022	lad_module_idH	2675
0x0023 + 1*n, n=0...31	Reserved	2675
0x0043 + 1*n, n=0...127	RSA Public Key	2675
0x00C3	RSA ExponentL	2676
0x00C4	RSA ExponentH	2676
0x00C5 + 1*n, n=0...127	Encrypted SHA256 Hash	2676
0x0145	Device Blank NVM Device ID	2676
0x0146	Max Module AreaL	2676
0x0147	Max Module AreaH	2676
0x0148	Current Module AreaL	2676
0x0149	Current Module AreaH	2676

Word Offset	Description	Reference
0x014A	Format Version + CRC	2677
0x014B	Code Revision	2677
0x014C	Reserved Spare Word	2677

38.20.40.1 Section Length (0x0000)

Bits	Field Name	NVM Image Value	Description
15:0	Section Length	0x5	

38.20.40.2 Recipe Type (0x0001)

Bits	Field Name	NVM Image Value	Description
15:0	Recipe Type	0xFFFF	0x0 = SFI. 0x1 = KR. 0x2 = SGMII. 0xFFFF = Reserved.

38.20.40.3 Reg Write Indirect List (0x0002)

Raw data module length: variable

38.20.40.4 moduleTypeL (0x0003)

Bits	Field Name	NVM Image Value	Description
15:0	moduleType	0x6	

38.20.40.5 moduleTypeH (0x0004)

Bits	Field Name	NVM Image Value	Description
15:0	moduleTypeH		

38.20.40.6 headerLenL (0x0005)

Bits	Field Name	NVM Image Value	Description
15:0	headerLenL	0xA1	

38.20.40.7 headerLenH (0x0006)

Bits	Field Name	NVM Image Value	Description
15:0	headerLenH		

38.20.40.8 headerVersionL (0x0007)

Bits	Field Name	NVM Image Value	Description
15:0	headerVersionL	0x00010000	

38.20.40.9 headerVersionH (0x0008)

Bits	Field Name	NVM Image Value	Description
15:0	headerVersionH		

38.20.40.10moduleIDL (0x0009)

Bits	Field Name	NVM Image Value	Description
15:0	moduleIDL	0x0	

38.20.40.11moduleIDH (0x000A)

Bits	Field Name	NVM Image Value	Description
15	signMode	0x1	
14:0	moduleIDH		

38.20.40.12moduleVendorL (0x000B)

Bits	Field Name	NVM Image Value	Description
15:0	moduleVendorL	0x00008086	

38.20.40.13moduleVendorH (0x000C)

Bits	Field Name	NVM Image Value	Description
15:0	moduleVendorH		

38.20.40.14dateL (0x000D)

0xMMDD

Bits	Field Name	NVM Image Value	Description
15:0	DateL	0x20130530	0xMMDD

38.20.40.15dateH (0x000E)

0xYYYY

Bits	Field Name	NVM Image Value	Description
15:0	DateH		

38.20.40.16sizeL (0x000F)

Bits	Field Name	NVM Image Value	Description
15:0	sizeL	0x00000800	

38.20.40.17sizeH (0x0010)

Bits	Field Name	NVM Image Value	Description
15:0	sizeH		

38.20.40.18keySizeL (0x0011)

Bits	Field Name	NVM Image Value	Description
15:0	keySizeL	0x40	

38.20.40.19keySizeH (0x0012)

Bits	Field Name	NVM Image Value	Description
15:0	keySizeH		

38.20.40.20modulusSizeL (0x0013)

Bits	Field Name	NVM Image Value	Description
15:0	modulusSizeL	0x40	

38.20.40.21modulusSizeH (0x0014)

Bits	Field Name	NVM Image Value	Description
15:0	modulusSizeH		

38.20.40.22exponentSizeL (0x0015)

Bits	Field Name	NVM Image Value	Description
15:0	exponentSizeL	0x1	

38.20.40.23exponentSizeH (0x0016)

Bits	Field Name	NVM Image Value	Description
15:0	exponentSizeH		

38.20.40.24lad_srevL (0x0017)

Bits	Field Name	NVM Image Value	Description
15:0	lad_srevL	0x0	

38.20.40.25lad_srevH (0x0018)

Bits	Field Name	NVM Image Value	Description
15:0	lad_srevH		

38.20.40.26reserved1L (0x0019)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.40.27reserved1H (0x001A)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.40.28lad_fw_entry_offsetL (0x001B)

Bits	Field Name	NVM Image Value	Description
15:0	lad_fw_entry_offsetL	0x14C	

38.20.40.29lad_fw_entry_offsetH (0x001C)

Bits	Field Name	NVM Image Value	Description
15:0	lad_fw_entry_offsetH		

38.20.40.30reserved2L (0x001D)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.40.31reserved2H (0x001E)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.40.32lad_image_unique_idL (0x001F)

Bits	Field Name	NVM Image Value	Description
15:0	lad_image_unique_idL	0x0	

38.20.40.33lad_image_unique_idH (0x0020)

Bits	Field Name	NVM Image Value	Description
15:0	lad_image_unique_idH		

38.20.40.34lad_module_idL (0x0021)

Bits	Field Name	NVM Image Value	Description
15:0	lad_module_idL	0x7	0x1 = EMP image. 0x2 = PE image. 0x3 = PCIe analog. 0x4 = PHY analog. 0x5 = Option ROM. 0x6 = PHY 0 pre-PLL analog. 0x7 = PHY 0 post-PLL analog. 0x8 = PHY 1 pre-PLL analog. 0x9 = PHY 1 post-PLL analog.

38.20.40.35lad_module_idH (0x0022)

Bits	Field Name	NVM Image Value	Description
15:0	lad_module_idH		

38.20.40.36reserved[n] (0x0023 + 1*n, n=0...31)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.40.37RSA Public Key [n] (0x0043 + 1*n, n=0...127)

Bits	Field Name	NVM Image Value	Description
15:0	RSA Public Key	0x0	

38.20.40.38RSA ExponentL (0x00C3)

Bits	Field Name	NVM Image Value	Description
15:0	RSA ExponentL	0x0	

38.20.40.39RSA ExponentH (0x00C4)

Bits	Field Name	NVM Image Value	Description
15:0	RSA ExponentH	0x0	

38.20.40.40Encrypted SHA256 Hash[n] (0x00C5 + 1*n, n=0...127)

Bits	Field Name	NVM Image Value	Description
15:0	RSA Public Key	0x0	

38.20.40.41Device Blank NVM Device ID (0x0145)

Bits	Field Name	NVM Image Value	Description
15:0	Device Blank NVM Device ID	0x37CC	

38.20.40.42Max Module AreaL (0x0146)

Bits	Field Name	NVM Image Value	Description
15:0	Max Module AreaL	0x1000	

38.20.40.43Max Module AreaH (0x0147)

Bits	Field Name	NVM Image Value	Description
15:0	Max Module AreaH	0x0009	

38.20.40.44Current Module AreaL (0x0148)

Bits	Field Name	NVM Image Value	Description
15:0	Current Module AreaL	0x1000	

38.20.40.45Current Module AreaH (0x0149)

Bits	Field Name	NVM Image Value	Description
15:0	Current Module AreaH	0x0000	

38.20.40.46Format Version + CRC (0x014A)

Bits	Field Name	NVM Image Value	Description
15	CRC Field Used	0x0	0x0 = CRC not used. 0x1 = CRC used.
14:8	Format Version	0x2	
7:0	CRC8	0xFF	

38.20.40.47Code Revision (0x014B)

Bits	Field Name	NVM Image Value	Description
15:8	Major Revision	0x0	
7:0	Minor Revision	0x0	

38.20.40.48Reserved Spare Word (0x014C)

Bits	Field Name	Default NVM Value	Description
15:0	Reserved Spare Word	0x0	

38.20.41 PHY_1 pre-PLL Section Summary Table

Includes static configurations of the internal PHY registers for PLL lock.

Word Offset	Description	Reference
0x0000	Section Length	2678
0x0001	Recipe Type	2679
0x0002	Reg Write Indirect List	2679
0x0003	moduleTypeL	2679
0x0004	moduleTypeH	2679
0x0005	headerLenL	2679
0x0006	headerLenH	2679
0x0007	headerVersionL	2679
0x0008	headerVersionH	2680
0x0009	moduleIDL	2680
0x000A	moduleIDH	2680
0x000B	moduleVendorL	2680
0x000C	moduleVendorH	2680
0x000D	dateL	2680
0x000E	dateH	2680
0x000F	sizeL	2681
0x0010	sizeH	2681
0x0011	keySizeL	2681
0x0012	keySizeH	2681

Word Offset	Description	Reference
0x0013	modulusSizeL	2681
0x0014	modulusSizeH	2681
0x0015	exponentSizeL	2681
0x0016	exponentSizeH	2681
0x0017	lad_srevL	2682
0x0018	lad_srevH	2682
0x0019	reserved1L	2682
0x001A	reserved1H	2682
0x001B	lad_fw_entry_offsetL	2682
0x001C	lad_fw_entry_offsetH	2682
0x001D	reserved2L	2682
0x001E	reserved2H	2683
0x001F	lad_image_unique_idL	2683
0x0020	lad_image_unique_idH	2683
0x0021	lad_module_idL	2683
0x0022	lad_module_idH	2683
0x0023 + 1*n, n=0...31	reserved	2683
0x0043 + 1*n, n=0...127	RSA Public Key	2683
0x00C3	RSA ExponentL	2684
0x00C4	RSA ExponentH	2684
0x00C5 + 1*n, n=0...127	Encrypted SHA256 Hash	2684
0x0145	Device Blank NVM Device ID	2684
0x0146	Max Module AreaL	2684
0x0147	Max Module AreaH	2684
0x0148	Current Module AreaL	2684
0x0149	Current Module AreaH	2684
0x014A	Format Version + CRC	2685
0x014B	Code Revision	2685
0x014C	Reserved Spare Word	2685

38.20.41.1 Section Length (0x0000)

Bits	Field Name	NVM Image Value	Description
15:0	Section Length	0x545	

38.20.41.2 Recipe Type (0x0001)

Bits	Field Name	NVM Image Value	Description
15:0	Recipe Type	0xFFFF	0x0 = SFI. 0x1 = KR. 0x2 = SGMII. 0xFFFF = Reserved.

38.20.41.3 Reg Write Indirect List (0x0002)

Raw data module length: variable

38.20.41.4 moduleTypeL (0x0003)

Bits	Field Name	NVM Image Value	Description
15:0	moduleType	0x6	

38.20.41.5 moduleTypeH (0x0004)

Bits	Field Name	NVM Image Value	Description
15:0	moduleTypeH		

38.20.41.6 headerLenL (0x0005)

Bits	Field Name	NVM Image Value	Description
15:0	headerLenL	0xA1	

38.20.41.7 headerLenH (0x0006)

Bits	Field Name	NVM Image Value	Description
15:0	headerLenH		

38.20.41.8 headerVersionL (0x0007)

Bits	Field Name	NVM Image Value	Description
15:0	headerVersionL	0x00010000	

38.20.41.9 headerVersionH (0x0008)

Bits	Field Name	NVM Image Value	Description
15:0	headerVersionH		

38.20.41.10moduleIDL (0x0009)

Bits	Field Name	NVM Image Value	Description
15:0	moduleIDL	0x0	

38.20.41.11moduleIDH (0x000A)

Bits	Field Name	NVM Image Value	Description
15	signMode	0x1	
14:0	moduleIDH		

38.20.41.12moduleVendorL (0x000B)

Bits	Field Name	NVM Image Value	Description
15:0	moduleVendorL	0x00008086	

38.20.41.13moduleVendorH (0x000C)

Bits	Field Name	NVM Image Value	Description
15:0	moduleVendorH		

38.20.41.14dateL (0x000D)

0xMMDD

Bits	Field Name	NVM Image Value	Description
15:0	DateL	0x20130530	0xMMDD

38.20.41.15dateH (0x000E)

0xYYYY

Bits	Field Name	NVM Image Value	Description
15:0	DateH		

38.20.41.16sizeL (0x000F)

Bits	Field Name	NVM Image Value	Description
15:0	sizeL	0x00000800	

38.20.41.17sizeH (0x0010)

Bits	Field Name	NVM Image Value	Description
15:0	sizeH		

38.20.41.18keySizeL (0x0011)

Bits	Field Name	NVM Image Value	Description
15:0	keySizeL	0x40	

38.20.41.19keySizeH (0x0012)

Bits	Field Name	NVM Image Value	Description
15:0	keySizeH		

38.20.41.20modulusSizeL (0x0013)

Bits	Field Name	NVM Image Value	Description
15:0	modulusSizeL	0x40	

38.20.41.21modulusSizeH (0x0014)

Bits	Field Name	NVM Image Value	Description
15:0	modulusSizeH		

38.20.41.22exponentSizeL (0x0015)

Bits	Field Name	NVM Image Value	Description
15:0	exponentSizeL	0x1	

38.20.41.23exponentSizeH (0x0016)

Bits	Field Name	NVM Image Value	Description
15:0	exponentSizeH		

38.20.41.24lad_srevL (0x0017)

Bits	Field Name	NVM Image Value	Description
15:0	lad_srevL	0x0	

38.20.41.25lad_srevH (0x0018)

Bits	Field Name	NVM Image Value	Description
15:0	lad_srevH		

38.20.41.26reserved1L (0x0019)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.41.27reserved1H (0x001A)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.41.28lad_fw_entry_offsetL (0x001B)

Bits	Field Name	NVM Image Value	Description
15:0	lad_fw_entry_offsetL	0x14C	

38.20.41.29lad_fw_entry_offsetH (0x001C)

Bits	Field Name	NVM Image Value	Description
15:0	lad_fw_entry_offsetH		

38.20.41.30reserved2L (0x001D)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.41.31reserved2H (0x001E)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.41.32lad_image_unique_idL (0x001F)

Bits	Field Name	NVM Image Value	Description
15:0	lad_image_unique_idL	0x0	

38.20.41.33lad_image_unique_idH (0x0020)

Bits	Field Name	NVM Image Value	Description
15:0	lad_image_unique_idH		

38.20.41.34lad_module_idL (0x0021)

Bits	Field Name	NVM Image Value	Description
15:0	lad_module_idL	0x8	0x1 = EMP image. 0x2 = PE image. 0x3 = PCIe analog. 0x4 = PHY analog. 0x5 = Option ROM. 0x6 = PHY 0 pre-PLL analog. 0x7 = PHY 0 post-PLL analog. 0x8 = PHY 1 pre-PLL analog. 0x9 = PHY 1 post-PLL analog.

38.20.41.35lad_module_idH (0x0022)

Bits	Field Name	NVM Image Value	Description
15:0	lad_module_idH		

38.20.41.36reserved[n] (0x0023 + 1*n, n=0...31)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.41.37RSA Public Key [n] (0x0043 + 1*n, n=0...127)

Bits	Field Name	NVM Image Value	Description
15:0	RSA Public Key	0x0	

38.20.41.38RSA ExponentL (0x00C3)

Bits	Field Name	NVM Image Value	Description
15:0	RSA ExponentL	0x0	

38.20.41.39RSA ExponentH (0x00C4)

Bits	Field Name	NVM Image Value	Description
15:0	RSA ExponentH	0x0	

38.20.41.40Encrypted SHA256 Hash[n] (0x00C5 + 1*n, n=0...127)

Bits	Field Name	NVM Image Value	Description
15:0	RSA Public Key	0x0	

38.20.41.41Device Blank NVM Device ID (0x0145)

Bits	Field Name	NVM Image Value	Description
15:0	Device Blank NVM Device ID	0x37CC	

38.20.41.42Max Module AreaL (0x0146)

Bits	Field Name	NVM Image Value	Description
15:0	Max Module AreaL	0x1000	

38.20.41.43Max Module AreaH (0x0147)

Bits	Field Name	NVM Image Value	Description
15:0	Max Module AreaH	0x0009	

38.20.41.44Current Module AreaL (0x0148)

Bits	Field Name	NVM Image Value	Description
15:0	Current Module AreaL	0x1000	

38.20.41.45Current Module AreaH (0x0149)

Bits	Field Name	NVM Image Value	Description
15:0	Current Module AreaH	0x0000	

38.20.41.46Format Version + CRC (0x014A)

Bits	Field Name	NVM Image Value	Description
15	CRC Field Used	0x0	0x0 = CRC not used. 0x1 = CRC used.
14:8	Format Version	0x2	
7:0	CRC8	0xFF	

38.20.41.47Code Revision (0x014B)

Bits	Field Name	NVM Image Value	Description
15:8	Major Revision	0x0	
7:0	Minor Revision	0x0	

38.20.41.48Reserved Spare Word (0x014C)

Bits	Field Name	NVM Image Value	Description
15:0	Reserved Spare Word	0x0	

38.20.42 PHY_1 PLL Section Summary Table

Includes static configurations of the internal PHY registers for link establishment

Word Offset	Description	Reference
0x0000	Section Length	2687
0x0001	Recipe Type	2687
0x0002	Reg Write Indirect List	2687
0x0003	moduleTypeL	2687
0x0004	moduleTypeH	2687
0x0005	headerLenL	2687
0x0006	headerLenH	2687
0x0007	headerVersionL	2688
0x0008	headerVersionH	2688
0x0009	moduleIDL	2688
0x000A	moduleIDH	2688
0x000B	moduleVendorL	2688
0x000C	moduleVendorH	2688
0x000D	dateL	2688
0x000E	dateH	2688
0x000F	sizeL	2689
0x0010	sizeH	2689
0x0011	keySizeL	2689
0x0012	keySizeH	2689

Word Offset	Description	Reference
0x0013	modulusSizeL	2689
0x0014	modulusSizeH	2689
0x0015	exponentSizeL	2689
0x0016	exponentSizeH	2690
0x0017	lad_srevL	2690
0x0018	lad_srevH	2690
0x0019	reserved1L	2690
0x001A	reserved1H	2690
0x001B	lad_fw_entry_offsetL	2690
0x001C	lad_fw_entry_offsetH	2690
0x001D	reserved2L	2690
0x001E	reserved2H	2691
0x001F	lad_image_unique_idL	2691
0x0020	lad_image_unique_idH	2691
0x0021	lad_module_idL	2691
0x0022	lad_module_idH	2691
0x0023 + 1*n, n=0...31	reserved	2691
0x0043 + 1*n, n=0...127	RSA Public Key	2691
0x00C3	RSA ExponentL	2692
0x00C4	RSA ExponentH	2692
0x00C5 + 1*n, n=0...127	Encrypted SHA256 Hash	2692
0x0145	Device Blank NVM Device ID	2692
0x0146	Max Module AreaL	2692
0x0147	Max Module AreaH	2692
0x0148	Current Module AreaL	2692
0x0149	Current Module AreaH	2692
0x014A	Format Version + CRC	2693
0x014B	Code Revision	2693
0x014C	Reserved Spare Word	2693

38.20.42.1 Section Length (0x0000)

Bits	Field Name	NVM Image Value	Description
15:0	Section Length	0x5	

38.20.42.2 Recipe Type (0x0001)

Bits	Field Name	NVM Image Value	Description
15:0	Recipe Type	0xFFFF	0x0 = SFI. 0x1 = KR. 0x2 = SGMII. 0xFFFF = Reserved.

38.20.42.3 Reg Write Indirect List (0x0002)

Raw data module length: variable

38.20.42.4 moduleTypeL (0x0003)

Bits	Field Name	NVM Image Value	Description
15:0	moduleType	0x6	

38.20.42.5 moduleTypeH (0x0004)

Bits	Field Name	NVM Image Value	Description
15:0	moduleTypeH		

38.20.42.6 headerLenL (0x0005)

Bits	Field Name	NVM Image Value	Description
15:0	headerLenL	0xA1	

38.20.42.7 headerLenH (0x0006)

Bits	Field Name	NVM Image Value	Description
15:0	headerLenH		

38.20.42.8 headerVersionL (0x0007)

Bits	Field Name	NVM Image Value	Description
15:0	headerVersionL	0x00010000	

38.20.42.9 headerVersionH (0x0008)

Bits	Field Name	NVM Image Value	Description
15:0	headerVersionH		

38.20.42.10moduleIDL (0x0009)

Bits	Field Name	NVM Image Value	Description
15:0	moduleIDL	0x0	

38.20.42.11moduleIDH (0x000A)

Bits	Field Name	NVM Image Value	Description
15	signMode	0x1	
14:0	moduleIDH		

38.20.42.12moduleVendorL (0x000B)

Bits	Field Name	NVM Image Value	Description
15:0	moduleVendorL	0x00008086	

38.20.42.13moduleVendorH (0x000C)

Bits	Field Name	NVM Image Value	Description
15:0	moduleVendorH		

38.20.42.14dateL (0x000D)

0xMMDD

Bits	Field Name	NVM Image Value	Description
15:0	DateL	0x20130530	0xMMDD

38.20.42.15dateH (0x000E)

0xYYYY

Bits	Field Name	NVM Image Value	Description
15:0	DateH		

38.20.42.16sizeL (0x000F)

Bits	Field Name	NVM Image Value	Description
15:0	sizeL	0x00000800	

38.20.42.17sizeH (0x0010)

Bits	Field Name	NVM Image Value	Description
15:0	sizeH		

38.20.42.18keySizeL (0x0011)

Bits	Field Name	NVM Image Value	Description
15:0	keySizeL	0x40	

38.20.42.19keySizeH (0x0012)

Bits	Field Name	NVM Image Value	Description
15:0	keySizeH		

38.20.42.20modulusSizeL (0x0013)

Bits	Field Name	NVM Image Value	Description
15:0	modulusSizeL	0x40	

38.20.42.21modulusSizeH (0x0014)

Bits	Field Name	NVM Image Value	Description
15:0	modulusSizeH		

38.20.42.22exponentSizeL (0x0015)

Bits	Field Name	NVM Image Value	Description
15:0	exponentSizeL	0x1	

38.20.42.23exponentSizeH (0x0016)

Bits	Field Name	NVM Image Value	Description
15:0	exponentSizeH		

38.20.42.24lad_srevL (0x0017)

Bits	Field Name	NVM Image Value	Description
15:0	lad_srevL	0x0	

38.20.42.25lad_srevH (0x0018

Bits	Field Name	NVM Image Value	Description
15:0	lad_srevH		

38.20.42.26reserved1L (0x0019)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.42.27reserved1H (0x001A)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.42.28lad_fw_entry_offsetL (0x001B)

Bits	Field Name	NVM Image Value	Description
15:0	lad_fw_entry_offsetL	0x14C	

38.20.42.29lad_fw_entry_offsetH (0x001C)

Bits	Field Name	NVM Image Value	Description
15:0	lad_fw_entry_offsetH		

38.20.42.30reserved2L (0x001D)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.42.31reserved2H (0x001E)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.42.32lad_image_unique_idL (0x001F)

Bits	Field Name	NVM Image Value	Description
15:0	lad_image_unique_idL	0x0	

38.20.42.33lad_image_unique_idH (0x0020)

Bits	Field Name	NVM Image Value	Description
15:0	lad_image_unique_idH		

38.20.42.34lad_module_idL (0x0021)

Bits	Field Name	NVM Image Value	Description
15:0	lad_module_idL	0x9	0x1 = EMP image. 0x2 = PE image. 0x3 = PCIe analog. 0x4 = PHY analog. 0x5 = Option ROM. 0x6 = PHY 0 pre-PLL analog. 0x7 = PHY 0 post-PLL analog. 0x8 = PHY 1 pre-PLL analog. 0x9 = PHY 1 post-PLL analog.

38.20.42.35lad_module_idH (0x0022)

Bits	Field Name	NVM Image Value	Description
15:0	lad_module_idH		

38.20.42.36reserved[n] (0x0023 + 1*n, n=0...31)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.42.37RSA Public Key [n] (0x0043 + 1*n, n=0...127)

Bits	Field Name	NVM Image Value	Description
15:0	RSA Public Key	0x0	

38.20.42.38RSA ExponentL (0x00C3)

Bits	Field Name	NVM Image Value	Description
15:0	RSA ExponentL	0x0	

38.20.42.39RSA ExponentH (0x00C4)

Bits	Field Name	NVM Image Value	Description
15:0	RSA ExponentH	0x0	

38.20.42.40Encrypted SHA256 Hash[n] (0x00C5 + 1*n, n=0...127)

Bits	Field Name	NVM Image Value	Description
15:0	RSA Public Key	0x0	

38.20.42.41Device Blank NVM Device ID (0x0145)

Bits	Field Name	NVM Image Value	Description
15:0	Device Blank NVM Device ID	0x37CC	

38.20.42.42Max Module AreaL (0x0146)

Bits	Field Name	NVM Image Value	Description
15:0	Max Module AreaL	0x1000	

38.20.42.43Max Module AreaH (0x0147)

Bits	Field Name	NVM Image Value	Description
15:0	Max Module AreaH	0x0009	

38.20.42.44Current Module AreaL (0x0148)

Bits	Field Name	NVM Image Value	Description
15:0	Current Module AreaL	0x1000	

38.20.42.45Current Module AreaH (0x0149)

Bits	Field Name	NVM Image Value	Description
15:0	Current Module AreaH	0x0000	

38.20.42.46Format Version + CRC (0x014A)

Bits	Field Name	NVM Image Value	Description
15	CRC Field Used	0x0	0x0 CRC not used. 0x1 CRC used.
14:8	Format Version	0x2	
7:0	CRC8	0xFF	

38.20.42.47Code Revision (0x014B)

Bits	Field Name	NVM Image Value	Description
15:8	Major Revision	0x0	
7:0	Minor Revision	0x0	

38.20.42.48Reserved Spare Word (0x014C)

Bits	Field Name	NVM Image Value	Description
15:0	Reserved Spare Word	0x0	

38.20.43 2nd Free Provisioning Area Section Summary Table

Free area used as the new bank when updating 8 KB long modules which are mapped outside the shadow RAM.

Word Offset	Description	Reference
0x1A000	[New Word]	2693

38.20.43.1 [New Word] (0x1A000)

Raw data module length: variable.

38.20.44 OROM Section Summary Table

Option ROM Module.

It contains pre-boot code and settings read by BIOS

Word Offset	Description	Reference
0x0000	OROM Data	2695
0x0001	moduleTypeL	2695
0x0002	moduleTypeH	2695
0x0003	headerLenL	2695
0x0004	headerLenH	2695
0x0005	headerVersionL	2695
0x0006	headerVersionH	2695
0x0007	moduleIDL	2695

Word Offset	Description	Reference
0x0008	moduleIDH	2696
0x0009	moduleVendorL	2696
0x000A	moduleVendorH	2696
0x000B	dateL	2696
0x000C	dateH	2696
0x000D	sizeL	2696
0x000E	sizeH	2696
0x000F	keySizeL	2697
0x0010	keySizeH	2697
0x0011	modulusSizeL	2697
0x0012	modulusSizeH	2697
0x0013	exponentSizeL	2697
0x0014	exponentSizeH	2697
0x0015	lad_srevL	2697
0x0016	lad_srevH	2697
0x0017	reserved1L	2698
0x0018	reserved1H	2698
0x0019	lad_fw_entry_offsetL	2698
0x001A	lad_fw_entry_offsetH	2698
0x001B	reserved2L	2698
0x001C	reserved2H	2698
0x001D	lad_image_unique_idL	2698
0x001E	lad_image_unique_idH	2698
0x001F	lad_module_idL	2699
0x0020	lad_module_idH	2699
0x0021 + 1*n, n=0...31	Reserved	2699
0x0041 + 1*n, n=0...127	RSA Public Key	2699
0x00C1	RSA ExponentL	2699
0x00C2	RSA ExponentH	2699
0x00C3 + 1*n, n=0...127	Encrypted SHA256 Hash	2699
0x0143	Device Blank NVM Device ID	2699
0x0144	Max Module AreaL	2699
0x0145	Max Module AreaH	2700
0x0146	Current Module AreaL	2700
0x0147	Current Module AreaH	2700
0x0148	Format Version + CRC	2700
0x0149	Code Revision	2700
0x014A	Reserved Spare Word	2700

38.20.44.1 OROM Data (0x0000)

Raw data module length: variable

38.20.44.2 moduleTypeL (0x0001)

Bits	Field Name	NVM Image Value	Description
15:0	moduleType	0x6	

38.20.44.3 moduleTypeH (0x0002)

Bits	Field Name	NVM Image Value	Description
15:0	moduleTypeH		

38.20.44.4 headerLenL (0x0003)

Bits	Field Name	NVM Image Value	Description
15:0	headerLenL	0xA1	

38.20.44.5 headerLenH (0x0004)

Bits	Field Name	NVM Image Value	Description
15:0	headerLenH		

38.20.44.6 headerVersionL (0x0005)

Bits	Field Name	NVM Image Value	Description
15:0	headerVersionL	0x00010000	

38.20.44.7 headerVersionH (0x0006)

Bits	Field Name	NVM Image Value	Description
15:0	headerVersionH		

38.20.44.8 moduleIDL (0x0007)

Bits	Field Name	NVM Image Value	Description
15:0	moduleIDL	0x0	

38.20.44.9 moduleIDH (0x0008)

Bits	Field Name	NVM Image Value	Description
15	signMode	0x1	
14:0	moduleIDH		

38.20.44.10moduleVendorL (0x0009)

Bits	Field Name	NVM Image Value	Description
15:0	moduleVendorL	0x00008086	

38.20.44.11moduleVendorH (0x000A)

Bits	Field Name	NVM Image Value	Description
15:0	moduleVendorH		

38.20.44.12dateL (0x000B)

0xMMDD

Bits	Field Name	NVM Image Value	Description
15:0	DateL	0x20130530	0xMMDD

38.20.44.13dateH (x000C)

0xYYYY

Bits	Field Name	NVM Image Value	Description
15:0	DateH		

38.20.44.14sizeL (0x000D)

Bits	Field Name	NVM Image Value	Description
15:0	sizeL	0x0004E000	

38.20.44.15sizeH (0x000E)

Bits	Field Name	NVM Image Value	Description
15:0	sizeH		

38.20.44.16keySizeL (0x000F)

Bits	Field Name	NVM Image Value	Description
15:0	keySizeL	0x40	

38.20.44.17keySizeH (0x0010)

Bits	Field Name	NVM Image Value	Description
15:0	keySizeH		

38.20.44.18modulusSizeL (0x0011)

Bits	Field Name	NVM Image Value	Description
15:0	modulusSizeL	0x40	

38.20.44.19modulusSizeH (0x0012)

Bits	Field Name	NVM Image Value	Description
15:0	modulusSizeH		

38.20.44.20exponentSizeL (0x0013)

Bits	Field Name	NVM Image Value	Description
15:0	exponentSizeL	0x1	

38.20.44.21exponentSizeH (0x0014)

Bits	Field Name	NVM Image Value	Description
15:0	exponentSizeH		

38.20.44.22lad_srevL (0x0015)

Bits	Field Name	NVM Image Value	Description
15:0	lad_srevL	0x0	

38.20.44.23lad_srevH (0x0016)

Bits	Field Name	NVM Image Value	Description
15:0	lad_srevH		

38.20.44.24reserved1L (0x0017)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.44.25reserved1H (0x0018)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.44.26lad_fw_entry_offsetL (0x0019)

Bits	Field Name	NVM Image Value	Description
15:0	lad_fw_entry_offsetL	0x14C	

38.20.44.27lad_fw_entry_offsetH (0x001A)

Bits	Field Name	NVM Image Value	Description
15:0	lad_fw_entry_offsetH		

38.20.44.28reserved2L (0x001B)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.44.29reserved2H (0x001C)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.44.30lad_image_unique_idL (0x001D)

Bits	Field Name	NVM Image Value	Description
15:0	lad_image_unique_idL	0x0	

38.20.44.31lad_image_unique_idH (0x001E)

Bits	Field Name	NVM Image Value	Description
15:0	lad_image_unique_idH		

38.20.44.32lad_module_idL (0x001F)

Bits	Field Name	NVM Image Value	Description
15:0	lad_module_idL	0x5	0x1 = EMP image. 0x2 = PE image. 0x3 = PCIe analog. 0x4 = PHY analog. 0x5 = Option ROM.

38.20.44.33lad_module_idH (0x0020)

Bits	Field Name	NVM Image Value	Description
15:0	lad_module_idH		

38.20.44.34RSA Public Key [n] (0x0041 + 1*n, n=0...127)

Bits	Field Name	NVM Image Value	Description
15:0	RSA Public Key	0x0	

38.20.44.35RSA ExponentL (0x00C1)

Bits	Field Name	NVM Image Value	Description
15:0	RSA ExponentL	0x0	

38.20.44.36RSA ExponentH (0x00C2)

Bits	Field Name	NVM Image Value	Description
15:0	RSA ExponentH	0x0	

38.20.44.37Encrypted SHA256 Hash[n] (0x00C3 + 1*n, n=0...127)

Bits	Field Name	NVM Image Value	Description
15:0	RSA Public Key	0x0	

38.20.44.38Device Blank NVM Device ID (0x0143)

Bits	Field Name	NVM Image Value	Description
15:0	Device Blank NVM Device ID	0x37CC	

38.20.44.39Max Module AreaL (0x0144)

Bits	Field Name	NVM Image Value	Description
15:0	Max Module AreaL	0x1000	

38.20.44.40Max Module AreaH (0x0145)

Bits	Field Name	NVM Image Value	Description
15:0	Max Module AreaH	0x0009	

38.20.44.41Current Module AreaL (0x0146)

Bits	Field Name	NVM Image Value	Description
15:0	Current Module AreaL	0x1000	

38.20.44.42Current Module AreaH (0x0147)

Bits	Field Name	NVM Image Value	Description
15:0	Current Module AreaH	0x0000	

38.20.44.43Format Version + CRC (0x0148)

Bits	Field Name	NVM Image Value	Description
15	CRC Field Used	0x0	0x0 = CRC not used. 0x1 = CRC used.
14:8	Format Version	0x2	
7:0	CRC8	0xFF	

38.20.44.44Code Revision (0x0149)

Bits	Field Name	NVM Image Value	Description
15:8	Major Revision	0x0	
7:0	Minor Revision	0x0	

38.20.44.45Reserved Spare Word (0x014A)

Bits	Field Name	NVM Image Value	Description
15:0	Reserved Spare Word	0x0	

38.20.45 EMP Image Section Summary Table

Word Offset	Description	Reference
0x0000	moduleTypeL	2702
0x0001	moduleTypeH	2702
0x0002	headerLenL	2702
0x0003	headerLenH	2702
0x0004	headerVersionL	2702

Word Offset	Description	Reference
0x0005	headerVersionH	2702
0x0006	moduleIDL	2703
0x0007	moduleIDH	2703
0x0008	moduleVendorL	2703
0x0009	moduleVendorH	2703
0x000A	dateL	2703
0x000B	dateH	2703
0x000C	sizeL	2704
0x000D	sizeH	2704
0x000E	keySizeL	2704
0x000F	keySizeH	2704
0x0010	modulusSizeL	2704
0x0011	modulusSizeH	2704
0x0012	exponentSizeL	2704
0x0013	exponentSizeH	2704
0x0014	lad_srevL	2705
0x0015	lad_srevH	2705
0x0016	reserved1L	2705
0x0017	reserved1H	2705
0x0018	lad_fw_entry_offsetL	2705
0x0019	lad_fw_entry_offsetH	2705
0x001A	reserved2L	2705
0x001B	reserved2H	2705
0x001C	lad_image_unique_idL	2706
0x001D	lad_image_unique_idH	2706
0x001E	lad_module_idL	2706
0x001F	lad_module_idH	2706
0x0020 + 1*n, n=0...31	reserved[n]	2706
0x0040 + 1*n, n=0...127	RSA Public Key	2706
0x00C0	RSA ExponentL	2706
0x00C1	RSA ExponentH	2706
0x00C2 + 1*n, n=0...127	Encrypted SHA256 Hash	2707
0x0142	Device Blank NVM Device ID	2707
0x0143	Max Module AreaL	2707
0x0144	Max Module AreaH	2707
0x0145	Current Module AreaL	2707
0x0146	Current Module AreaH	2707
0x0147	Format Version + CRC	2707

Word Offset	Description	Reference
0x0148	Code Revision	2708
0x0149	Reserved Spare Word	2708
0x014A	EMP Image Raw Data	2708
0x014B	RO Trailer	2708

38.20.45.1 moduleTypeL (0x0000)

Bits	Field Name	NVM Image Value	Description
15:0	moduleType	0x6	

38.20.45.2 moduleTypeH (0x0001)

Bits	Field Name	NVM Image Value	Description
15:0	moduleTypeH		

38.20.45.3 headerLenL (0x0002)

Bits	Field Name	NVM Image Value	Description
15:0	headerLenL	0xa1	

38.20.45.4 headerLenH (0x0003)

Bits	Field Name	NVM Image Value	Description
15:0	headerLenH		

38.20.45.5 headerVersionL (0x0004)

Bits	Field Name	NVM Image Value	Description
15:0	headerVersionL	0x00010000	

38.20.45.6 headerVersionH (0x0005)

Bits	Field Name	NVM Image Value	Description
15:0	headerVersionH		

38.20.45.7 moduleIDL (0x0006)

Bits	Field Name	NVM Image Value	Description
15:0	moduleIDL	0x0	

38.20.45.8 moduleIDH (0x0007)

Bits	Field Name	NVM Image Value	Description
15	signMode	0x1	
14:0	moduleIDH		

38.20.45.9 moduleVendorL (0x0008)

Bits	Field Name	NVM Image Value	Description
15:0	moduleVendorL	0x00008086	

38.20.45.10moduleVendorH (0x0009)

Bits	Field Name	NVM Image Value	Description
15:0	moduleVendorH		

38.20.45.11dateL (0x000A)

0xMMDD

Bits	Field Name	NVM Image Value	Description
15:0	DateL	0x20130530	0xMMDD

38.20.45.12dateH (0x000B)

0xYYYY

Bits	Field Name	NVM Image Value	Description
15:0	DateH		

38.20.45.13sizeL (0x000C)

Bits	Field Name	NVM Image Value	Description
15:0	sizeL	0x0004E000	

38.20.45.14sizeH (0x000D)

Bits	Field Name	NVM Image Value	Description
15:0	sizeH		

38.20.45.15keySizeL (0x000E)

Bits	Field Name	NVM Image Value	Description
15:0	keySizeL	0x40	

38.20.45.16keySizeH (0x000F)

Bits	Field Name	NVM Image Value	Description
15:0	keySizeH		

38.20.45.17modulusSizeL (0x0010)

Bits	Field Name	NVM Image Value	Description
15:0	modulusSizeL	0x40	

38.20.45.18modulusSizeH (0x0011)

Bits	Field Name	NVM Image Value	Description
15:0	modulusSizeH		

38.20.45.19exponentSizeL (0x0012)

Bits	Field Name	NVM Image Value	Description
15:0	exponentSizeL	0x1	

38.20.45.20exponentSizeH (0x0013)

Bits	Field Name	NVM Image Value	Description
15:0	exponentSizeH		

38.20.45.21lad_srevL (0x0014)

Bits	Field Name	NVM Image Value	Description
15:0	lad_srevL	0x0	

38.20.45.22lad_srevH (0x0015)

Bits	Field Name	NVM Image Value	Description
15:0	lad_srevH		

38.20.45.23reserved1L (0x0016)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.45.24reserved1H (0x0017)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.45.25lad_fw_entry_offsetL (0x0018)

Bits	Field Name	NVM Image Value	Description
15:0	lad_fw_entry_offsetL	0x3000	

38.20.45.26lad_fw_entry_offsetH (0x0019)

Bits	Field Name	NVM Image Value	Description
15:0	lad_fw_entry_offsetH	0x0013	

38.20.45.27reserved2L (0x001A)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.45.28reserved2H (0x001B)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.45.29lad_image_unique_idL (0x001C)

Bits	Field Name	NVM Image Value	Description
15:0	lad_image_unique_idL	0x0	

38.20.45.30lad_image_unique_idH (0x001D)

Bits	Field Name	NVM Image Value	Description
15:0	lad_image_unique_idH		

38.20.45.31lad_module_idL (0x001E)

Bits	Field Name	NVM Image Value	Description
15:0	lad_module_idL	0x1	0x1 = EMP Image. 0x2 = PE Image. 0x3 = PCIe Analog. 0x4 =PHY Analog. 0x5 = Option ROM.

38.20.45.32lad_module_idH (0x001F)

Bits	Field Name	NVM Image Value	Description
15:0	lad_module_idH		

38.20.45.33reserved[n] (0x0020 + 1*n, n=0...31)

Bits	Field Name	NVM Image Value	Description
15:0	reserved	0x0	

38.20.45.34RSA Public Key [n] (0x0040 + 1*n, n=0...127)

Bits	Field Name	NVM Image Value	Description
15:0	RSA Public Key	0x0	

38.20.45.35RSA ExponentL (0x00C0)

Bits	Field Name	NVM Image Value	Description
15:0	RSA ExponentL	0x0	

38.20.45.36RSA ExponentH (0x00C1)

Bits	Field Name	NVM Image Value	Description
15:0	RSA ExponentH	0x0	

38.20.45.37Encrypted SHA256 Hash[n] (0x00C2 + 1*n, n=0...127)

Bits	Field Name	NVM Image Value	Description
15:0	RSA Public Key	0x0	

38.20.45.38Device Blank NVM Device ID (0x0142)

Bits	Field Name	NVM Image Value	Description
15:0	Device Blank NVM Device ID	0x37CC	

38.20.45.39Max Module AreaL (0x0143)

Bits	Field Name	NVM Image Value	Description
15:0	Max Module AreaL	0x1000	

38.20.45.40Max Module AreaH (0x0144)

Bits	Field Name	NVM Image Value	Description
15:0	Max Module AreaH	0x0009	

38.20.45.41Current Module AreaL (0x0145)

Bits	Field Name	NVM Image Value	Description
15:0	Current Module AreaL	0xE000	

38.20.45.42Current Module AreaH (0x0146)

Bits	Field Name	NVM Image Value	Description
15:0	Current Module AreaH	0x0006	

38.20.45.43Format Version + CRC (0x0147)

Bits	Field Name	NVM Image Value	Description
15	CRC Field Used	0x1	0x0 = CRC not used. 0x1 = CRC Used.
14:8	Format Version	0x2	
7:0	CRC8		CRC-8-CCITT: Start Section -> Word: EMP Image -> moduleTypeL End Section -> Word: EMP Image -> RO Trailer

38.20.45.44Code Revision (0x0148)

Bits	Field Name	NVM Image Value	Description
15:8	Major Revision	0x0	
7:0	Minor Revision	0x0	

38.20.45.45Reserved Spare Word (0x0149)

Bits	Field Name	NVM Image Value	Description
15:0	Reserved Spare Word	0x0	

38.20.45.46EMP Image Raw Data (0x014A)

Raw data module length: variable

38.20.45.47RO Trailer (0x014B)

Raw data module length: variable

38.20.46 1st Free Provisioning Area Section Summary Table

Free area used as the new bank when updating 880 KB long modules.

Word Offset	Description	Reference
0x0000	[New Word]	2708

38.20.46.1 [New Word] (0x0000)

Raw data module length: variable.

38.20.47 4th Free Provisioning Area Section Summary Table

Free area to be used as the new bank when updating future 64 KB long modules mapped outside the shadow RAM.

Word Offset	Description	Reference
0x0000	[New Word]	2708

38.20.47.1 [New Word] - 0x0000

Raw data module length: variable

38.20.48 3rd Free Provisioning Area Section Summary Table

Free area to be used as the new bank when updating future 16 KB long modules mapped outside the shadow RAM.

Word Offset	Description	Reference
0x0000	[New Word]	2709

38.20.48.1 [New Word] 0x0000)

Raw data module length: variable.

38.20.49 Firmware Scratch Pad Area Section Summary Table

Firmware scratch pad area used by firmware only.

Word Offset	Description	Reference
0x0000	[New Word]	2709

38.20.49.1 [New Word] - 0x0000

Raw data module length: variable.

38.20.50 Configuration Metadata Section Summary Table

NVM Adaptive Configuration Metadata

Word Offset	Description	Reference

38.21 Shared Resources

38.21.1 Receive Classification Filters

38.21.1.1 Introduction

This section describes the receive classification filters that direct inbound packets to a processing engine or cluster out of the following: LAN engine and RDMA engine (also named as Protocol Engine or PE). The filters also define a specific queue or a context within the selected engine. All the filters described in this section relate to filtering within a specific VSI which is defined by the switch filters.

The classification filters and other header processing units in the device inspect the first 480 bytes of the received packets. If the identified headers exceed 480 bytes, they are reported as an unidentified L2 packet type with no offloads other than Ethernet CRC. They are also handled this way by the classification filters.

38.21.1.1.1 Association with a Packet Engine

The general rule is that a frame is tested to be PE types. If the frame is neither, it is considered to be a LAN frame. The PE protocol uses the services of the LAN engine for certain frames. Such frames are forwarded to the LAN engine.

- Filtering to the PE or LAN is based on a set of rules defined in Section 38.21.3.8.1 and a set of the following filters: APBVT L4 port table (described in Section 38.21.3.8.2) and PE quad-hash filter (described in Section 38.21.3.8.3).

38.21.1.1.2 Receive Classification Filters Priority and Usage

Received traffic goes through a set of filters that determine the destination of each received frame. The receive classification filters operate on frames received from the network as well as frames forwarded by the internal switch from a local port (VSI). If a frame is replicated by the internal switch, each replica goes independently through the filters.

The programming interface of the filters are described in the following sections.

The following filters operate on received frames listed in priority ordering. A frame can match multiple filters. If multiple filters define the same action, then the higher priority enabled matched filter takes precedence. If a frame does not match any of the filters below (with a queue action), the frame is directed to the default queue (queue zero of the VSI).

- PE quad-hash filters — see Section 38.21.3.8.3
- Ether-type — see Section 38.21.3.4
- Flow Director (FD)
- MAC/VLAN — see Section 38.21.3.4
- Hash filters (including RSS) — see Section 38.21.3.6

Note: In proper programming, the following filters are expected to be mutually exclusive. Meaning, the same packet is not expected to match more than one of them: PE Quad hash filter, Ether-type filter and FD filter.

The PE quad hash filter are enabled per VSI by the VSIQF_CTL registers. The Hash filter is enabled per function per packet type by the PFQF_HENA for the PFs and the VFQF_HENA for the VFs. The Ether-type filter, MAC/VLAN filter and FD filter are enabled per PF and its VFs by the PFQF_CTL_0 registers.

The Table 38-124 lists the typical use cases of the classification filters and its programming option per function.

Table 38-124.Classification filters per PF/VF

VSI	MAC/VLAN, S-Tag and Cloud Switch Filters	Ether-type	PE Quad Hash	Flow Director (FD)	Hash
Main Usage	VMDq1	Control Port / L2 Protocols	RDMA	Flex Filters[1]	Load Balance
Programming exposed to the PF	Yes	Yes	Yes [a]	Yes	Yes
Programming exposed to the VF	No	No	Yes [a]	No	No
1.					

Notes:

1. There are two main usages for FD filters: software driver controlled filters for ATR and user/operating system controlled filters.

Once a frame matches a filter, the filter provides one or more actions to be performed on the frame. Listed below are those actions enabled by the filters while Table 38-125 lists which of these actions are enabled per filter type:

- Accept / reject — A decision whether the frame is dropped by the hardware or posted to the device engines for its processing and optionally posted to host memory.
- LAN / PE — A decision whether the frame is posted to LAN or PE engines.
- Receive queue — Associate the frame to a queue index within the VSI space.

- Context ID — Match the frame to a context ID associated with the flow (PE QPN).
- Increment statistical counter — Increment the statistical counters associated with the filter.
- Software field (filter ID) — Filter ID provided at filter programming and reported back with the matched received packets.
- Packet Bytes — Post bytes from the packet to the receive descriptor. Bytes that should be reported in the receive descriptor can be taken from the *Flexible Bytes* in the field vector.

Table 38-125 lists the type of actions possible by each filter type.

Table 38-125.Filter actions by filter type

Action	MAC / VLAN / Ether-type / S-Tag / Cloud Switch Filters	PE Quad Hash	Flow Director	Hash
LAN/PE	LAN	PE	LAN	LAN
Receive Queue index / Context ID	Queue index within the VSI	QPN + LAN queue index within the VSI	Queue index within the VSI	Queue index within the TC range of queues
Accept / Reject			Yes	
Increment Stat Counter (one of 512 GLQF_PCNT)			Yes	
Report Filter ID		QPN	Filter ID	Hash signature
Report selected bytes from the receive packet			Programmable 4 or 8 bytes	

38.21.1.1.3 Receive Queue Index

As indicated in the previous section, one of the filter's action is the packet classification to LAN queues. The LAN queue index defined by the classification filters is a relative index within a specific VSI that is defined by the embedded switch. Assuming a classification filters defines a queue index 'n' then:

- VSI that is defined by the VSILAN_QTABLE (VSILAN_QBASE.VSIQTABLE_ENA = 1b). The queue index in the PF space equals to: VSILAN_QTABLE[n/2].QINDEX_0 for even 'n' and VSILAN_QTABLE[(n-1)/2].QINDEX_1 for odd 'n'.
- VSI that is defined by the VSILAN_QBASE (VSILAN_QBASE.VSIQTABLE_ENA = 0b. The queue index in the PF space equals to VSILAN_QBASE.VSIBASE + 'n'.
- VF queues — The mapping of the previous queue indexes to relative indexes in the VF space is not a direct mapping. The mapping is defined by the queue indexes in the PF space that are programmed in the VSILAN_QTABLE and the VPLAN_QTABLE registers. For example, assuming a VF with 4 queues in a single VSI with the following settings: VPLAN_QTABLE[0:3]=A,B,C,D (while A...D are the queue indexes in the PF space); VSILAN_QTABLE[0].QINDEX_0,QINDEX_1 = C,D and VSILAN_QTABLE[1].QINDEX_0,QINDEX_1=A,B. Then receiving a packet to queue 0 of the VSI is mapped to queue C in the PF space which is queue index 2 in the VF space.

The hash filter is a little bit more complex, while the queue index is a relative index within a specific range of queues (usually related to a TC) of a VSI. See Section 38.21.3.6.

Following are some exception rules for the receive queue index:

- If a frame does not match any of the classification filters (with a queue action), the frame is directed to the default queue (queue zero of the VSI).
- If the packet has a MAC error (reflected by the RXE flag in the receive descriptor), the packet is directed to the default queue (queue zero of the VSI that is defined by the PRT_SBPVSI register). Note that a packet with MAC error is posted to the host only if enabled by the SBP flag in the PRT_SBPVSI register.
- The receive queue index that defines an invalid queue causes the dropping of the matched received packets. Such packets are counted by the GLV_REPC counter of the VSI.
- The following sub-bullets list the cases on which a queue index is considered as invalid:
  - A VSI that is defined by VSILAN_QTABLE is limited to 16 queues, which is the size of this table. Therefore, a queue index that equals or is larger than 16 is considered invalid.
  - The queues in a VSI that is defined by VSILAN_QTABLE are enabled by the QINDEX_0 and QINDEX_1 parameters. A queue index that points to a VSILAN_QTABLE entry with a value of 0x7FF is considered invalid.
  - A VSI that is defined by VSILAN_QBASE.VSIBASE is limited only by the PF queue space. A queue index that exceeds the PF space is considered invalid.

38.21.1.1.4 Initialization

This section lists all registers associated with the classification filters directly or indirectly and its required initialization. Note that some of the text in this section might not be clear, lacking a complete description of the registers. This description is given in the following sections as well as in the Programming Interface chapter. The initialization sequence should be executed in the following order shown. Also described in the sub-sections that follow:

- Function Private Memory Allocation
- Queue Allocation
- Static Classification Filters Registers
- Dynamic Classification Filters Registers

Function private memory allocation

The PE quad hash filters are stored in the Function Private Memory (FPM) while part of them are fetched to the on-chip cache. These registers should be initialized per PF prior to any settings of the LAN queues and before any enablement of the PE filters.

Queue allocation

The LAN queue should be allocated before packets are directed to these queues. If this is not done, the packets are dropped.

PE quad hash filters as well as TCP port filters should be programmed before matched packets are expected. Receiving packets with non-matched filters are directed to the LAN queues of the function. The same rule applies for the DDP context table.

Static classification filters registers

Table 38-126.Static Classification Filters Registers Initialization

Register	Initialization Comments
PRTQF_CTL_0	Symmetric hash enablement is loaded from the NVM.
GLQF_CTL	Global classification parameters are loaded from the NVM.
VSIQF_CTL	Filtering enablement per VSI is programmed by the Add/Update VSI admin commands.
PFQF_CTL_0, PFQF_CTL_2	PF classification parameters should be initialized before the PE and DDP contexts programming.
VPQF_CTL	VF PE and DDP filters parameters should be initialized after the PFQF_CTL_0 register of the PF and before the VF enablement and the PE and DDP contexts programming.
GLQF_SWAP and GLQF_HSYM	Swap registers and symmetric hash registers are loaded from the NVM.
GLQF_NETKEY_PIT	The UDP network key structure is programmed by the Set Network Key Structure admin commands before any of the Add Tunneling UDP admin command are initiated.
PRTQF_FLX_PIT	The field vector registers related to the flexible payload are either loaded from the NVM or programmed by the PFs at driver initialization time.
PRTQF_FD_MSK	Mask registers for the input set of the flexible payload for the FD filter is loaded from the NVM or programmed by the PFs at driver initialization time.
VSIQF_TCREGION	The queue regions for 8 x TCs are programmed by the Add/Update VSI admin commands.
PFQF_FDALLOC	FD filter table allocation to PFs is loaded from the NVM.

Dynamic classification filters registers

Table 38-127.Dynamic Classification Filters Registers Initialization

Register	Initialization Comments
PFQF_HENA and VFQF_HENA	PCTYPE enablement for the Hash filters are programmed by the function (PF or VF) at run time.
PFQF_HREGION and VFQF_HREGION	Override the TC queue regions for specific PCTYPEs registers are programmed by the function (PF or VF) at run time.
PFQF_HKEY, PFQF_HLUT, VFQF_HKEY and VSIQF_HLUT	The hash key and LUT registers are programmed by the function (PF or VF) at run time.
GLQF_APBVT	The TCP port number table is programmed internally at run time by the PE.

38.21.1.2 Packet Types and Input Set

The 10 GbE controller parses the first 480 bytes of the received packets. The hardware identifies the protocol layers upon which packet classification types (PCTYPEs) are defined. The supported PCTYPEs are listed in the Table 38-128. For flow director, a second PCTYPE is derived from the original PCTYPE according to Table 38-129. The fields used for the classification filters (named as input set) are defined per filter per PCTYPE as listed in Table 38-128. The table lists a shared column for the input set of the FD filter and PE quad hash. The following section describes which PCTYPEs are relevant to each filter. See Section 38.21.3.8.3 for PE quad hash input set.

Table 38-128.Packet Classifier Types and its Input Sets (Sheet 1 of 2)

PCTYPE	PCTYPE Description	Hash Input Set	FD Input Set[1]
Reserved			
0 ... 25	Reserved for future use	-	-
Tunneling			
26	Geneve OAM	Source Outer UDP Port, VNI	Source Outer UDP Port, VNI
27	VXLAN-GPE OAM	Source Outer UDP Port, VNI	Source Outer UDP Port, VNI

Table 38-128.Packet Classifier Types and its Input Sets (Sheet 2 of 2)

PCTYPE	PCTYPE Description	Hash Input Set	FD Input Set[1]
[Inner] IPv4 packets			
28	Reserved		
29	NonF Unicast IPv4, UDP	IP4-S, IP4-D, UDP-S, UDP-D	IP4-S, IP4-D, UDP-S, UDP-D
30	NonF Multicast IPv4, UDP	IP4-S, IP4-D, UDP-S, UDP-D	IP4-S, IP4-D, UDP-S, UDP-D
31	NonF IPv4, UDP	IP4-S, IP4-D, UDP-S, UDP-D	IP4-S, IP4-D, UDP-S, UDP-D
32	NonF IPv4, TCP, SYN, no ACK	IP4-S, IP4-D, TCP-S, TCP-D	IP4-S, IP4-D, TCP-S, TCP-D
33	NonFIPv4, TCP	IP4-S, IP4-D, TCP-S, TCP-D	IP4-S, IP4-D, TCP-S, TCP-D
34	NonFIPv4, SCTP	IP4-S, IP4-D, SCTP-S, SCTP-D, SCTP-Verification-Tag	IP4-S, IP4-D, SCTP-S, SCTP-D, SCTP-Verification-Tag
35	NonFIPv4, Other	IP4-S, IP4-D	IP4-S, IP4-D
36	Frag IPv4	IP4-S, IP4-D	IP4-S, IP4-D
37	Reserved	-	-
[Inner] IPv6 packets			
38	Reserved		
39	NonF Unicast IPv6, UDP	IP6-S, IP6-D, UDP-S, UDP-D	IP6-S, IP6-D, UDP-S, UDP-D
40	NonF Multicast IPv6, UDP	IP6-S, IP6-D, UDP-S, UDP-D	IP6-S, IP6-D, UDP-S, UDP-D
41	NonF IPv6, UDP	IP6-S, IP6-D, UDP-S, UDP-D	IP6-S, IP6-D, UDP-S, UDP-D
42	NonF IPv6, TCP, SYN, no ACK	IP6-S, IP6-D, TCP-S, TCP-D	IP6-S, IP6-D, TCP-S, TCP-D
43	NonFIPv6, TCP	IP6-S, IP6-D, TCP-S, TCP-D	IP6-S, IP6-D, TCP-S, TCP-D
44	NonFIPv6, SCTP	IP6-S, IP6-D, SCTP-Verification-Tag	IP6-S, IP6-D, SCTP-Verification-Tag
45	NonFIPv6, Other	IP6-S, IP6-D	IP6-S, IP6-D
46	Frag IPv6	IP6-S, IP6-D	IP6-S, IP6-D
47	Reserved	-	-
L2 Packet Types			
52 ... 62	Reserved for future use	-	-
63	L2 packet type (1)	L2 Ether-type	L2 Ether-type
Note 1: L2 packet types are identified only for the last Ether-type in the outer L2 header.			

Table 38-129.Flow Director PCTYPE Translation (Sheet 1 of 2)

PCTYPE	PCTYPE Description	Flow Director PCTYPE[1]
Reserved		
0 ... 27	Reserved for future use	0 ... 27
[Inner] IPv4 packets		
28	Reserved	28
29	NonF Unicast IPv4, UDP	31
30	NonF Multicast IPv4, UDP	31
31	NonF IPv4, UDP	31
32	NonF IPv4, TCP, SYN, no ACK	33
33	NonF IPv4, TCP	33
34	NonF IPv4, SCTP	34
35	NonF IPv4, Other	35

Table 38-129.Flow Director PCTYPE Translation (Sheet 2 of 2)

PCTYPE	PCTYPE Description	Flow Director PCTYPE[1]
36	Frag IPv4	36
37	Reserved	37
[Inner] IPv6 packets		
38	Reserved	38
39	NonF Unicast IPv6, UDP	41
40	NonF Multicast IPv6, UDP	41
41	NonF IPv6, UDP	41
42	NonF IPv6, TCP, SYN, no ACK	43
43	NonFIPv6, TCP	43
44	NonFIPv6, SCTP	44
45	NonFIPv6, Other	45
46	Frag IPv6	46
47	Reserved	47
L2 Packet Types		
52 ... 62	Reserved for future use	52 ... 62
63	L2 Payload	63

Notes:

1. The FDPCTYPE uses the inputs from the original PCTYPE configuration (GLQF_PTYPE and GLQF_PTYPE_ENA) and to input-set and mask selection (PRTQF_FD_INSET, PRTQF_FD_MSK, GLQF_FD_MSK), so the translation should be to a valid PCTYPE in the original list.

38.21.1.3 Filter's Input Set

The input set is defined per filter type as follows:

- Layer 2 filters that are used by the switch logic. These include: MAC filters; VLAN filters; S-Tag and Ether-type. The input set for these filters is defined in the Internal Switch chapter.
- PE APBVT. This is a non flexible filter and its input set is the destination TCP/UDP port number.
- The input set for the FD filter is flexible. Its input set for the 64 global PCTYPEs has two components:
  - A pre-defined byte stream is listed in Table 38-128.
  - A flexible input set defined by the PRTQF_FD_FLXINSET registers. A PRTQF_FD_FLXINSET register 'n' matches PCTYPE 'n' while each bit 'i' in the 8 bit INSET field in these registers enables word 'i' in the flexible field vector. If only a portion of a 16 bit word is required for the input set, the non-relevant bits can be masked out by the PRTQF_FD_MSK registers.
- The input set for the hash filter for the 64 global PCTYPEs is listed in Table 38-128.
- The input set for the PE quad hash filter for the 8 PE PCTYPEs is listed in Table 38-141 (programmed by the GLQF_PE_INSET registers that are loaded from the NVM).

38.21.1.4 Flexible Payload

The 10 GbE controller supports a flexible payload of 8 words (16 bytes) that are extracted from the payload of the packet. These 8 words can be extracted from up to 3 offsets within the payload. Payload is defined as follows:

- If a packet is identified as a L2 packet, payload is anything after the last Ether-type.
- If a packet is identified as a L3 packet, payload is anything after the inner identified L3 header.
- If a packet is identified as a L4 packet, payload is anything after the inner identified L4 header.

The flexible payload is stored in a field vector. This vector can be used by the FD filter as described in the following sections.

The 10 GbE controller parses the packets to protocol layers. For each protocol layer (Layers 2...4), Fort Park enables filtering on flexible payload that are defined by the GLQF_ORT[33:35] registers and the PRTQF_FLX_PIT registers. The GLQF_ORT[33:35] registers are loaded from the NVM and the PRTQF_FLX_PIT registers can be either loaded from the NVM or programmed by the PF at driver initialization time. The programming of the GLQF_ORT[33:35] registers are listed in Table 38-130. The programming of the PRTQF_FLX_PIT registers is listed in Table 38-131.

Register Initialization — All the registers related to flexible payload should be initialized before enabling the transmit and receive queues: PRTQF_FLX_PIT, PRTQF_FD_FLXINSET and PRTQF_FD_MSK registers.

Table 38-130.Flexible Payload (GLOQF_ORT[33:35] Registers)

Protocol Layer
L2 payload (starts after the last Ether-type[1]). The L2 payload is active only if L3 protocol is not found. The L2 payload is directed to the field vector by the PRTQF_FLX_PIT[0:2] registers. These are indicated by the GLQF_ORT[33] register. The GLQF_ORT[33] registers loaded from the NVM with the following values: FLX_PAYLOAD = 1b; PIT_INDX = 0x0; FIELD_CNT = 0x3
L3 payload (starts after the inner identified L3 header). The L3 payload is active only if L4 protocol is not found. Supported L3 protocols are: IPv4 (including any optional headers) and IPv6 (including all supported extension headers). The L3 payload is directed to the field vector by the PRTQF_FLX_PIT[3:5] registers. These are indicated by the GLQF_ORT[34] register. The GLQF_ORT[34] register is loaded from the NVM with the following values: FLX_PAYLOAD = 1b; PIT_INDX = 0x3; FIELD_CNT = 0x3
L4 payload (starts after the inner identified L4 header). Supported L4 protocols are: UDP, TCP, SCTP, ICMP, ICMPv6. The L4 payload is directed to the field vector by the PRTQF_FLX_PIT[6:8] registers. These are indicated by the GLQF_ORT[35] register. The GLQF_ORT[35] register is loaded from the NVM with the following values: FLX_PAYLOAD = 1b; PIT_INDX = 0x6; FIELD_CNT = 0x3

Notes:

1. An ARP packet is a special case on which the payload starts after the entire ARP header.

38.21.1.4.1 PRTQF_FLX_PIT Programming

As previously indicated, the PRTQF_FLX_PIT registers map the protocol layers payload (protocol layers = L2, L3, L4) to the field vector. During normal operation only one protocol layers payload is meaningful so the entire flexible payload in the field vector is available for each of them. The fields of the PRTQF_FLX_PIT registers are described in the Programming Interface chapter.

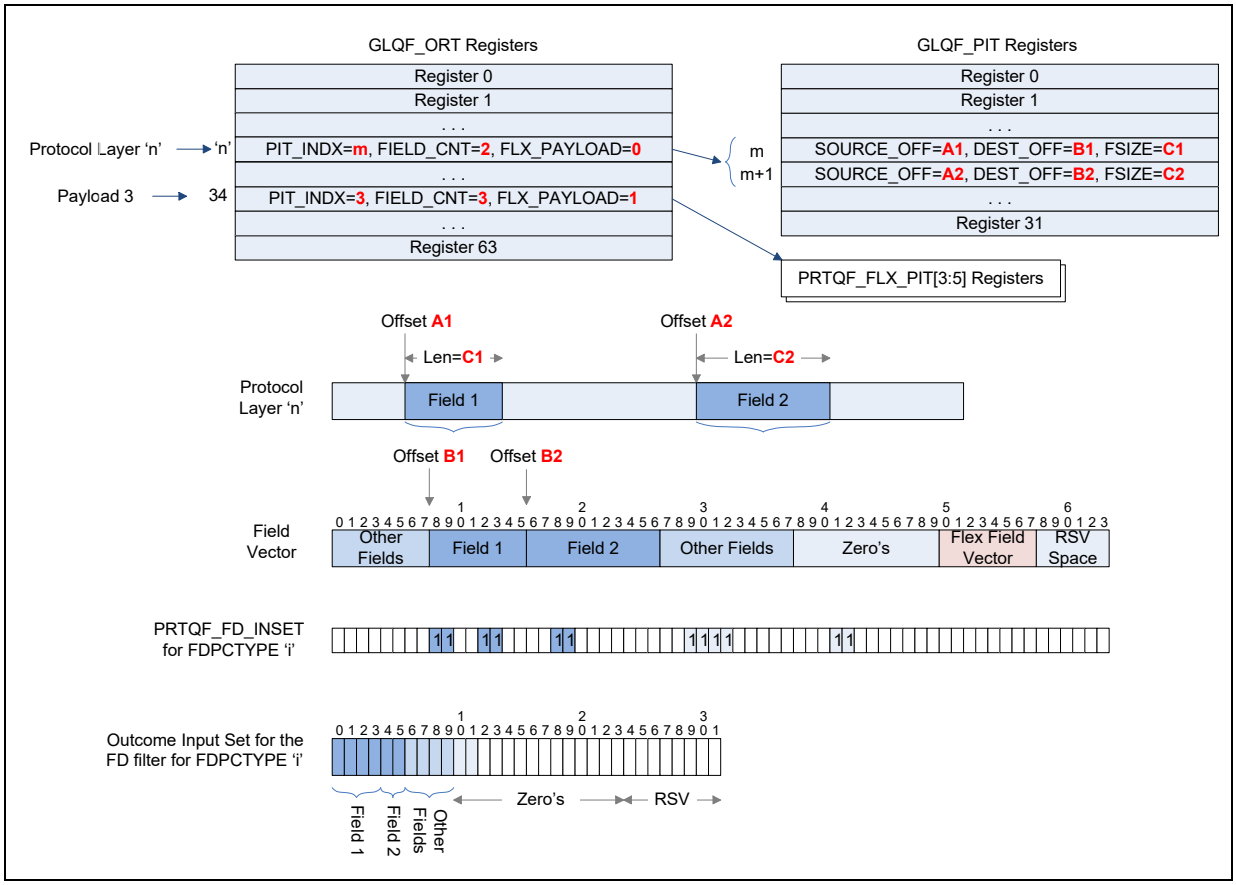
Table 38-131.PRTQF_FLX_PIT Registers' Field

Field	Description
SOURCE_OFF	Source word offset in the mapped protocol layer starting from its beginning. Setting source offset rules: • The SOURCE_OFF plus FSIZE should not exceed byte 480 of the packet. • Must be programmed in ascending order: Current Offset >= previous offset + previous FSIZE. • The previous rule applies for all entries, including non-used ones.
DEST_OFF	Destination word offset in the field vector. The destination offset can be set to 50...57 matching offset 0...7 in the flexible field vector, respectively. While DEST_OFF plus FSIZE must not be greater than 58. For non-used registers the DEST_OFF must be set to 0x63 (outside the range for active entries).
FSIZE	Field size defined in word units. For non-used registers the FSIZE must be set to 0x1.

38.21.2 Protocol Layers Filter's Fields and Packet Classifier Types

The 10 GbE controller parses the first 480 bytes of the received packets. The hardware identifies the protocol layers in this range and constructs a field vector used for all the filters. Each filter uses a different input set taken from the shared field vector. The protocol layers; field vector; filter fields as well as the input set are defined in the following subsections. The fields extracted from the protocol layers to the field vector are pre-defined. It is programmed by the GLQF_ORT and GLQF_PIT registers as well as the PRTQF_FLX_PIT registers. The default setting of the GLQF_ORT and GLQF_PIT registers loaded from the NVM is described in the sections that follow. The input set is defined per-filter per-packet classifier type as shown in the following sections. See Figure 38-1 for an input set example of the FD filter and the registers used to construct the input set.

Figure 38-1. Input Set Example for an FD Filter



38.21.3 ORT PIT Register Settings

This section defines the default hardware and NVM setting of the GLQF_ORT and GLQF_PIT registers.

Table 38-132.GLQF_ORT Register Settings (Sheet 1 of 2)

Reg. Indx	Protocol Layer	FLX_PAY LOAD [bit 7]	FIELD_ CNT [bit 5:6]	PIT_Indx [bits 4:0]	Reg. Indx	Protocol Layer	FLX_PAY LOAD [bit 7]	FIELD_ CNT [bit 5:6]	PIT_Indx [bits 4:0]
0	1588 ETyp 0x88F7	0	0	0	32	Inner cloud MAC no VLAN	0	1	24
1	FIP ETyp 0x8914	0	0	0	33	L2 Payload	0	0	0
2	VXLAN/VXLAN-GPE Tunnel	0	3	26	34	L3 Payload	0	0	0
3	Geneve Tunnel	0	3	26	35	L4 Payload	0	0	0
4	LLDP ETyp 0x88CC	0	0	0	36	Reserved	0	0	0
5	ECP EType	0	0	0	37	MAC	0	1	0
6	UDP Tunnel	0	1	21	38	Reserved	0	1	9
7	MPLS - 0x8847	0	0	0	39	STAG - 0x88A8	0	1	10
8	IEEE 802.1X ETyp 0x888E	0	0	0	40	External VLAN (NVM)	0	1 (0)	10 (0)

Table 38-132.GLQF_ORT Register Settings (Sheet 2 of 2)

Reg. Indx	Protocol Layer	FLX_PAY LOAD [bit 7]	FIELD_ CNT [bit 5:6]	PIT_Indx [bits 4:0]	Reg. Indx	Protocol Layer	FLX_PAY LOAD [bit 7]	FIELD_ CNT [bit 5:6]	PIT_Indx [bits 4:0]
9	ARP Etype 0x0806	0	1	12	41	VLAN	0	1	1
10	GRE Key No XSUM	0	1	18	42	Reserved	0	0	0
11	GRE Key With XSUM	0	2	19	43	Reserved	0	0	0
12	IPv4	0	3	2	44	Reserved	0	0	0
13	IPv6	0	2	4	45	Reserved	0	0	0
14	Reserved	0	2	7	46	Reserved	0	0	0
15	GRE No Key	0	1	19	47	Reserved	0	0	0
16	Teredo	0	1	21	48	Reserved	0	0	0
17	TCP	0	1	6	49	Reserved	0	0	0
18	UDP (NVM)	0 (0)	1 (1)	6 (16)	50	Reserved	0	0	0
19	SCTP (NVM)	0 (0)	1 (1)	6 (16)	51	Reserved	0	0	0
20	ICMPv4 (NVM)	0 (0)	1 (1)	6 (17)	52	External IPv4	0	1	14
21	Reserved	0	0	0	53	External IPv6	0	1	15
22	Reserved	0	0	0	54	Reserved	0	0	0
23	ICMPv6 (NVM)	0 (0)	1 (1)	6 (17)	55	Reserved	0	0	0
24	IPv6 Extensions	0	0	0	56	Reserved	0	0	0
25	IPv6 Dest/ Routing Extensions	0	0	0	57	Reserved	0	0	0
26	FC-RDY/FC-Read (NVM)	0 (0)	0 (1)	0 (11)	58	Reserved	0	0	0
27	FC DATA	0	1	11	59	Reserved	0	0	0
28	FC VFT Header	0	0	0	60	Reserved	0	0	0
29	FC RSP	0	1	11	61	Reserved	0	0	0
30	FC-Other (NVM)	0 (0)	0 (1)	0 (11)	62	Reserved	0	0	0
31	Inner Cloud MAC+VLAN	0	2	24	63	Last EType (NVM)	0 (0)	0 (1)	0 (13)

Table 38-133.GLQF_PIT Registers Settings from NVM (Sheet 1 of 2)

Reg. Indx	Protocol Layer	DEST_ OFF [15:10]	FSIZE [9:5]	SOURCE_ OFF [4:0]	Reg. Indx	Protocol Layer	DEST_ OFF [15:10]	FSIZE [9:5]	SOURCE_ OFF [4:0]
0	MAC	0	6	0	16	Ext-IPv6	50	8	12
1	VLAN	8	1	0	17	UDP / SCTP (NVM)	0 (29)	0 (4)	0 (0)
2	IPv4	9	2	0	18	ICMPv4/v6 (NVM)	0 (29)	0 (2)	0 (0)
3	TTL and Protocol Word	11	4	4	19	GRE No XSUM	42	4	0
4	IPv4	21	2	8	20	GRE with XSUM	42	2	0
5	IPv6	9	8	0	21	GRE with XSUM	44	2	4
6	IPv6	17	12	8	22	UDP-Tunnel	42	2	0
7	TCP	29	8	0	23	UDP-Tunnel	0	0	0

Table 38-133.GLQF_PIT Registers Settings from NVM (Sheet 2 of 2)

Reg. Indx	Protocol Layer	DEST_OFF [15:10]	FSIZE [9:5]	SOURCE_OFF [4:0]	Reg. Indx	Protocol Layer	DEST_OFF [15:10]	FSIZE [9:5]	SOURCE_OFF [4:0]
8	Reserved	9	1	0	24	UDP-Tunnel	0	0	0
9	Reserved	10	1	6	25	Inner MAC	39	3	0
10	Reserved	6	2	0	26	Inner VLAN	7	1	7
11	STAG	6	1	0	27	VXLAN / Geneve / VXLAN-GPE	42	1	0
12	FCRSP/ FCDATA/ FCANY	11	12	0	28	VXLAN / Geneve / VXLAN-GPE	43	1	4
13	ARP	9	14	0	29	VXLAN / Geneve / VXLAN-GPE	44	2	6
14	Last EtherType	49	1	0	30	RSV	0	0	0
15	Ext-IPv4	56	2	8	31	RSV	0	0	0

38.21.3.1 Protocol Layers (Type Rule Bus)

Protocol layers are portions of the packet that are associated with specific protocols as listed in the Table 38-134. The protocol layers table includes L2 layers (like LLDP), L3 protocols (like IP header), L3 protocols (like TCP header) and extracted flags from the packet (like IP fragmentation indication and TCP URG). The protocol layers and the flags are used to construct the field vector and define packet types as explained in the proceeding sections.

Table 38-134.Protocol Layers Table (Type Rule Bus) (Sheet 1 of 2)

Indx	Protocol Layer	Indx	Protocol Layer
0	L2 payload for ETYPE = 0x88F7[1]	32	Inner cloud MAC without VLAN
1	L2 payload for ETYPE = 0x8914 (FIP packet)[1]	33	L2 Payload
2	VXLAN / VXLAN-GPE key (Key 0)	34	L3 Payload
3	Geneve key (Key 1)	35	L4 Payload
4	L2 payload for ETYPE = 0x88CC (LLDP) [1]	36	L2 header 0 - PreL2 header
5	L2 payload for ETYPE = ECP ETyp (1)	37	L2 header 1 - MAC Addresses
6	Place holder for Generic UDP tunneling (key 2)	38	L2 header 2 - Reserved
7	NSH ETYPE = 0x894F	39	L2 header 3 - S-Tag
8	L2 payload for ETYPE = 0x888E (IEEE 802.1X)[1]	40	L2 header 4 - External VLAN (in the outer MAC header in case of MAC in UDP or MAC in GRE packets)
9	L2 payload for ETYPE = 0x0806 (ARP)[1]	41	L2 header 5 - VLAN (Inner one in case of double VLANs. Both are in the outer MAC header in case of MAC in UDP or MAC in GRE packets)
10	GRE key[1]	42	L2 header 6 - CTS
11	Reserved	43	L2 header 7 - Top MPLS Header
12	IPv4 layer (inner IPv4 in case of tunneling)	44	L2 header 8 - Reserved
13	IPv6 layer (inner IPv6 in case of tunneling)	45	L2 header 9 - Reserved
14	Reserved	46	Reserved
15	GRE no Key (1)	47	Reserved
16	Teredo protocol layer	48	IP header is fragmented
17	TCP protocol layer	49	Inner IPv4 header includes IP options
18	UDP protocol layer	50	TCP header includes options

Table 38-134.Protocol Layers Table (Type Rule Bus) (Sheet 2 of 2)

Indx	Protocol Layer	Indx	Protocol Layer
19	SCTP protocol layer	51	Reserved
20	ICMP protocol layer	52	External IPv4 layer
21	Reserved	53	External IPv6 layer
22	Reserved	54	Reserved
23	ICMPv6 protocol layer	55	TCP SYN flag
24	IPv6 Extension protocol layer(s) other than destination/ routing (in case of tunneling it is an 'OR' function of the inner and outer headers)	56	Reserved
25	IPv6 Extensions destination/routing (in case of tunneling it is an 'OR' function of the inner and outer headers)	57	OAM flag in VxLAN-GPE header
26	FC RDY or RD/WR Request (R_CTL = 0x05 or 0x06)	58	OAM flag in Geneve header
27	FC Data (_RCTL = 0x01)	59	Reserved
28	FC VFT Header	60	Reserved for flexible flag
29	FC RSP (R_CTL = 0x07)	61	Unicast Destination MAC Address indication
30	FC other	62	Multicast Destination MAC Address indication
31	Inner Cloud MAC + VLAN	63	Broadcast Destination MAC Address indication (multiplexed with last EtherType internally)

Notes:

1. With tunneled packets, these flags indicate the existence of matched protocol layers in the outer L2 header.

38.21.3.2 Filter Field

Filter field is a consecutive byte stream extracted from a protocol layer to an internal field vector that might be used for the packet classification. A field size can be up to 30 bytes defined in 2-byte granularity.

38.21.3.3 Field Vector

The field vector is a 128-byte string composed of multiple fields that are extracted from identified protocol layers of the packet. The field vector is initialized to zero before it is filled by received packet fields. Out of the 128 bytes of the field vector only the first 116 bytes are available for packet's fields while the other bytes are reserved for internal parameters. Among these parameters are the function ID, VSI index, identified protocol layers and protocol flags. Table 38-135 shows the fields that are extracted from the packets (according to the device setting in the default NVM image). The fields are posted to the field vector in a mixed big/little endian ordering while the first byte on the wire is posted in the MS byte of the first word number of the field. For example, the first byte on the wire of the destination MAC address (the MS byte) is posted to high byte of word number 0 in the field vector. Note that mutual exclusive protocol layers can share the same space in the field vector (like TCP and UDP).

In general, mutually exclusive protocol layers can share the same words in the field vector (like TCP and UDP). In some special cases it might make sense to share the same words in the filter for non mutually exclusive protocol layers. In such a case, the most recent detected protocol layer would step over previous protocols layer's words that are defined in the same location in the field vector.

Table 38-135.Default Field Vector Table

Word Num	Protocol Layers			
	L2 Protocol Layers			
0:2	Destination MAC Address (in outer or single L2 header)			
3:5	Source MAC Address (in outer or single L2 header)			
6	Reserved	S-Tag	Reserved	
7		-		VLAN Tag (in the inner L2 header) see Table 38-136
8	Inner or single VLAN Tag (in outer or single L2 header) see Table 38-136			
	L3 Protocol Layers			
	Inner or single IPv4	Inner or single IPv6		ARP
9	First two word of the IPv4 Header (Version + TOS + length)	First 4 words of the IPv6 Header (up to including the Hop Limit)		Hardware type ... Sender IP address (first 28 bytes of the header)
10				
11:12	TTL+ Protocol word, Header checksum			
13:14	IPv4 source address	IPv6 Source address		
15:20	0x00			
21:22	destination IP address	IPv6 Destination address		
23:26	0x00		0x00	0x00
27:28	0x00		0x00	0x00
	L4 Protocol Layers			
	TCP	UDP	SCTP	ICMPv6
29:30	First 16 bytes of the TCP Header	First 8 bytes of the UDP Header	First 8 bytes of the SCTP Header	Words 0, 1 of the header
31:32				0x00
33:36		0x00	0x00	
37:38	0x00	0x00	0x00	
	Tunneling Layers and Flexible Payload + Last EtherType			
	Non Tunneled	IP in IP Tunneling	UDP Tunneling	GRE Tunneling
39:41	0x00	0x00	Destination MAC Address (in the Inner L2 header). Relevant only for MAC in UDP or MAC in GRE.	
42:43	0x00	0x00	Tunneling UDP Source and Destination Port numbers	GRE ver and Protocol Type
44:45	0x00	0x00	Tunneling UDP Keys	GRE Key
46	0x00	0x00	Geneve / VXLAN Version (word 0 of the header)	
46:48	0x00	0x00		0x00
49	Last EtherType (in case of tunneled packets it is the last EtherType in the outer L2 header)			
50:57	Flexible Payload programmed by the PRTQF_FLX_PIT and GLQF_ORT[33:35] registers	Outer or single destination IP address Note that if this field is required then the Flexible payload must not be enabled or else the flexible payload will override this field.		
	Other Internal Fields			
58:63				

Table 38-136.Detailed L2 Tags in the Field Vector (Assuming all L2 Tags are VLANs)

Description	Word 7: VLAN Tag (in the inner L2 header)	Word 8: Inner or single VLAN Tag (in outer or single L2 header)
Non-tunneled received packet with a Single VLAN	X	VLAN
Non-tunneled receive packet with double VLANs	X	Inner VLAN
Tunneled received packet with a single VLAN in the outer L2 header and no VLAN in the inner L2 header	X	VLAN in the outer L2 header
Tunneled received packet with double VLAN in the outer L2 header and no VLAN in the inner L2 header	X	Inner VLAN in the outer L2 header
Tunneled received packet with a single VLAN in the outer L2 header and VLAN in the inner L2 header	VLAN in the inner L2 header	VLAN in the outer L2 header
Tunneled received packet with double VLAN in the outer L2 header and VLAN in the inner L2 header	VLAN in the inner L2 header	Inner VLAN in the outer L2 header

38.21.3.4 Layer 2 Classification Filter

The 10 GbE controller supports L2 filters (MAC / VLAN / L2 EtherType) in the embedded switch. These filters act on the following layers: S-Tag; MAC address; VLAN and EtherType. These filters define a VSI and can define also a unique LAN queue within that VSI. The LAN queue is enabled by the *ToQueue* flag and the queue is defined by the queue number for each filter. These parameters are programmed by admin commands. The filters can be defined as the second or fifth priority filters for receive queue classification as follows:

Table 38-137.Second or Fifth Priority Filters for Receive Queue Classification

Priority #	Filters defined by the following Admin command(s)
2	Add Control Packet Filter
5	Add MAC, VLAN pair; Add S-tag; Add Cloud Filters

Design limitation — Packet match to multiple L2 filters with active *ToQueue* flag that direct the packet to different queues is out of scope. Prioritization between the L2 filters in this case is defined by internal micro architecture rules that are not exposed. Therefore, software cannot rely on a specific arbitration and should avoid such cases for deterministic response.

The EtherType filter and MAC/VLAN filter affect on queue routing are enabled per PF and its VFs by the PFQF_CTL_0 registers.

38.21.3.5 Tunneling Fields

The 10 GbE controller supports tunneled packet formats including tunneled IP address, UDP Teredo, MAC in UDP, IP in GRE and MAC in GRE. These tunneling packets are supported for transmit and receive data processing offloads. When it comes to packet classification, it is based only on the encapsulated packet (exactly the same as non tunneled packets). Following are the supported tunneled packets:

IP Tunneling

For a non-tunneled IP packet, there is only one IP header that is used for the packet classification. For tunneling, The inner IP header is used for packet classification while the outer IP header is used for the switch filters.

Teredo

Teredo are supported protocol layers for the classification filtering. See Section 38.21.3.5.1 for more details. Teredo support is mutually exclusive with MAC in UDP and GRE tunneling.

MAC in UDP

MAC in a UDP packet is identified by the destination UDP port number. The UDP ports used for MAC in UDP are programmed by the same admin commands as the Teredo UDP ports as described in Section 38.21.3.5.1. Classification to receive queues is based on the encapsulated packet according to its structure. MAC in UDP support is mutually exclusive with Teredo and GRE tunneling.

GRE Tunneling

GRE tunneling is identified by the last EtherType in the outer L2 header. IP in GRE or MAC in GRE are identified by the *Protocol Type* field in the GRE header. MAC in UDP support is mutually exclusive with Teredo and MAC in UDP.

The previous excerpt states that packets are classified to receive queues based on the encapsulated packet fields. Packets with Geneve header or VXLAN-GPE with an active OAM flag are the exception to this rule. These packets are identified as unique PCTYPEs as listed in Table 38-128. Such packets with an active OAM flag can be directed to receive queue zero of the VSI (default queue) if programmed by software. In order to get this functionality, the 10 GbE controller should not process these packets by the hash and FD filters. Software can do so by disabling the OAM PCTYPEs in the xxQF_HENA registers of the function (affecting the hash filter) and disabling the OAM PCTYPEs in the GLQF_FDENA registers (affecting the FD filter) for the device.

38.21.3.5.1 Unique UDP Port Filters

There are some destination UDP port numbers that have unique meaning. These ports include 1588 packet identification and Teredo tunneling. The 10 GbE controller supports the following port numbers:

- Two global port numbers for 1588 equal to 319 and 320.
- A Tunneling UDP table with 16 global port numbers for Teredo tunneling or MAC in UDP tunneling that are programmed at run time by admin commands described later in this section. The tunneling UDP port numbers should not be one of the values reserved for 1588 and should not be set to 0x0000 or 0xFFFF.

Note that the 10 GbE controller does not process (offload) the checksum field in the tunneling UDP header.

The tunneling UDP table is a global resource available for all PFs on a first come first served policy. This table defines the port number and the tunneling type as listed in Table 38-139. The PFs can add an entry or delete an entry by admin commands. Any add or remove command is acknowledged by a completion command that provides a status of the requested action and an updated status of the table resources.

A table entry is added by the first PF that program it. Any additional PF that programs the same UDP port number just register to the existing entry. Once a table entry is programmed, it affects received packets for all PFs on all LAN ports (regardless of which PFs are registered to that UDP port number).

The admin commands used to program the tunneling headers are listed in Table 38-138 and detailed in the following sub-sections:

Table 38-138.Tunneling UDP Admin Commands

Name	Opcode	Ref	Admin Command Type
Add Tunneling UDP	0x0B00	Section 38.21.3.10.1	Direct
Remove Tunneling UDP	0x0B01	Section 38.21.3.10.2	Direct

Table 38-139.Supported Tunneling Types

Tunneling Type	Tunneling UDP Protocol Type"	Fixed Header Length [bytes]	Variable header Length	Network Key
Teredo	0x10 (next protocol is IP)	0x8	No	No
VXLAN	0x00 (next protocol is MAC)	0x10	No	Yes, see Section , "Network Key Structure of Tunneling Header"
Geneve	0x01 (next protocol is the optional Geneve header length)	0x10	Defined by bits 5:0 in byte 0x8 in the header	Yes, see Section , "Network Key Structure of Tunneling Header"
Place holder for Generic MAC in UDP	0x02 (next protocol is MAC)	Not Defined Yet	No	Yes, see Section , "Network Key Structure of Tunneling Header"
Reserved	Else	n/a	n/a	n/a

Network Key Structure of Tunneling Header

There are 4 global network keys, shared for all LAN ports. The network keys are used to identify a cloud tenant. The 10 GbE controller supports a programmable structure for these network keys that can be programmed by the PFs, using the Program Network Key Structure admin command. In NVGRE and Geneve headers, it is defined as a 3-byte field at a pre-defined offset in the header. The matched network key structure parameters for the supported tunneling headers are listed in Table 38-140:

Table 38-140.Network Key Structure of Tunneling Header

Key Index	Tunneling Header	Key Offset	Key Length
0	VXLAN	12	3
1	Geneve	12	3
2	Place holder for generic MAC in UDP	offset 1: 8 <= Offset 1 < 32 offset 2: 8 + Offset 1 + Length 1<= Offset 2 < 64 - Length 2	0 <= Length 1, Length 2 <= 6 (*) Length 1 + Length 2 <= 10 (*) (*) assuming word aligned keys
3	NVGRE	4	3

38.21.3.6 Hash Filter

The hash filter is a mechanism to statistically distribute received packets into several receive queues. Software allocates the queues among the different processors, therefore sharing the load of packet processing among several processors. One of the most common use cases of hash filters is the RSS hash defined by Microsoft* and used widely by other operating systems as well. Other generic hash functions are used by embedded systems as well. The hash filter directs the received packets to queue index within a region of queues of the VSI as shown in Figure 38-2. Following are some terms relating the hash filter.

Packet Classifier Type

The PCTYPE is the lowest index PCTYPE that is enabled for the function. The PCTYPE's are enabled for the hash filter by the PFQF_HENA registers for the PFs and by the VFQF_HENA registers for the VFs. See Table 38-128 for the PCTYPE's and its priority order.

Input Set

The input set for the hash filter is listed in the Table 38-128.

Hash Function

The 32- bit hash function is based on Toeplitz algorithm or simple XOR scheme as explained in Section 38.21.3.9. The hash can be calculated on the packet's fields as is or can be symmetric hash as explained in Section 38.21.3.9.2. The hash signature is reported in the receive descriptor if the space is not taken by other filters. The hash is calculated only on the input set words. Masked bits within the input set words are replaced by zero's for the hash calculation.

Queue Index LUT

The queue index look up table (LUT) gets the LS bits of the hash output and provides a queue index within a region. There are per PF lookup tables (PFQF_HLUT) that can be accessed using the 7 or 9 LS bits of the hash output having either 128 or 512 entries, respectively (as defined by the *HASHLUTSIZE* field in the PFQF_CTL_0 register). Each entry in these LUTs defines receive queues in the range of 0 to 127. In

addition a per VSI LUT is defined (VSIQF_HLUT). The LUT in each VSI gets the 6 LS bits of the hash output having 64 entries while each entry defines receive queues in the range of 0 to 15. For each VSI, the PF device driver can define whether it use the PF LUT or the VSI LUT using the RSS_LUT field in the Add VSI command.

Receive Queue Regions

The VSIs support 8 regions of receive queues that are aimed mainly for the TCs. The TC regions are defined per VSI by the VSIQF_TCREGION register. The region sizes (defined by the TC_SIZE fields) can be any of the following values: 1, 2, 4, 8, 16, 32, or 64 as long as the total number of queues do not exceed the VSI allocation. These regions starts at the offset defined by the TC_OFFSET parameter. According to the region size, the 'n' LS bits of the queue index from the LUT are enabled. The hash filter defines the queue index within the queue region of the VSI. The queue region is defined by the TC or the PCTYPE as programmed by the *OVERRIDE_ENA* and *Region* fields in the PFQF_HREGION registers for the PFs and VFQF_HREGION registers for the VFs (see Figure 38-2).

VSI Queue Index

The outcome VSI queue index is shown using the following expressions:

LUT_OUT + VSIQF_TCREGION.TC_OFFSET, while...

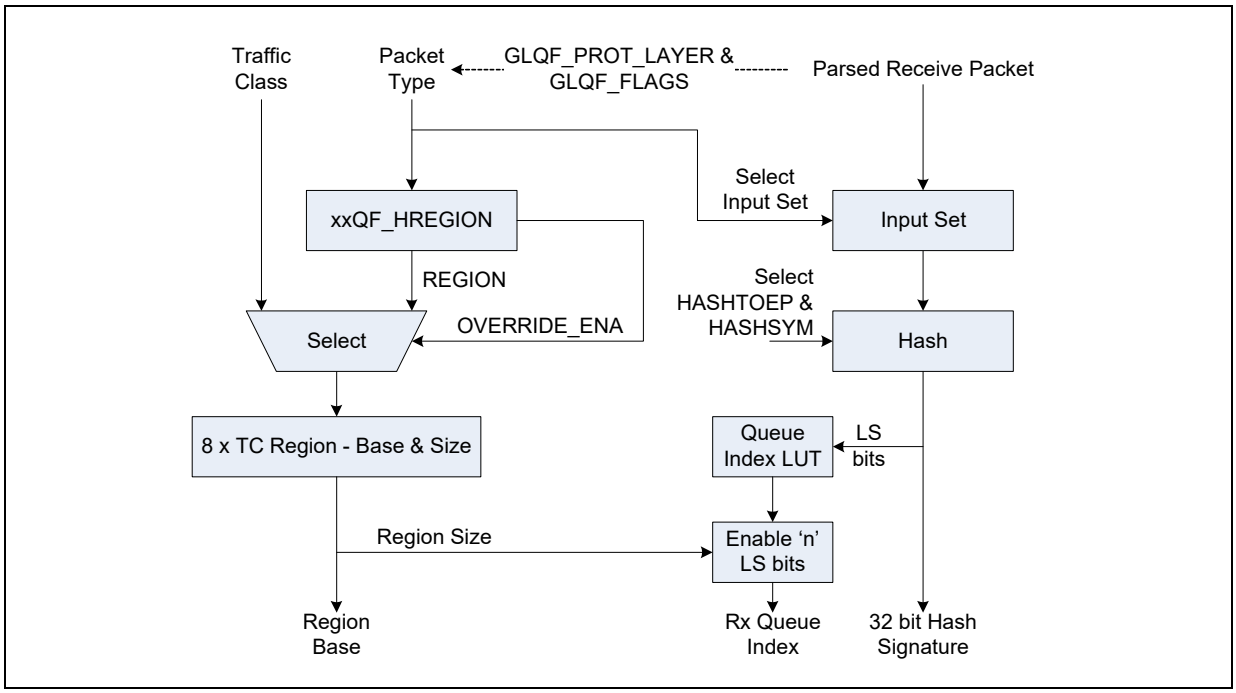
LUT_OUT = (VSIQF_TCREGION->TC_SIZE) LS bits of the QF_LUT

QF_LUT = PFQF_HLUT[9 LS bits of the packet hash] or VSIQF_HLUT[6 LS bits of the packet hash] according to VSIQF_CTL.RSS_LUT_TYPE.

Outcome Queue Index

Mapping the VSI queue index to the PF index space is done by the VSILAN_QTABLE or VSILAN_QBASE.

Figure 38-2. RSS Block Diagram



38.21.3.6.1 Comments for the Init Section

Dynamic setting options (expected at run time):

- VSIQF_TCREGION registers per VSI that define the hash region sizes (should be dynamic b/c of the dynamic LQP allocation)
- PFQF_HLUT registers and VSIQF_HLUT registers per function.

38.21.3.7 Flow Director Filter

The Flow Director (FD) filter is aimed to match specific flow or flows with some action. The FD filter is based on an exact match of the selected tuples named as input set for filtering purposes. The FD filter is composed of the following components.

Filter Enable Option

The FD filtering is enabled by the following parameters: filtering is enabled per PF and its VFs by the PFQF_CTL_0 registers.

Programming

FD filter programming is done by the flow director programming descriptor followed by packet structure that contains the filter fields as shown in Figure 38-3. Failed programming and removal of filter entries are reported by the programming status descriptor. The packet that is used to program the filter can be optionally transmitted depending on the dummy flag in the transmit data descriptor. Note also that the packet used for the programming can be a single packet or part of a TSO. Removing a single filter is done by the same programming descriptor.

Filter programming is possible only from transmit queues that are enabled by the FDENA flag in the transmit queue context. Setting the FDENA flag is

expected only for the PF queues. The PF can program a filter for any of its own VSIs or to VSIs that belong to its VFs. The target VSI for the matched packets is defined by the DEST_VSI parameter in the filter programing descriptor. The PF is permitted to program a FD filter only if the FD filter is enabled by the FD_ENA flag in the PFQF_CTL_0 register. When using the PCTYPE in the FD programming descriptor, the software is permitted to set it only to those values enabled by the GLQF_FDENA registers. When the PCTYPE in the FD programming descriptor is not used (FD_AUTO_PCTYPE flag in the GLQF_CTL is set), the software is unaware of the PCTYPE. Programmed filters with unsupported PCTYPE by the GLQF_FDENA registers won't match any received packets.

The PCTYPE in the FD programming descriptor is translated using the GLQF_FD_PCTYPE table before the filter is programmed.

The *Source* and *Destination* fields in the transmitted packet are presented in a reversed order with respect to the expected received packets. During filter programming time, the hardware swap these fields.

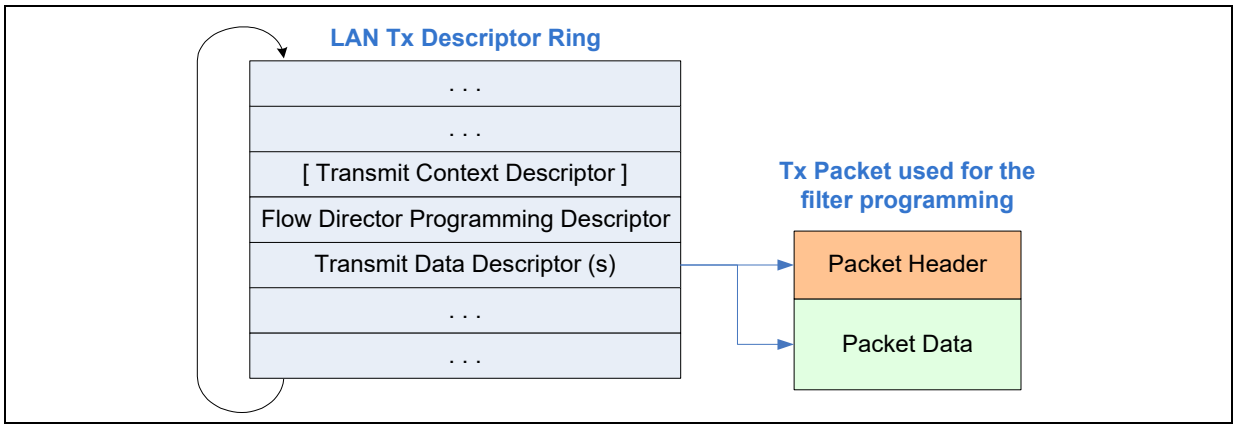
Attempt to re-program an existing filter entry updates the filters parameters.

At programming time the filter can be added on the expense of the guaranteed space or the best effort space of the function as explained in Section 38.21.3.7.1. In those cases, it is required to know up front that a filter programming is accepted by hardware. Software should track its guaranteed space by reading the *GUARANT_CNT* in the *PFQF_FDSTAT* register.

Software-based Removal

In some cases software might need to clear the FD table. Setting the *CLEARFDTABLE* flag in the *PFQF_CTL_1* register, all entries of the PF are invalidated from the FD table. This action can be done during normal transmission and reception. Note that this action might take some time. Software should poll the FD filter counters of the PF in the *PFQF_FDSTAT* register until it is cleared as an indication that the clear all PF entries sequence completed.

Figure 38-3. Flow Director Filter Programming



Hardware-based Removal

Filters might be removed by the device automatically when a transmit or receive packet indicating a TCP flow is closing is detected. These packets are identified by a RST or FIN flag being set. The device removes the filter matching packets for which the PCTYPE bit is set in the GLQF_FDEVICTENA registers and at least one of the flags enabled in GLQF_FDEVICTFLAG.TXFLAG_ENA or GLQF_FDEVICTFLAG.RXFLAG_ENA is set.

In order to define a filter as a candidate for removal by RST or FIN, the *ATR* bit in the *CMD* field (quad word 1, bit 14) in the filter programming descriptor must be set. This allows some differentiation between ATR filters and regular flow director filters managed by software. GLQF_FDEVICTENA relates to the PCTYPEs before GLQF_FD_PCTYPE translation.

Note: The eviction can occur even if the filter is overridden by a PE filter.

Eviction occurs only if the evicting packet is sent or received via the VSI declared in the *DEST_VSI* field when the filter was programmed.

Packet Classifier Type

All PCTYPEs described in Table 38-128 and translated as listed in Table 38-129 are supported for the FD filter.

Input Sets

As opposed to all other filters, the input sets for the FD filter are defined per-port enabling different input set for the flexible payload layers (by the PRTQF_FD_FLXINSET registers). The masking options for the flexible payload protocol layers are defined per-port by the PRTQF_FD_MSK registers. These registers can be loaded from the NVM or programmed by software at init time. Table 38-128 lists the default setting that address the standard protocol layers.

FD Filter Entry Context

Each filter entry consists of 64 bytes. As previously indicated, the input set can be defined as large as 48 bytes while the other 16 bytes are reserved for the filter action, table management parameters and fields that are compared against the received packet:

— PCTYPE and the DEST_VSI that are provided in the FD filter programming descriptor.

- The programming PF index.
- The QINDEX; FLEXOFF; STAT_CNT; FDID; DEST; FD_STATUS.

Filter Match Criteria

A packet matches a filter entry if the following conditions are met:

(1) The target VSI equals to the programmed DEST_VSI.

(2) The identified PCTYPE equals to the programmed PCTYPE.

(3) The received packet pattern matches the programmed one (the relevant fields for the PCTYPE as defined by the FD input set in Table 38-128).

Note that the device checks concurrently for packet match by the FD filter and the PE quad-hash filter. If a packet matches both filters, the action is made by the PE quad-hash filter. Still, the matched FD statistic counters are incremented (GLQF_PCNT).

Filter Action

The filter action is programmed as part of the filter programming command. The actions enabled for the FD filter are listed in Table 38-125. The filter action is taken when the packet matches the filter entry.

FD Table Size

The 10 GbE controller supports 8 KB x 64 KB filters. Out of these bytes, only 48 bytes can be used for the input set.

Hash and Buckets

The FD table is organized in buckets. The number of buckets equals to 16 KB (twice the number of supported filters). Each bucket might be: empty; include a single filter entry; or multiple filter entries. These buckets are managed autonomously by hardware.

38.21.3.7.1 FD Table Allocation

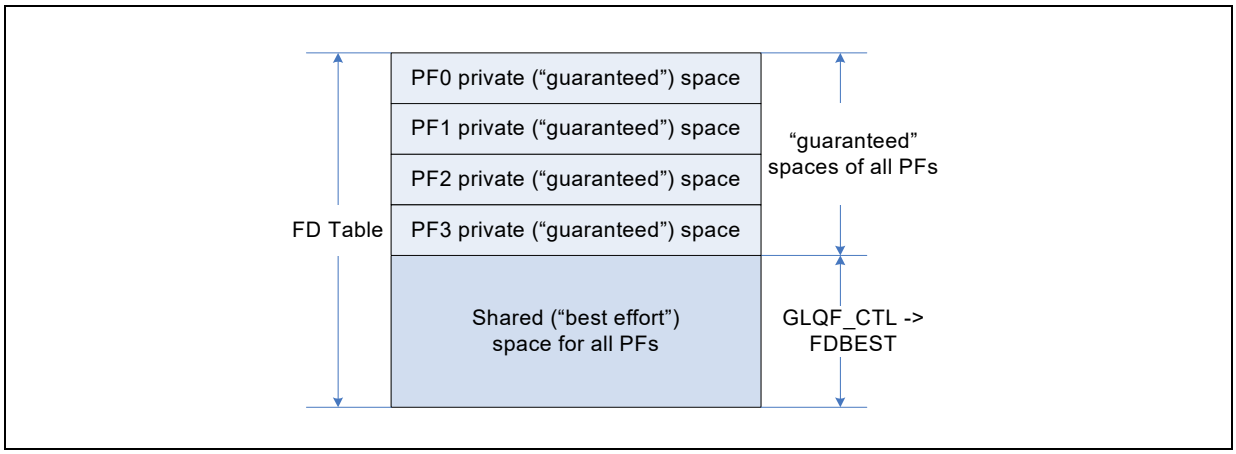
The FD table is a shared resource for all functions. The allocation between the PFs is loaded from the NVM to the PFQF_FDALLOC registers per PF. The FD supports 2 kinds of allocated spaces per PF: guaranteed space and best effort space. Each PF gets a private space, which is a guaranteed number of entries in the FD table (defined by the FDALLOC parameter). The sum of the guaranteed spaces of all PFs must be smaller than or equal to the size of the FD table. If the sum of the FDALLOC parameters is smaller than the size of the FD table, the rest of the space can be used by all function as best effort. The FDBEST parameter in the PFQF_FDALLOC registers define the permitted use of the best effort space by each PF. The total size of the best effort space is defined by a global FDBEST in the GLQF_CTL register (loaded from the NVM). This parameter must not exceed the size of the FD table minus the sum of the FDALLOC parameters of all PFs.

Note: The maximum value of the global FDBEST parameter in the GLQF_CTL register can be set to 8 KB minus 32. In order to make use of all 8 KB filter entries, at least one of the PF's FDALLOC parameters should be larger than zero.

At filter programming, hardware tries both spaces: guaranteed and best effort. Priority between usage of these spaces is set by the PROGPRIO flag in the global GLQF_CTL register. At filter invalidation, the budget is gained back to either the guaranteed space or the best effort space according to a priority setting by the INVALPRIO flag in the global GLQF_CTL register.

Software can track the FD table population (best effort space as well as the guaranteed space). The GLQF_FDCNT_0 reports the total number of filter entries in the entire FD table while the PFQF_FDSTAT reports the number of filter entries used by the PF.

Figure 38-4. Flow Director Table Allocation (Example for 4 x PFs: PF0 ... PF3)



38.21.3.7.2 Statistic Counters

The FD filter might count packets by dedicated GLQF_PCNT counters. A filter can be associated to any of the GLQF_PCNT counters allocated to the PF by setting the CNT_ENA flag and the CNT_INDEX field in the filter programming descriptor. The statistics counters are allocated statically to the PFs according to the PCNT_ALLOC setting in the GLQF_CTL register, as follows:

PCNT_ALLOC Setting	Enabled PFs	GLQF_PCNT counters Per PF	GLQF_PCNT Index Range Per PF	Physical GLQF_PCNT Index In The Device Space
000b	Any number	Flexible	0 ... 511	All 9 bits of the index provided at filter programming. Allocation to PFs should be handled by software. Possible ways to address it is by NVM setting readable to software or any sideband messages between the PFs, which are outside the scope of this document.
100b	PF0 and PF1	256 = 512 / 2	0 ... 255	MS bit of the index is the LS bit of the PF index. 8 LS bits of the index equals the 8 LS bits of the filter index provided at filter programming.
101b	PF0 ... PF3	128 = 512 / 4	0 ... 127	2 MS bits of the index are the 2 LS bit of the PF index. 7 LS bits of the index equals the 7 LS bits of the filter index provided at filter programming.
110b	PF0 ... PF7	64 = 512 / 8	0 ... 63	3 MS bits of the index are the 3 LS bit of the PF index. 6 LS bits of the index equals the 6 LS bits of the filter index provided at filter programming.
111b	PF0 ... PF15	32 = 512 / 16	0 ... 31	4 MS bits of the index are the 4 LS bit of the PF index. 5 LS bits of the index equals the 5 LS bits of the filter index provided at filter programming.

- Should be programmed only if the VSI associated to this filter is enabled to lookup for such packets in the queueing option section of the Add VSI admin command. Hardware does not enforce this.
- VFs should not be enabled before a hash region is allocated to all the filters it is enabled for (VPQF_CTL).

38.21.3.8 Protocol Engine (PE) filters

Filtering to PE is based on a set of rules and filters described in the sections that follow.

38.21.3.8.1 Packet Enablement for the PE

Packets that meet all the following conditions are directed to the PE. Packets that fail criteria 1 through 3 are not candidates for the quad hash filter. Packets that do not meet all the following criteria are directed to the LAN engine's queues.

- Criteria 1: Generic validity
  - The target VSI supports the PE quad hash filter by the PExxx_ENA flags PETCP_ENA flag in the VSIQF_CTL[n] register for the VSI.
  - IP and L4 checksum verification.
    - Case 1 - Received packets from the network: The packet does not have any errors (parsing errors, IP checksum, L4 checksum, etc.).
    - Case 2 - Loopback packets for which the hardware calculated the transmit checksums are candidates for the PE quad hash filter.

Note: Loopback transmit packets for which checksum offload was not requested (checksum done in software) is not a candidate for the quad hash filter, as there is no mechanism to validate the checksum.

  - The packet matches any of the packet types that are candidates for the PE as listed in Table 38-141 (TCP over IPv4 and IPv6).
  - The packet does not have any IPv4 options or IPv6 extension headers and is not fragmented.
  - If The packet is TCP then it is MAC unicast.
  - If the packet is UDP then the packet is not fragmented.
- Criteria 2: If the packet is unicast and matching to the Accelerated Port Bit Vector table (APBVT).
- Criteria 3: Matching to the PE quad hash filter.

Packets that are posted to the LAN engine are directed to queues according to the decision tree.

38.21.3.8.2 APBVT

The L4 destination port number is compared against an internal APBVT (GLQF_APBVT registers). The 10 GbE controller includes a 64 KB x 1-bit vector for the entire device (all 4 LAN ports). An active bit in the table means that the port number is enabled by one (or more) functions. Programming the table is done by the PE interface.

38.21.3.8.3 PE Quad Hash Filter

The quad hash filter is aimed to match specific flows with the PE. The 10 GbE controller maintains some of the quad hash filter in internal cache while the other filters are stored in host memory (FPM). The number of quad hash filters per function is a function of the number of RDMA and UDA flows supported by that function. The quad hash filter is composed of the following components:

Filter Enable Option

The PE quad hash filters are enabled per VSI by the VSIQF_CTL registers.

Packet Classifier Type

The following PCTYPEs are supported by the PE filters: TCP/IPv4; TCP/IPv6; UDP/IPv4; UDP/IPv6. These PCTYPEs are enabled by the PETCP_ENA, PEUUDP_ENA and PEMUDP_ENA flags in the VSIQF_CTL register per VSI. The PCTYPEs for the PE quad hash filter are defined globally by the GLQF_PETYPE registers listed in the Table 38-141.

Cache

Internal cache that can hold up to 1 KB filters. This cache is shared between the PE quad hash filter. In order to minimize access to the host memory in case of cache miss, the PE quad hash filter includes two pre-filters previously described: (1) IP address filters (see Section 38.21.3.8.2); (2) L4 port filter (see Section 38.21.3.8.2).

Input Sets

The input set is defined per PCTYPE by the GLQF_PE_INSET[n] registers. The only validated input set of these registers is listed in Table 38-141.

Table 38-141.PE Quad Hash Input Set

Index 'n'	GLQF_PETYPE[n]	PCTYPE	Input Set (all MSK registers are set to zero)
0	PCTYPE_INDEX = 32 PCTYPE_ENA = 1	TCP/IPv4 with SYN flag and ACK flag clear	Destination MAC address, innermost or only VLAN tag (if present), destination IP address and destination port.
1	PCTYPE_INDEX = 42 PCTYPE_ENA = 1	TCP/IPv6 with SYN flag and ACK flag clear	Destination MAC address, innermost or only VLAN tag (if present), destination IP address and destination port.
2	PCTYPE_INDEX = 33 PCTYPE_ENA = 1	TCP/IPv4 with/no SYN flag or SYN flag with ACK flag set	Destination MAC address, innermost or only VLAN tag (if present), source and destination IP addresses TCP ports.
3	PCTYPE_INDEX = 43 PCTYPE_ENA = 1	TCP/IPv6 with/no SYN flag or SYN flag with ACK flag set	Destination MAC address, innermost or only VLAN tag (if present), source and destination IP addresses TCP ports.

Filter Match Criteria

A packet matches a filter entry if the following conditions are met:

(1) The target function (PF/VF) equals to the function that programmed the filter. If VM, use PF for the filter function type.

(2) The identified PCTYPE equals to the programmed PCTYPE. (3) The received packet pattern matches the relevant input set (as listed in Table 38-141).

Filter Action

The filter provides a hit/miss indication. In the case of hit, the packet is a candidate to the PE. The packet type (PTYPE) and the matched quad hash filter index (QPN) are reported together with the packet. The following cases miss the PE quad hash filters and the packets are posted to the LAN queues according to the other classification filters.

Table 38-142.Packet Types that Miss the PE Quad Hash Filter

Packet Type	Reporting In The LAN Queue
All packet types with any error.	Detected errors.
Unicast packets that miss the APBVT.	LAN packet with no indications from the PE filters.
Non TCP/UDP packets.	
Multicast TCP or fragmented IP.	
Packet with IPv4 options or IPv6 extension headers.	

Quad Hash Context Table

The 10 GbE controller supports up to 256 KB quad hash contexts for the entire device. The PFs can define a smaller (DMA) context table for its usage and its VFs by the *PEDSIZE* parameters in the *PFQF_CTL_2* register and *VPQF_CTL* register, respectively. At filter programming time, software defines a unique context index. The hardware accepts the context programming only if it is defined within the boundaries of the context table size.

Hash and Buckets

The PE quad hash table is organized in buckets. The number of supported buckets is set per function by the *PEHSIZE* parameter in the *PFQF_CTL_2* and *VPQF_CTL* registers for the PFs and VFs, respectively. The bucket space and additional space for buckets that have multiple entries is shown in the Figure 38-5. The bucket is defined by a hash function on the input set for the filter. The hash function is based on Toeplitz algorithm using a global key defined by the *GLQF_HKEY* registers. The concept is similar to the FD while the filter objects are in host memory as shown in Figure 38-6. Note that software must clear (write all zero's) to the bucket space in host memory before enabling the table usage.

FPM Size Limitation

The total number of PE QP contexts for the PF and its VFs are defined by *PEDSIZE* in the *PFQF_CTL_2* registers and *VPQF_CTL* registers, respectively. This number should not exceed the FPM allocation defined by the *GLHMC_PEQPCNT* registers for the PFs and *GLHMC_VFPEQPCNT* registers for the VFs. In addition, the total number of PE quad hash filters for the PF and its VFs is defined by *PEHSIZE* + *PEDSIZE* in the *PFQF_CTL_2* registers and *VPQF_CTL* registers, respectively. This number should not exceed the FPM allocation defined by the *GLHMC_PEHTCNT* registers for the PFs and *GLHMC_VFPEHTCNT* registers for the VFs. Intel recommends to keep a minimal ratio of *PEHSIZE = 4 * PEDSIZE* to achieve reasonable performance.

Filter Search

The packet type is identified, its tuples are extracted and the hash is calculated. If the packet type matches the PE criteria and the packet passes the APBVT criteria it is a candidate for the quad hash filter. A matched filter is searched in the cache. If the filter is not in the cache, hardware fetches the filter from the host memory space of the relevant function at the location indicated by the calculated hash. In case of an exact match, then the process completes and the filter is kept in the cache. In case of a miss, the next filter in the same bucket is fetched to hardware repeating the previous step until the end of the bucket or a matched filter is found. If a matched filter is not found, it is considered a miss and the relevant tuples of the packet are stored in the cache. Even if the packet misses the PE context filter, its tuples are stored in the cache.

Figure 38-5. Buckets and Collided Filter Entries Space

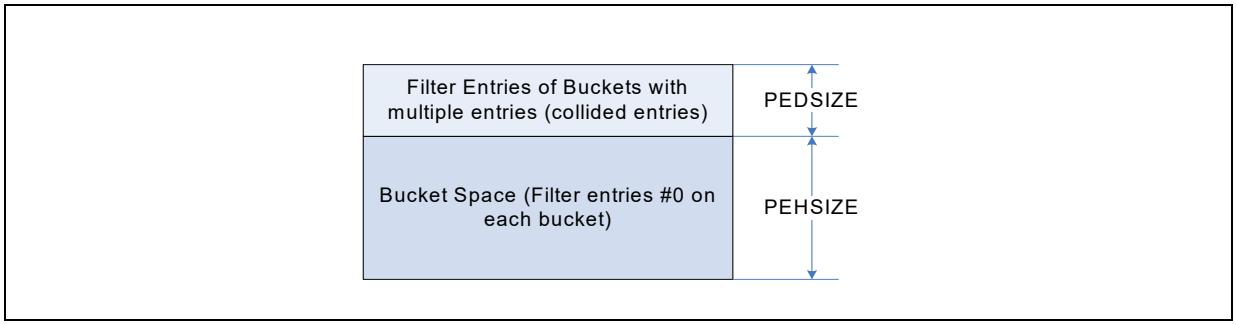
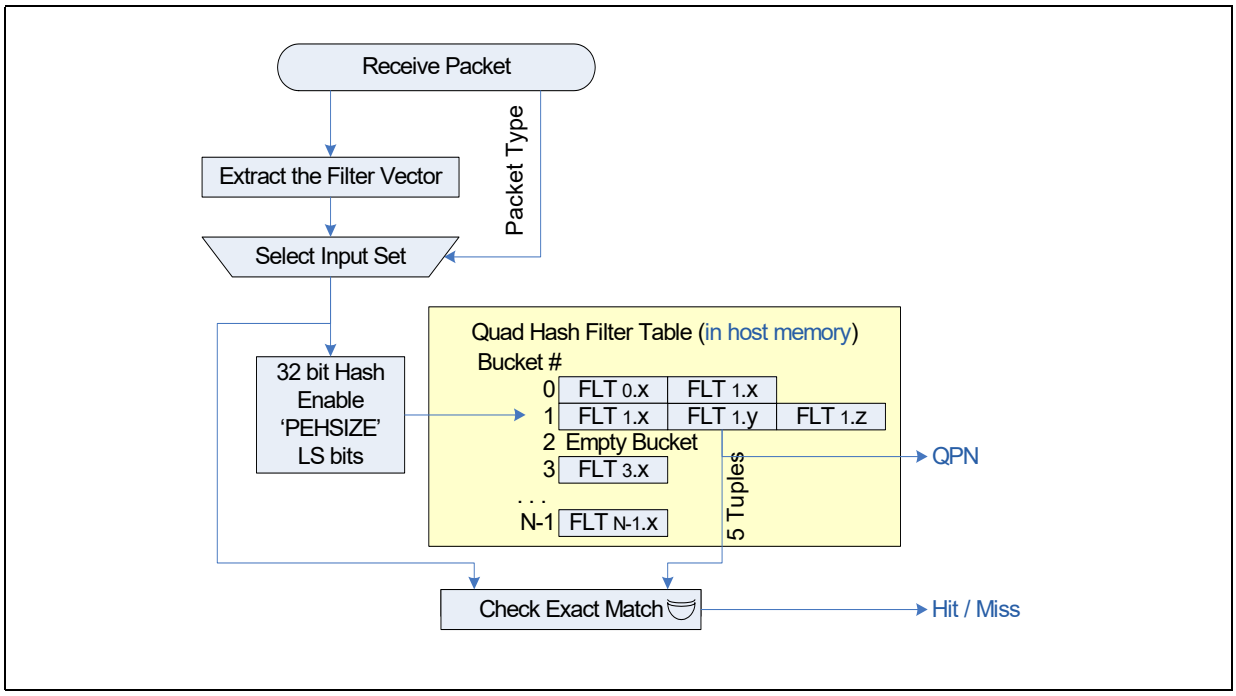


Figure 38-6. Quad Hash Filter



Filter Programming

- A filter can be added or removed from the quad hash filter table only by the PE using an internal PE-to-filter interface. As part of the filter programming parameters, the PE indicates quad hash filter ID (QPN). It is the PE responsibility to provide unique QPN for each filter per function.
- The *Source* and *Destination* fields in the field vector should be programmed in the ordering of the received packet.
- The PF software is expected to initialize (set to zero) the entire space in the FPM assigned for the PE quad hash filter table. This step should be part of the software device driver initialization flow. The initialized FPM provides an indication to hardware that the table is empty.
- Note that a filter should be programmed only if the VSI associated to this filter is enabled to lookup for such packets in the *Queueing Option* section of the *Add VSI* admin command. Hardware does not enforce this.

Performance

Hardware is able to sustain line rate when the PE filters are in the cache. If the filter is not in the cache some latencies is added to the processing in order to fetch the context from the host memory. If the latency of fetching a filter is larger than 1.8 µs, performance is affected as well. The 10 GbE controller reports the total number of buckets of a function in the xxQF_PECNT_0 registers. This counter provides an indication for the PE filter table distribution rather than exact indication of the hardware performance. The hardware performance depends also on the average packet length and the rate of hitting the cache vs. long vs. short buckets. Still, when LONGBCNT exceeds some threshold, a PF software might chose to initialize the filter and restart while increasing the number of buckets for the filter. The PE table counts each packet positioning in its bucket at reception time. Processed PF packets by the PE filter at a position shorter or equal than MAXPEBLEN is counted by PFQF_PECNT_2.HITSBCNT.

Packets at position longer than MAXPEBLEN are counted by PFQF_PECNT_3.HITLBCNT and also reported by a PELONGB flag in the packet status to the PE and the receive descriptor.

Comments for the Init Section

The following parameters can be modified only when the quad hash filter is disabled for the function by all *PExxx_ENA* flags in the matched *VSIQF_CTL* register(s).

- The number of quad hash table buckets — *PEHSIZE* field in the *PFQF_CTL_2* or *VPQF_CTL* registers for the PF or VF, respectively.

VFs should not be enabled before a hash region is allocated to all the filters it is enabled for (*VPQF_CTL*).

Dynamic setting options (expected at run time).

- VF reset and VFLR as well as PF reset and FLR:
  - PE firmware removes the relevant APBVT entries.

38.21.3.9 Hash Functions

The 10 GbE controller supports several hash functions to be used by the various filters:

Microsoft* Toeplitz-based hash vs. simple hash. These hash methods are explained in the following sub-sections. Selection between the two schemes is controlled by the global GLQF_CTL register, as follows:

- HTOEP controls the scheme used by the hash filters for all PCTYPE's
- Symmetric hash (explained in Section 38.21.3.9.2)

The following notation is used to describe the hash functions:

- Ordering is little endian in both bytes and bits. For example, the IP address 161.142.100.80 translates into 0xa18e6450.
- A ^ denotes bit-wise XOR operation of same-width vectors.
- @x-y denotes bytes x through y (including both of them) of the incoming packet, where byte 0 is the first byte of the IP header. In other words, we consider all byte-offsets as offsets into a packet where the framing layer header has been stripped out. Therefore, the source IPv4 address is referred to as @12-15, while the destination v4 address is referred to as @16-19.
- @x-y, @v-w denotes concatenation of bytes x-y, followed by bytes v-w, preserving the order in which they occurred in the packet.

38.21.3.9.1 Microsoft Toeplitz-based Hash

This hash uses a random secret key of 416 bits (52 bytes). The key is defined per VSI by the VSIQF_HKEY registers. The algorithm works by examining each bit of the hash input from left to right. Our nomenclature defines left and right for a byte-array as follows: Given an array K with k bytes, our nomenclature assumes that the array is laid out as follows: K[0] K[1] K[2] … K[k-1]

While K[0] is the left-most byte, and the MSB of K[0] is the left-most bit. K[k-1] is the right-most byte, and the LSB of K[k-1] is the right-most bit.

```
ComputeHash(input[], N)

    For input[] of length N bytes (8N bits) and a random secret key K of 416 bits
```

```
    Result = 0;

    For each bit b in input[] {

        if (b == 1) then Result ^= (left-most 32 bits of K);

        shift K left 1 bit position; // cyclic shift while the right-most bit of K
gets the left-most bit of K

    }

    return Result;
```

Pseudo-code examples

The following four pseudo-code examples are intended to help clarify exactly how the hash is performed in four cases, IPv4 with and without ability to parse the L4 header, and IPv6 with and without a L4 header.

Table 38-143.Examples of Input Fields for the Hash Function

Packet Type	Hash Input	Hash Result
IPv4 with TCP or UDP	SourceAddress, DestinationAddress, SourcePort, DestinationPort: Input[12] = @12-15, @16-19, @20-21, @22-23	Result = ComputeHash(Input, 12)
IPv4 without L4	SourceAddress, DestinationAddress: Input[12] = @12-15, @16-19	Result = ComputeHash(Input, 8)
IPv6 with TCP or UDP	SourceAddress, DestinationAddress, SourcePort, DestinationPort: Input[12] = @8-23, @24-39, @40-41, @42-43	Result = ComputeHash(Input, 36)
IPv6 without L4	SourceAddress, DestinationAddress: Input[12] = @8-23, @24-39	Result = ComputeHash(Input, 32)

RSS verification suite

This section provides a verification suite used to validate that the hash function is computed according to Microsoft nomenclature.

Assume that the random key byte-stream is:

```
0x6d, 0x5a, 0x56, 0xda, 0x25, 0x5b, 0x0e, 0xc2,
0x41, 0x67, 0x25, 0x3d, 0x43, 0xa3, 0x8f, 0xb0,
0xd0, 0xca, 0x2b, 0xcb, 0xae, 0x7b, 0x30, 0xb4,
0x77, 0xcb, 0x2d, 0xa3, 0x80, 0x30, 0xf2, 0x0c,
0x6a, 0x42, 0xb7, 0x3b, 0xbe, 0xac, 0x01, 0xfa
```

Table 38-144.IPv4

Destination Address / Port	Source Address / Port	IPv4 only	IPv4 with TCP
161.142.100.80 / 1766	66.9.149.187 / 2794	0x323e8fc2	0x51ccc178
65.69.140.83 / 4739	199.92.111.2 / 14230	0xd718262a	0xc626b0ea
12.22.207.184 / 38024	24.19.198.95 / 12898	0xd2d0a5de	0x5c2b394a
209.142.163.6 / 2217	38.27.205.30 / 48228	0x82989176	0xafc7327f
202.188.127.2 / 1303	153.39.163.191 / 44251	0x5d1809c5	0x10e828a2

The IPv6 address tuples are only for verification purposes, and may not make sense as a tuple.

Table 38-145.IPv6

Destination Address / Port	Source Address / Port	IPv6 only	IPv6 with TCP
3ffe:2501:200:3::1 / 1766	3ffe:2501:200:1fff::7 / 2794	0x2cc18cd5	0x40207d3d
ff02::1 / 4739	3ffe:501:8::260:97ff:fe40:efab / 14230	0x0f0c461c	0xdde51bbf
fe80::200:f8ff:fe21:67cf / 38024	3ffe:1900:4545:3:200:f8ff:fe21:67cf / 44251	0x4b61e985	0x02d1feef

Simple Hash

Simple hash is an alternative hash function that trades runtime for effectiveness. It is provided for applications that recompute the hash in software and wish a lighter version of the hash function. Simple hash provides a more skewed distribution vs. the Microsoft hash. It is enabled only for the hash filter by the HTOEP flag in the global GLQF_CTL register.

Simple hash is performed by partitioning the sequence of input bits into 32-bit entities and XOR those to generate a 32-bit output.

```
ComputeSimpleHash(input[], N)

    For input[] of length N bytes (8N bits)

    Result = 0;

    For each DWord (32 bits) in input[] {

        Result ^= (left-most 32 bits of input);

        shift input left 32 bit position;

    }

return Result;
```

The input[] vector is padded by 0 to 3 bytes of zero's making it whole number of Dwords. In those cases that the hash function is used as an address to a lookup table smaller or equal then 64 KB entries, the 16 MS bits of the results are XORed with the 16 LS bits creating a 16 bit output. If the hash function is used as an address to a lookup table smaller or equal than 256 entries, the 4 bytes of the previous results are XORed together creating an 8 bit output.

38.21.3.9.2 Symmetric Hash

A symmetric hash provides the same value if its respective source and destination fields are swapped. For example, suppose that the hash is done on the frame source and destination IP addresses. A symmetric hash guarantees that: **hash(src IP, dst IP) = hash(dst IP, src IP)**.

The motivation behind symmetric hash is to route frame between two addresses to the same queue independent of the direction of transmission.

The symmetric hash is useful for field pairs like the IP addresses, L4 port numbers, MAC addresses, FC_IDs and FC exchange IDs. The field pairs are defined globally per packet type by the GLQF_SWAP registers. The symmetric hash is obtained by replacing the original source and destination fields by a XOR value of these fields before calculating the hash.

Symmetric hash is enabled per PCTYPE by the GLQF_HSYM and per port by the HSYM_ENA flag in the PRTQF_CTL_0 register.

Symmetric Hash Example for IPv4 with TCP

Non-symmetric input vector is: Input[12] = @12-15, @16-19, @20-21, @22-23.

Symmetric input vector is: Input[12] = @12-15 ^ @16-19, @12-15 ^ @16-19, @20-21 ^ @22-23, @20-21 ^ @22-23.

38.21.3.10 Rx filter Admin Queue Commands

38.21.3.10.1 Add Tunneling UDP (0x0B00)

Add the Tunneling UDP admin command (listed in Table 38-146) is used to define a tunneling UDP filter. It defines the tunneling UDP port number and its header size, associating the filter to the PF that initiated the Add command. The device response to this command is as follows:

- If the requested UDP port is already defined for the PF then report a completion with an EEXIST error code.
- If the requested UDP port is already defined with a different tunneling UDP protocol type, then report a completion with EMODE error code.
- If the requested UDP port is already defined with matched parameters for other PF(s) then update the filter entry:
  - Set the matched PF flag of the filter
  - Report back a successful programming completion
- If the requested UDP port does not exist then add a new filter entry:
  - Search for a free entry. If there is no free entry in the table then report completion with "ENOSPC" error code.
  - Add the entry to the table:
    - Set the matched PF flag of the filter
    - Report back a successful programming completion

Table 38-146.Add Tunneling UDP Command (Opcode: 0x0B00) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1		
Opcode	2-3	0x0B00	Command opcodes.
Datalen	4-5	0x00	N/A (reserved zero).
Return Value/ VFID	6-7	0x00	N/A (reserved zero).
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
UDP Port	16-17		A 16-bit value of the UDP port number to be added (while byte 17 is the MS byte, which is first on the wire). Note that a zero value is a reserved value and should not be used by this command.
Reserved	18		Reserved.
Reserved	19		Reserved.
Reserved	20		Reserved.

Table 38-146.Add Tunneling UDP Command (Opcode: 0x0B00) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Tunneling UDP Protocol Type	21		Tunneling UDP type should be programmed as follows: 0x00 = For VXLAN key. 0x01 = For Geneve key (next protocol is the optional Geneve header length). 0x02 = Reserved option for MAC in UDP tunneling. 0x10 = For Teredo. 0x11 = For VXLAN-GPE (next protocol is defined in the header) Else = Reserved.
Reserved	22-31		Reserved.

Table 38-147.Completion for the Add Tunneling UDP Command (Opcode: 0x0B00)

Name	Bytes.Bits	Value	Remarks
Flags	0-1		
Opcode	2-3	0x0800	Command opcode
Datalen	4-5	0x00	N/A
Return value/VFID	6-7		Some comments on specific errors: No Error = No error. EEXIST = Add filter rejected because it already exist. ENOSPC = Add filter rejected because of no space. EMODE = Add filter rejected because the filter parameters in the add Tunneling UDP command do not match the programmed parameters in an existing filter with the same UDP port.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
UDP Port	16-17		A 16-bit value of the UDP port number that was added.
Filter Entry Index	18		Index of the added tunneling UDP filter (0 to 15). This index is assigned by the device to be used by the PF to remove the filter.
Multiple PFs	19		This field equals to 0x1 when other PF(s) own the same filter entry and equals to 0x0 otherwise.
Total Filters	20		Total number of Tunneling UDP filters used by all PFs after the completion of the Add Filter command.
Reserved	21-31		Reserved.

38.21.3.10.2 Remove Tunneling UDP (0x0B01)

The remove Tunneling UDP admin command (listed in Table 38-149) is used to delete a tunneling UDP filter from the table. The device response to this command as follows:

- If the requested filter (defined by the filter entry index) is not owned by the PF then report an ENOENT error code.
- Else, (the requested filter to be removed is owned by the PF), clear an internal flag that associates the filter entry with the PF and report back successful filter removal completion.
  - If the filter is not associated with any PFs then set the *Entry Free* flag in the reported completion status.

Table 38-148.Remove Tunneling UDP Port Command (Opcode: 0x0B01)

Name	Bytes.Bits	Value	Remarks
Flags	0-1		
Opcode	2-3	0x0B01	Command opcodes.
Datalen	4-5	0x00	N/A (reserved zero).
Return Value/ VFID	6-7	0x00	N/A (reserved zero).
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-17		Reserved, zero.
Tunneling UDP Filter Index	18.0:3		Filter entry index to be removed (0 to 15).
Reserved	18.4-31		Reserved, zero.

Table 38-149.Completion for Remove Tunneling UDP Command (Opcode: 0x0B01)

Name	Bytes.Bits	Value	Remarks
Flags	0-1		
Opcode	2-3	0x801	Command opcode.
Datalen	4-5	0x00	N/A
Return Value/ VFID	6-7		Some comments on specific errors: No Error = No error. ENOENT = Remove filter rejected because no matched entry was found.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
UDP Port	16-17		A 16-bit value of the UDP port number.
Filter Entry Index	18		The index of the requested filter entry to be removed (between 0 to 15).
Multiple PFs	19		This field equals to 0x1 when other PF(s) still own the same filter entry and equals to 0x0 otherwise.
Total Filters	20		Total number of tunneling UDP filters used by all PFs after the completion of the remove filter command.
Reserved	21-31		Reserved, zero.

Set RSS Key (0x0B02)

This command is used to set the RSS key of a VSI. It can be used by a PF or a VF for VSIs assigned to it.

Table 38-150.Set RSS Key Command (Opcode: 0x0B02) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0B02	Command opcode.
Datalen	4-5	0x34	Length of buffer. Should be equal to 0x34 (hash key size).
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.

Table 38-150.Set RSS Key Command (Opcode: 0x0B02) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
VSI#	16-17	0x0	Bit 15 = Valid VSI#. Must always be set. Ignored by firmware. Bit 14:9 = Reserved. Bit 8:0 = VSI number
Reserved	20-23	0x0	Reserved.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

The command buffer of this command contains the RSS key to set.

Offset	Content
0x0 - x027	Standard RSS key.
0x28 - 0x33	Extended hash key. Must be zero for regular RSS.

Table 38-151.Set RSS Key Response (Opcode: 0x0B02)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0B02	Command opcode.
Datalen	4-5	0x0	No return buffer.
Return Value/ VFID	6-7		Return value. The following error values can be returned: ENOENT = If not a valid VSI. EACCES = If the VSI is not owned by this PF/VF.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-23		Reserved.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

38.21.3.10.3Set RSS LUT (0x0B03)

This command is used to set the RSS LUT of a VSI. It can be used by a PF or a VF for VSIs assigned to it.

Table 38-152.Set RSS LUT Command (Opcode: 0x0B03) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0B03	Command opcode.
Datalen	4-5	64/128/512	Length of buffer (should be equal to 0x).
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
VSI#	16-17		Bit 15 = Valid VSI# (must be set if table type = VSI, ignored by firmware otherwise. Bit 14:9 = Reserved. Bit 8:0 = VSI number.

Table 38-152.Set RSS LUT Command (Opcode: 0x0B03) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Command Flags	18-19		Table type: 0 = VSI 1 = PF. Only a PF can set this bit. 15:1 = Reserved.
Reserved	20-23	0x0	Reserved.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

The command buffer of this command contains the LUT to set.

Offset	Content
0x0 - LUT size	LUT content must match the format of the VSIQF_HLUT or PFQF_HLUT registers. VSI LUT = Must be 64 bytes. PF LUT = must be 512 bytes or 128 bytes. If the buffer size is set to 128 bytes only the first 128 bytes of the LUT are set.

Table 38-153.Set RSS LUT Response (Opcode: 0x0B03)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0B03	Command opcode.
Datalen	4-5	0x0	No return buffer.
Return Value/ VFID	6-7		Return value. The following error values can be returned: ENOENT = If not a valid VSI. EACCES = If the VSI is not owned by this PF/VF.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-23		Reserved.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

38.21.3.10.4 Get RSS Key (0x0B04)

This command is used to get the RSS key of a VSI. It can be used by a PF or a VF for VSIs assigned to it.

Table 38-154.Get RSS Key Command (Opcode: 0x0B04) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0B04	Command opcode.
Datalen	4-5	0x34	Length of buffer (should be equal to 0x34 (hash key size) or more).
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.

Table 38-154.Get RSS Key Command (Opcode: 0x0B04) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
VSI#	16-17	0x0	Bit 15 = Valid VSI#. Must always be set. Ignored by firmware. Bit 14:9 = Reserved. Bit 8:0 = VSI number.
Reserved	20-23	0x0	Reserved.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

Table 38-155.Get RSS Key Response (Opcode: 0x0B04)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0B04	Command opcode.
Datalen	4-5	0x0	0x34.
Return Value/ VFID	6-7		Return value. The following error values can be returned: ENOENT = If not a valid VSI. EACCES = If the VSI is not owned by this PF/VF.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-23		Reserved.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

The response buffer of this command contains the RSS key currently set.

Offset	Content
0x0 - x027	Standard RSS key.
0x28 - 0x33	Extended hash key (must be zero for regular RSS).

38.21.3.10.5Get RSS LUT (0x0B05)

This command is used to get the RSS LUT of a VSI. It can be used by a PF or a VF for VSIs assigned to it. A PF can query the RSS LUT of its VFs.

Table 38-156.Get RSS LUT Command (Opcode: 0x0B05) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0B02	Command opcode.
Datalen	4-5	64/512	Length of buffer. Should be equal to 512 or more for a PF table or 64 or more for a VSI table.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
VSI#	16-17		Bit 15: Valid VSI# (Must be set if table type = VSI. Ignored by firmware otherwise). Bit 14:10 = Reserved. Bit 9:0 = SEID number of the VSI.

Table 38-156.Get RSS LUT Command (Opcode: 0x0B05) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Command Flags	18-19		Table type: 0 = VSI 1 = PF. Only a PF can set this bit. 15:1 = Reserved.
Reserved	20-23	0x0	Reserved.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

Table 38-157.Get RSS LUT Response (Opcode: 0x0B05)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0B05	Command opcode.
Datalen	4-5	0x0	No return buffer.
Return Value/ VFID	6-7		Return value. The following error values can be returned: ENOENT - If not a valid VSI. EACCES - If the VSI is not owned by this PF.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-23		Reserved.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

The response buffer contains the LUT currently set.

Offset	Content
0x0 = LUT Size	LUT content, as read from the VSIQF_HLUT or PFQF_HLUT registers. VSI LUT = 64 bytes. PF LUT = 512 bytes.

38.21.4 L2 Packet Processing

38.21.4.1 CRC Handling

38.21.4.1.1 Ethernet CRC Insertion

The 10 GbE controller calculates and inserts the Ethernet CRC for all packets transmitted to the network according to a per port setting configured by setting the *CRC Enable* bit of the Set MAC Config Admin Queue command.

38.21.4.1.2 Ethernet CRC Stripping

The 10 GbE controller checks the integrity of the Ethernet CRC and possibly strips it. On packets that are posted to LAN queues, The Ethernet CRC bytes are stripped according to setting of the *CRCStrip* flag in the target LAN queue context.

Packets that are routed to PE queue pairs are striped from the Ethernet CRC (with no setting option).

38.21.4.2 L2 Padding

Transmit packets to the network are padded with zeros to 60 bytes) guaranteeing that their length including the CRC is at least 64 bytes.

Receive packets to the host are padded as well. Packets are padded with zero guaranteeing that they are never shorter than 60 bytes if the following conditions are met:

- Received packet to host memory from the network with no CRC bytes (stripped by the device) and additional stripped fields (like VLAN) that their remaining length is shorter than 60 bytes.
- Loopback packets to host memory (VM to VM) with stripped fields (like VLAN) that their remaining length is shorter than 60 bytes.
- Same rules for packets destined to the EMP.

38.21.4.3 L2 Tag Handling

38.21.4.3.1 Overview

This section describes the generic mechanism used to handle L2 tags in the 10 GbE controller.

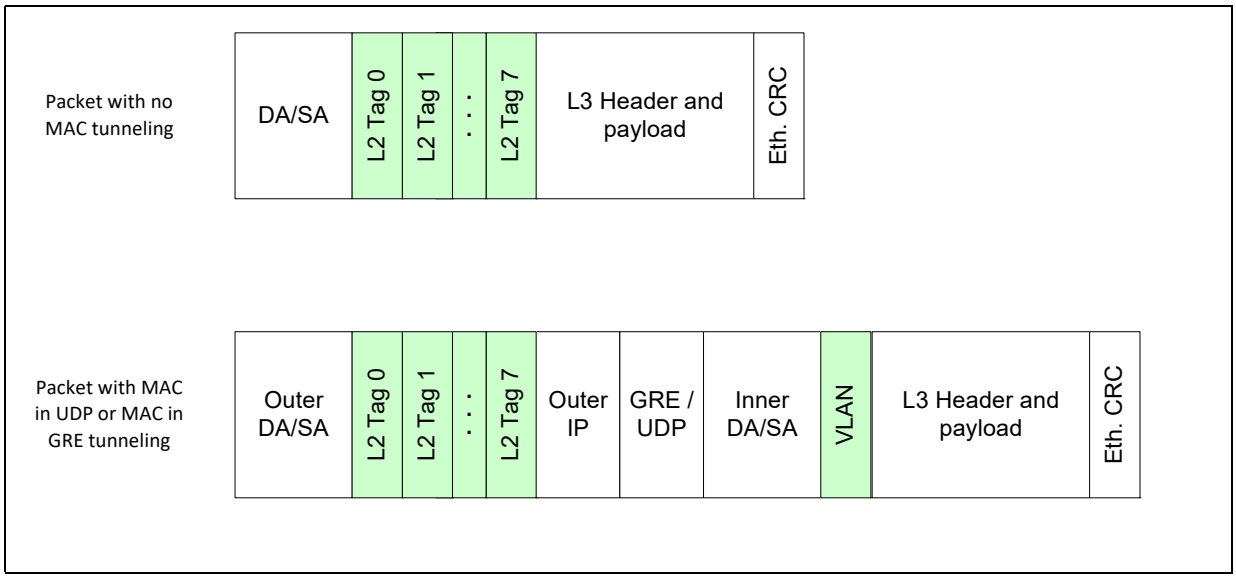
The 10 GbE controller supports up to 8 tags. In general, the text that follows applies to each tag independently.

The order these tags are expected in a packet is according to their index in the array listed in Table 38-161. Index 7 is the most inner L2 tag (closest to the L3 header) and index 0 is the most outer L2 tag (closest to the MAC address) as described in Figure 38-7.

Note: The following text refers to offloads provided by the device. A packet received from the network can have any number of tags, and a packet transmitted might have any number of tags added by software provided no offload is required.

See Section 38.21.4.3.4 for the programming interface to describe the supported tags and the way to handle them.

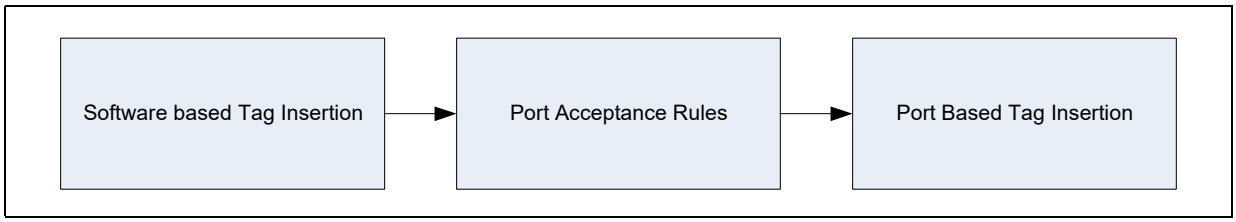
Figure 38-7. L2 Tags Order



38.21.4.3.2 Transmit Tag Handling

The transmit tag handling has three logical parts as described in Figure 38-8:

Figure 38-8. Transmit Tag Handling



Software-based tag insertion is described in Software-based Tag Insertion Rules. The port acceptance rules and port based tag insertion are described in Port (VSI) Based Tag Handling (Accept Rules).

Section 38.21.4.3.3 describes the receive tag handling and Section 38.21.4.3.4 describes the software interface used to manipulate the tags.

Software-based Tag Insertion Rules

Software can require to send packets with a different L2 tag, it might do so either directly through the data for all the tags or using the transmit descriptor for up to two tags.

The tags available for descriptor-based insertion are fixed by the device according to the mode of operation.

The type of the tag taken from the *L2TAG1* field in this descriptor when the *IL2TAG1* field is set is defined by the *VSI_L2TAGSTXVALID.L2TAG1INSERTID* field when *VSI_L2TAGSTXVALID.L2TAG1INSERTID_VALID* is set. If the type of the L2 tag requires more than 2 variable bytes, then additional bytes are taken from the *L2TAG2* field while the *L2TAG1* field is first on the wire. In this case, the *IL2TAG2* flag must be cleared.

The type of the tag taken from the *L2TAG2* field in this descriptor when the *IL2TAG2* field is set is defined by the *VSI_L2TAGSTXVALID.L2TAG2INSERTID* field when *VSI_L2TAGSTXVALID.L2TAG2INSERTID_VALID* is set.

After the tags are inserted, according to the software request, the rules per VSI described in Port (VSI) Based Tag Handling (Accept Rules) are applied to decide if the packet can be sent.

Single Tag Handling

Table 38-158 lists the tag insertion in different cases when each tag is identified by a different EtherType.

Table 38-158.Single Tag Handling

Tag in The Data Buffer	Tag In The Tx Descriptor	Software Requested Action
No	No	Do nothing (send untagged packet)
No	Yes	Insert Tag from the descriptor
Yes	No	Do Nothing (send packet with the tag from the buffer)
Yes	Yes	This is not a valid configuration and should not be used.

Note: Table 38-158 relates to the configuration of a single tag. The configuration of different tags is independent.

Double Tag Handling

If two tags uses the same EtherType (such as VLAN and double VLAN), there might be some cases where hardware might not be able to identify which of the two tags were inserted by the software device driver.

In this case, if only one tag is seen in the packet and no insertion was requested by the software device driver, it is assumed to be the outer of the two tags (in this case, the outer VLAN). Any further action (anti-spoof, UP translation, tag accept policy and port based tagging) interprets this tag according to this. For example, if anti-spoof is applied only to inner VLAN, a packet with a single VLAN is treated as a packet with no VLAN for anti-spoof. If a single tag is present in the buffer sent by the host, but the descriptor request insertion of one of the tags (like an outer VLAN), then it is assumed that the tag present in the buffer is the tag for which offload was not requested (in this case, the inner VLAN).

After the decision on the identity of each tag as previously described, the handling of each tag is as described in Single Tag Handling.

Port (VSI) Based Tag Handling (Accept Rules)

The previous section described how tags are inserted via the data buffer or the transmit descriptor. This section describes the rules applied per VSI to decide if the software request is acceptable. Table 38-159 lists the applied rule according to tag accept, tag insert and the tag presence in the packet for a specific tag.

Table 38-159.Port-Based Tag Handling (Transmit)

Tag Accept Mode	Tag Inserted By Driver?	Port-based Tag	Allowed Driver Behavior[1]	Action
Allow untagged only: VSI_TAR.ACCEPTTAGGED = 0b ACCEPTUNTAGGED = 1b	No	No	Yes	Send the packet as is.
	No	Yes	Yes	Tag inserted by the VSI.
	Yes - from descriptor	Yes	No	VSI's VLAN tag overrides software while keeping software priority.
	Yes - in packet	Yes	No	Drop packet.
	Yes	No	No	Drop packet.
	VLAN ID = 0[2]	No	Yes	Send the packet as is.
	VLAN ID = 0	Yes	Yes	VSI's VLAN tag overrides software while keeping software priority. This mode is supported only if a tag is inserted in the descriptor.
Allow tagged and untagged: VSI_TAR.ACCEPTTAGGED = 1b ACCEPTUNTAGGED = 1b)	X	Yes	Unexpected VSI setting	Undefined.
	X	No	Yes	Send the packet as is.
Allow tagged only: VSI_TAR.ACCEPTTAGGED = 1b ACCEPTUNTAGGED = 0b	X	Yes	Unexpected VSI setting	Undefined.
	No	No	No	Drop packet.
	Yes	No	Yes	Send the packet as is.
Accept nothing VSI_TAR.ACCEPTTAGGED = 0b ACCEPTUNTAGGED = 0b	X	X	X	Non-legal configuration.

Notes:

1. Unexpected VSI setting means that the combination of the tag accept mode and port-based tag should not be requested when creating a VSI.
2. If *GL_SWT_L2TAGCTRL.ISVLAN* is set (can be set only in one tag, the inner VLAN).

Port-based Tag Insertion Mechanism

After the packet that was sent by the software device driver was accepted, the 10 GbE controller might add up to three tags based on the VSI that sent the packets. Two of the added tags can have a variable part of up to 16 bits and one tag can have a variable part of up to 32 bits. The tags that are allowed to be port-based are determined via the *VSI_L2TAGSTXVALID.TIR[012]_INSERT* and *VSI_L2TAGSTXVALID.TIR[012]INSERTID* bits.

The tags for which port-based insertion is done are fixed by the device according to the mode of operation.

For tags up to 8 bytes (not including the EtherType), the entire tag can be inserted by hardware. Either as a request of the software via the descriptor as described in Software-based Tag Insertion Rules or as part of a port-based tag insertion.

Each tag might contain the following parts:

- The EtherType
- A fixed part
- A variable part (up to 16 or 32 bits)

The EtherType is taken from the *GL_SWT_L2TAGCTRL.ETHERTYPE* field.

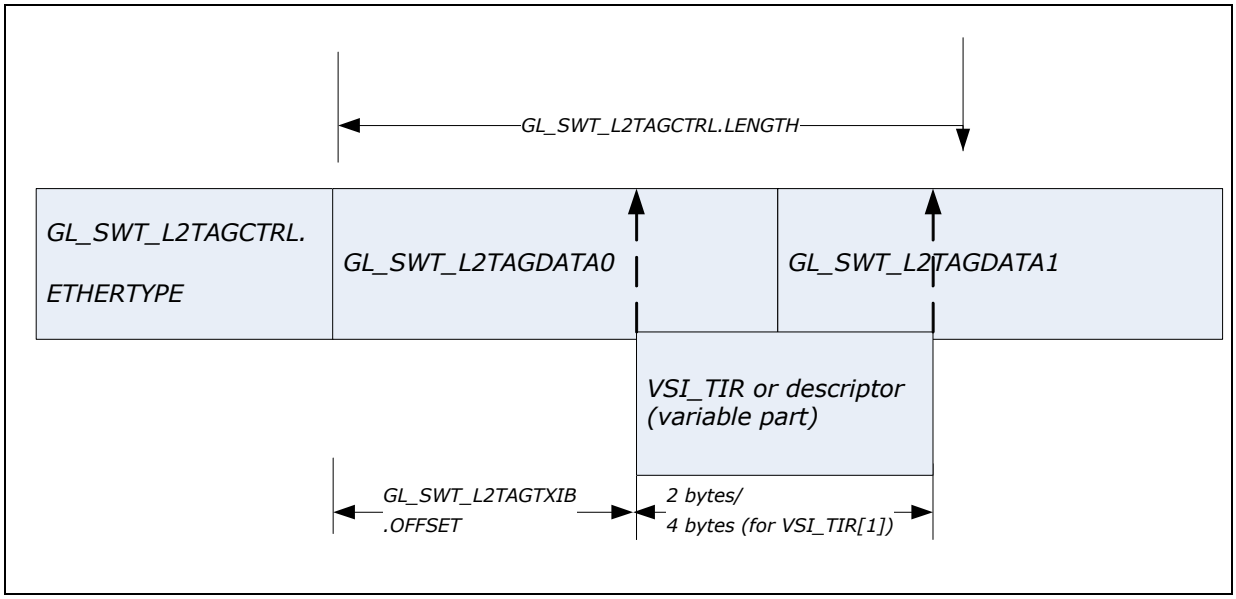
The fixed part is taken from the GL_SWT_L2TAGDATA0 and GL_SWT_L2TAGDATA1 register. The *GL_SWT_L2TAGCTRL.LENGTH* field indicates the length of the EtherType payload.

Note: The unused part of the fixed part and the word in which the variable part is inserted should be set to zero.

The variable part is extracted from the descriptor or from the VSI_TIR register. The *GL_SWT_L2TAGTXIB.OFFSET* and *GL_SWT_L2TAGTXIB.LENGTH* fields define the location and length of the variable part. Figure 38-9 shows the relationship between the different parameters.

The order of the tags inserted from the port-based tags is according to the order in the bit map. Thus, the first bit set uses the value from the VSI_TIR[0] register. The second bit set uses the value from the VSI_TIR[1] register. The third bit set uses the value from the VSI_TIR[2] register.

Figure 38-9. Tag Insertion



Queue-based tag insertion

For one of the tags, the 10 GbE controller might insert different port-based tags based on the queue from which the packet was sent. If the *ALT_VLAN* bit in the transmit queue context is set, the tag is taken from the alternate tag register; *VSI_TAIR* instead of being taken from *VSI_TIR.*

This capability is available for each VSI in the tag inserted to the *VSI_L2TAGSTXVALID.TIR0INSERTID* tag.

38.21.4.3.3 Received Tag Extraction Rules

The tag extraction from receive packets and insertion in the receive descriptor write back is controlled by the *VSI_TSR.STRIPTAG, VSI_TSR.SHOWTAG,* and *VSI_TSR.SHOWPRIONLY* bit fields. The options supported for each tag are:

- Do nothing
- Remove this tag from receive packets and do not insert into descriptor

- Remove this tag from receive packets and insert into descriptor
- Remove this tag from receive packets and insert only priority bits into descriptor

Up to two L2 tags can be extracted into the Rx descriptor. One of the tags can supply up to 32 bits of data, and the other tag can supply up to 16 bits of data (a total of three 16-bit words maximum). If more than one word needs to be extracted, then 32 bytes descriptors needs to be used (the *Dsize* flag in the receive queue context is set to 32B_Descriptor); otherwise, 16 bytes descriptors can be used (the *Dsize* flag is set to 16B_Descriptor).

The order of the tags extracted to the descriptor is according to the order in the *VSI_TSR.SHOWTAG* bit map and according to the *L2TSEL* field in the queue context. If the *L2TSEL* bit is cleared, the first bit set extracts the value to the *L2TAG1* field in the receive descriptor, the second bit set, extracts the value to the *L2TAG2* (2nd) and *L2TAG2* (1st) fields in the receive descriptor. If the *L2TSEL* bit is set, the first bit set extracts the value to the *L2TAG2* (2nd) and *L2TAG2* (1st) fields in the receive descriptor, the second bit set, extracts the value to the *L2TAG1* field in the receive descriptor.

Note: The *L2TAG2* (2nd) field is used only if *GL_SWT_L2TAGRXEB.LENGTH* is set to 10b or 11b (24 or 32 bits extracted part).

When removal from the packet of a tag is requested, the total L2 tag (determined by the *GL_SWT_L2TAGCTRL [7:0].LENGTH* field), including the EtherType is extracted from the packet.

If the *SHOWIV* bit is set in queue context, the inner VLAN value replaces the tag extracted to *L2TAG2*.

Caution: Extracting two tags should be used only if the software device driver knows which tags are expected in the packets.

The following table describes the content of the tags in the receive descriptor writeback in different cases assuming the first tag is an S-tag and the second tag is a VLAN.

Table 38-160.Tag Extraction Example (Sheet 1 of 2)

SHOWIV	L2TSEL	Packet			L2TAG1	L2TAG2 - 2nd	L2TAG2 - 1st
		S tag present	VLAN present	Inner VLAN present			
0	0	y	y	X	VLAN	S tag	0
		y	n		Not Valid	S tag	0
		n	y		Not Valid	VLAN	0
		n	n		Not Valid	Not Valid	Not Valid
	1	y	y		S tag	VLAN	0
		y	n		S tag	Not Valid	Not Valid
		n	y		VLAN	Not Valid	Not Valid
		n	n		Not Valid	Not Valid	Not Valid

Table 38-160.Tag Extraction Example (Sheet 2 of 2)

SHOWIV	L2TSEL	Packet			L2TAG1	L2TAG2 - 2nd	L2TAG2 - 1st
		S tag present	VLAN present	Inner VLAN present			
1	0	y	X	y	S tag	Inner VLAN	0
		y	X	n	Not Valid	S-tag	0
		n	y	y	Inner VLAN	VLAN	0
		n	y	n	Not Valid	VLAN	0
		n	n	y	Inner VLAN	Not Valid	Not Valid
		n	n	n	Not Valid	Not Valid	Not Valid
	1	y	y	y	S tag	Inner VLAN	0
		n	y	y	VLAN	Inner VLAN	0
		y	y	n	S tag	Not Valid	Not Valid
		n	y	n	VLAN	Not Valid	Not Valid
		y	n	y	S tag	Inner VLAN	0
		y	n	n	S tag	Not Valid	Not Valid
		n	n	y	Not Valid	Inner VLAN	0
		n	n	n	Not Valid	Not Valid	Not Valid

38.21.4.3.4 Tag Handling (Programming Interface)

This section describes the CSR interface available to control the L2 tags handling of the 10 GbE controller. The actual programming of these capabilities is either through an NVM image loading of the device-wide behavior, through port-wide commands such as the Set Port Parameters command for the port-wide behavior or through the Add VSI admin command for the per-VSI behavior.

The 10 GbE controller supports up to 8 types of programmable L2 tags. The types of L2 tags are defined in the NVM and in registers. For each tag the following parameters are defined:

Table 38-161.L2 Tag Control Registers - Global (Sheet 1 of 2)

Register	Field	Description
GL_SWT_L2TAGCTRL [7:0]	ETHERTYPE	The EtherType of the L2 tag.
	ISVLAN	Is this tag a VLAN tag? This information is used to define if priority tagging should be supported for this tag.
	INNERUP	If this bit is set, the UP remapping is done on this field. If this bit is set, then ISVLAN should also be set. If set in multiple tags, then the inner UP is taken from the first tag with this bit set in the packet.
	OUTERUP	If this bit is set, then this is the tag on which the inner-to-outer UP remapping is applied. Should be set only in one tag.
	LENGTH	The length of the L2 tag (not including the EtherType). The length can be 2, 4, 6 or 8 bytes.
	HAS_UP	Defines if this tag includes UP bits that should be used for UP to TC translation. The first such tag found in the packet is used for UP to TC translation.
PRT_L2TAGSEN	NONLAST_TAG	Indicates that there are more tags with the same EtherType expected.
GL_SWT_L2TAGTXIB [7:0]	OFFSET	Describes the offset in the header to which the variable data should be inserted (can be up to 8 bytes).

Table 38-161.L2 Tag Control Registers - Global (Sheet 2 of 2)

Register	Field	Description
GL_SWT_L2TAGDATA0[7:0] GL_SWT_L2TAGDATA1[7:0]	L2TAGDATA	The fixed part to insert in transmit packets.

Each port defines which tags to expect in packets sent and received through this port using the *PRT_L2TAGSEN.ENABLE* field.

Each VSI defines which of the 8 L2 tags are offloaded for its Tx and Rx traffic using the *VSI_L2TAGSTXVALID* and VSI_TSR registers. These registers define which receive tags and transmit tags to handle.

Each VSI can be configured with a different behavior for each of the tags. The possible behaviors relates to the type of tags the software device driver is allowed to insert in transmit packets, the type of tags inserted by the switch, the tags removed from received packets and the tags posted to the receive descriptor write back.

Table 38-162 lists the registers fields used to set the expected behavior for each tag.

Table 38-162.L2 Tag Control Registers - Per VSI

Register	Field	Description
VSI_L2TAGSTXVALID	L2TAG[12]INSERTID[1], L2TAG[12]INSERTID_VALID	Defines the tags for which descriptor-based insertion is supported. The ID is based on the L2 tags mapping listed in Table 38-163.
	TIR[012]INSERTID, TIR[012]_INSERT	Defines the tags for which port-based insertion is supported.
VSI_TAR[2]	ACCEPTTAGGED[n]	A bitmap describing if a packet with tag N is accepted.
	ACCEPTUNTAGGED[n]	A bitmap describing if a packet without tag N is accepted.[3,4]
VSI_TIR_n (n = 0..2)	PORT_TAG_ID	The tag to insert.
VSI_TAIR	PORT_TAG_ID	The alternate tag that can be used instead of VSI_TIR[0].
VSI_TSR	STRIPTAG[n] (n = 0..9)	Defines if the tag should be extracted from the packet.
	SHOWTAG[n] (n = 0..9)	Defines which of the tags should be extracted to the descriptor. Valid only if corresponding bit in STRIPTAG is set. The *SHOWPRIONLY* field defines which part of the tag to extract to the descriptor. At most two of these bits should be set. If more than two bits are set, only the two first are considered.
	SHOWPRIONLY[n] (n = 0..9)	A per-tag bitmap defining which part of the tags to extract to the descriptor. If set, only the priority bits are extracted; otherwise, the entire tag is used. Relevant only if the corresponding bit in SHOWTAG is set.

Notes:

1. In order to enable insertion of priority bits from a descriptor and the VLAN tag value from hardware as listed in Table 38-158. Note that the same ID should be used for L2TAG1INSERTID and TIR0INSERTID or L2TAG2INSERTID and TIR1INSERTID.
2. The tag number in this register relates to the 10 tags defined in Section 38.21.4.3.5.
3. If *GL_SWT_L2TAGCTRL.ISVLAN* is set, admits also priority-tagged packets (VLAN tag = 0b).
4. This bit should be set for all the tags not expected in the packet.

38.21.4.3.5 L2 Tags Configuration

This section describes the value to set in the different registers to implement the 10 GbE controller POR.

Table 38-163 lists the L2 tags that are currently defined.

Table 38-163.L2 Tags

Tag	Tag ID (In Table)	Tag ID In VSI_TAR And VSI_TSR Bitmaps
Reserved	0	2
S-tag	1	3
Outer VLAN	2	4
VLAN	3	5
Reserved		

Table 38-164 lists the configuration for each of the tags.

Table 38-164.L2 Headers Support

Register/Tag	Reserved	S-tag	Outer VLAN	VLAN	Reserved
Index	0	1	2	3	5-7
Global Configuration					
GL_SWT_L2TAGCTRL. ETHERTYPE		0x88A8[1]	0x8100	0x8100	0
GL_SWT_L2TAGCTRL. ISVLAN[2]		0	0	1	0
GL_SWT_L2TAGCTRL. NON_LAST_TAG	0	0	1	0	0
GL_SWT_L2TAGCTRL. INNERUP		0	0	1	0
GL_SWT_L2TAGCTRL. OUTERUP		1	0	0	0
GL_SWT_L2TAGCTRL. LONG		0	0	0	0
GL_SWT_L2TAGCTRL. HAS_UP		1	1	1	0
GL_SWT_L2TAGCTRL. LENGTH		2	2	2	0
GL_SWT_L2TAGTXIB. OFFSET		0	0	0	0
GL_SWT_L2TAGTXIB. LENGTH[3]		01b	01b	01b	0
GL_SWT_L2TAGRXEB. OFFSET		0	0	0	0
GL_SWT_L2TAGRXEB. LENGTH[3]		01b	01b	01b	0
GL_SWT_L2TAGDATA0, GL_SWT_L2TAGDATA1	All zeros.				
Per Port Configuration Controlled By Firmware					
PRT_L2TAGSEN.ENABLE		0/1	0/1[1]	1	0
Per VSI Configuration Controlled By Add VSI Command					
VSI_TSR. STRIPTAG		0/1[4]	0	0/1[5]	0
VSI_TSR.SHOWTAG		0/1	0	0/1[6]	
VSI_TSR.SHOWPRIONLY		0	0	0/1[7]	
VSI_TAR.ACCEPTTAGGED		0/1	1	0/1[8]	
VSI_TAR.ACCEPTUNTAGGED		1	1	0/1	

Notes:

1. If the *Channel Identifier* field in the features enable word in the EMP SR settings module header is set, the value of tag 1 is 0x8100 and tag 2 (outer VLAN) is not available.
2. The ISVLAN must be set on one tag and only on one tag (the inner VLAN).
3. 00b = 8 bits, 01b = 16 bits, 10b = 24 bits, 11b = 32 bits.
4. Should be set, unless cascaded PE VSI without offload.

5. Should be set if VSI is VLAN aware and VM requested VLAN extraction offload.
6. Should be set if VSI in non-VLAN aware but is DCB aware.
7. Should be set for VLAN aware VSIs.
8. Should be set if VSI must add VLAN.

Note: *PRT_L2TAGSEN.ENABLE* should not enable outer VLAN and S-tag concurrently.

38.21.4.4 VLAN Handling

This section describes the handling of IEEE 802.1Q VLAN tags based on the features previously described.

There can be up to two VLAN tags in the outer header of a packet identified as VLAN (tag index 3) and outer VLAN (tag index 2).

In addition, the tunneled header may also include a VLAN. The handling of the tunneled VLAN is described in Section 38.21.4.4.3.

Each port can be set to expect packets with outer VLAN using the Set Port Parameters admin command. When enabled, a packet with a single VLAN is treated as a packet with outer VLAN only; otherwise, a single VLAN in a packet is treated as inner VLAN.

In any case, the outer VLAN is not part of the forwarding decision and should be handled by the software device driver or by the MC both in transmit and receive. There is no offload of outer VLAN insertion or extraction.

The sections that follow describe the handling of the inner VLAN.

In UDP tunnels, an internal VLAN might also be present. The handling of this VLAN is described in Section 38.21.4.4.3.

See the section about UP translation in inner and outer VLAN for more detail.

38.21.4.4.1 Transmit Flow

This section describes the handling of VLAN as part of the flow of packets sent by the host, the MC or the EMP.

Tag Insertion

A VLAN tag can be inserted to the packets in three ways:

1. As part of the packet buffer.
2. As part of the transmit descriptor in the *L2TAG1* field if the *IL2TAG1* field is set.
3. By the device from the VSI context.

The two first options are enabled if the VSI is allowed to add a VLAN tag by the VLAN driver insertion mode in the Add VSI command. If a packet is sent with a VLAN tag from a VSI not allowed to add a tag, it is dropped.

Note: If the *IL2TAG_IL2H* field in the transmit context descriptor is set, *L2TAG1* represents the inner VLAN and regular VLAN insertion from the descriptor is not available. See Insertion of Tunneled VLAN from Descriptor for details on the inner tag insertion. Inner tag insertion is allowed irrespective of the VLAN configuration in the *Add VSI* command.

The third option is enabled by setting the *Insert PVID* in the same command. The tag to insert is defined in the *PVID+Default UP* field of the command.

Note: Setting both *Insert PVID* and VLAN driver insertion mode to enable software to insert VLAN is not allowed.

It is expected that MC traffic arrives with the right VLAN tagging.

VLAN Anti-Spoofing

After the packet is VLAN tagged by one of the previous methods, if the *Enable VLAN anti spoof* bit in the Add VSI command is set, it is compared to the ingress VLAN list and dropped if the inserted VLAN is not in the list.

VLAN Filtering

If the packet passed the previous stage, the VLAN tag is used as part of the forwarding process. It is compared to the MAC, VLAN filters added by the Add MAC, VLAN pair command to determine if the packet should be sent to a local address or should be sent to the network. It is also compared to the PRT_MNG_MAVTV filters as part of the manageability filtering to define if it should be sent to the MC.

38.21.4.4.2 Receive Flow

This section describes the handling of VLAN as part of the flow of packets received by the host, the MC or the EMP.

VLAN Filtering

When a packet is received from the network (or from the host) the VLAN tag is used as part of the forwarding process. It is compared to the MAC, VLAN filters added by the Add MAC, VLAN pair command and compared to the VLAN egress filtering set by the Add VLAN admin command to determine if the packet should be sent to a local VSI. It is also compared to the PRT_MNG_MAVTV filters as part of the manageability filtering to define if it should be sent to the MC.

VLAN Extraction

Before a packet is stored in the host memory the VLAN tag might be stripped and optionally stored in the receive descriptor. The action done is defined per VSI in the *VLAN and UP expose mode (Rx)* field in the Add VSI command. The possible actions are:

- Show VLAN and UP in descriptor (legacy behavior)
- Hide VLAN show UP in descriptor (VLAN ID exposed as 0)
- Hide VLAN and UP
- Do nothing (leave VLAN in packet)

If the VLAN or the UP are exposed in the descriptor then it shows in the *L2TAG1* field and the *L2TAG1P* flag is set. If L2TSEL bit is set, VLAN tag is extracted to the L2TAG2 (1st) field in the receive descriptor instead. In this case, 32 bytes descriptors must be used (*DSize* field in the receive queue context must be set).

Note: When a packet is sent to the MC the VLAN is kept in the packet.

38.21.4.4.3 VLAN in Tunnel Packets

During MAC in UDP and MAC in GRE encapsulations, a VLAN within the tunneled MAC header might also be present.

Insertion of Tunneled VLAN from Descriptor

The insertion of this tag is controlled via the *IL2TAG_IL2H* field in the transmit descriptor. When set, the L2TAG2 field in the context descriptor contains the tunneled VLAN. In this case, the values of *VSI_L2TAGSTXVALID.L2TAG2INSERTID* and *VSI_L2TAGSTXVALID.L2TAG2INSERTID_VALID* fields are ignored.

This mode is enabled irrespective of the VLAN configuration in the *Add VSI* command.

Extraction of Tunneled VLAN to Descriptor

The *SHOWIV* field in the receive queue context controls the extraction of an internal VLAN to the receive descriptor. If set, the tunneled VLAN is inserted in L2TAG2 (1st) field of the receive descriptor write back. If L2TSEL bit is cleared, the inner VLAN tag is extracted to the L2TAG1 field in the receive descriptor instead. When the SHOWIV field is set and L2TSEL bit is set, 32 bytes descriptors must be used (*DSize* field in the receive queue context must be set).

Tunneled VLAN in Pass Through Traffic

Tunneled VLAN is not offloaded (not inserted nor extracted) for MC pass-through traffic.

38.21.4.5 Port-Based VLAN

The 10 GbE controller supports a port-based VLAN feature by enabling any VSI to specify the default VLAN that it belongs to. The port-based VLAN association is done when a packet received on VSI is untagged or priority tagged and protocol VLAN association is not done. Port-based VLANs map packets received on a given VSI with the corresponding port-based VLAN identifier called PVID. The PVID list should be programmed with the port VLAN IDs for each VSI.

The port VLAN list is provided which is part of the VSI context. Table 38-165 lists the port VLAN parameters in the port VLAN list and the matching parameters in the Add VSI command.

Table 38-165.Port VLAN List and VSI Parameters (Sheet 1 of 2)

Port Based VLAN list	Add VSI parameter	Notes
VSI number	VSI Number	Returned by firmware in response.
PVID	PVID + Default UP	
Default UP		
Admit.1Q tagged only	VLAN driver insertion mode	
Admit untagged/Priority tagged only		
Admit all		
Ingress VLAN check enable	Enable VLAN anti spoof	
Insert PVID	Insert PVID	
Expose VID and UP of received packets	VLAN and UP expose mode (Rx)	
Expose UP only of received packets		
Do not expose VID or UP of received packets		

Table 38-165.Port VLAN List and VSI Parameters (Sheet 2 of 2)

Port Based VLAN list	Add VSI parameter	Notes
Ingress UP translation table	Ingress UP translation table	Refer to the Transmit Functionality section for more detail.
Egress UP translation table	Egress UP translation table	Refer to the Transmit Functionality section for more detail.

The switch should insert the PVID to the packet if the *Insert PVID* parameter is set for the VSI and replace the UP bits according to the algorithm described in the Transmit Functionality section.

Once the VLAN ID association is made, the packet forwarding is performed using the (VLAN, MAC) forwarding table and VLAN membership table.

38.21.4.6 S-tag Handling

This section describes the handling of IEEE 802.1Qbg S-tags based on the features previously described.

There can be up to one S-tag in a packet (tag index 1). The sections that follow describe the handling of the S-tag.

Note: The S-tag EtherType can be either 0x88A8 or 0x8100 according to the Channel Identifier field in the Features enable word in the EMP SR Settings Module Header (0x0 = 0x88A8, 0x1=0x8100).

38.21.4.6.1 Transmit Flow

This section describes the handling of S-tag as part of the flow of packets sent by the host, the MC or the EMP.

There is no anti-spoofing capability for S-tags.

Tag Insertion

An S-tag can be inserted to the packets in three ways:

1. As part of the packet buffer.
2. As part of the transmit descriptor in the *L2TAG2* field if the *IL2TAG2* field is set.
3. By the device from the VSI context.

The first two options are enabled if the VSI is allowed to add an S-tag by the clearing the *S-tag insert enable* field and setting the *Accept tag from host* field in the Add VSI command. If a packet is sent with an S-tag from a VSI not allowed to add a tag, it is dropped.

The third option is enabled by setting the *S-tag insert enable* in the same command. The tag to insert is defined in the *S-tag* field of the command.

Note: Setting both *S-tag insert enable* and *Accept tag from host* to enable software to insert VLAN is not allowed.

38.21.4.6.2 Receive Flow

This section describes the handling of S-tag as part of the flow of packets received by the host, the MC or the EMP.

S-tag Extraction

Before a packet is stored in the host memory the S-tag might be stripped and optionally stored in the receive descriptor. The action done is defined per VSI in the *S-tag extract mode* field in the Add VSI command. The possible actions are:

- Show S-tag and UP in descriptor (legacy behavior) — used for cascaded port virtualizer with offload.
- Hide S-tag and UP.
- Do nothing (leave S-tag in packet) — used for cascaded port virtualizer without offload.

If the S-tag is exposed in the descriptor then it shows in the *L2TAG2* field and the *L2TAG2P* flag is set. In this case, 32 bytes descriptors must be used (*DSize* field in the receive queue context must be set).

Note: If the *SHOWIV* field is set in the queue context, then the *L2TAG2* field is allocated for the inner VLAN of tunnel packet. In this case, the S-tag is never inserted in the descriptor.

38.21.5 User Priority Bits (802.1p) Handling

The UP bits are used to differentiate between multiple classes of traffic. The 10 GbE controller supports multiple use cases related to the handling of the UP bits:

1. An operating system that is not DCB/traffic type aware and uses a single TCID.
2. An operating system that is traffic type aware, but not DCB aware. It can distribute the traffic to different queues, but cannot tag them with the right UP. This case refers to LAN and iWARP flows. The operating system is not aware but the software device driver knows the difference and hence can assign the different types of traffic to different flows. There is no DCBX agent and hence the software device driver does not know the UP or TC for flows.
3. An un-trusted operating system that might set UP bits of TCs is not allowed to use.
4. An operating system that can use a single queue per TCID, but would like to use more UPs in order to gain from the differentiated QoS in the network.

The following sections describes the handling of UP bits in transmit and receive to support the previous use cases.

38.21.5.1 Transmit Functionality

Each transmit packet is assigned a UP, either by the software device driver or as part of the port VLAN table as described in Section 38.21.4.5.

This UP is translated using two different translation vectors.

The first vector is specific to a VSI and is part of the port VLAN table (*transmit UP translation table - VSI_TUPR*). This vector reflects the mapping of the user priorities as seen by the operating system to the user priorities as seen by the network.

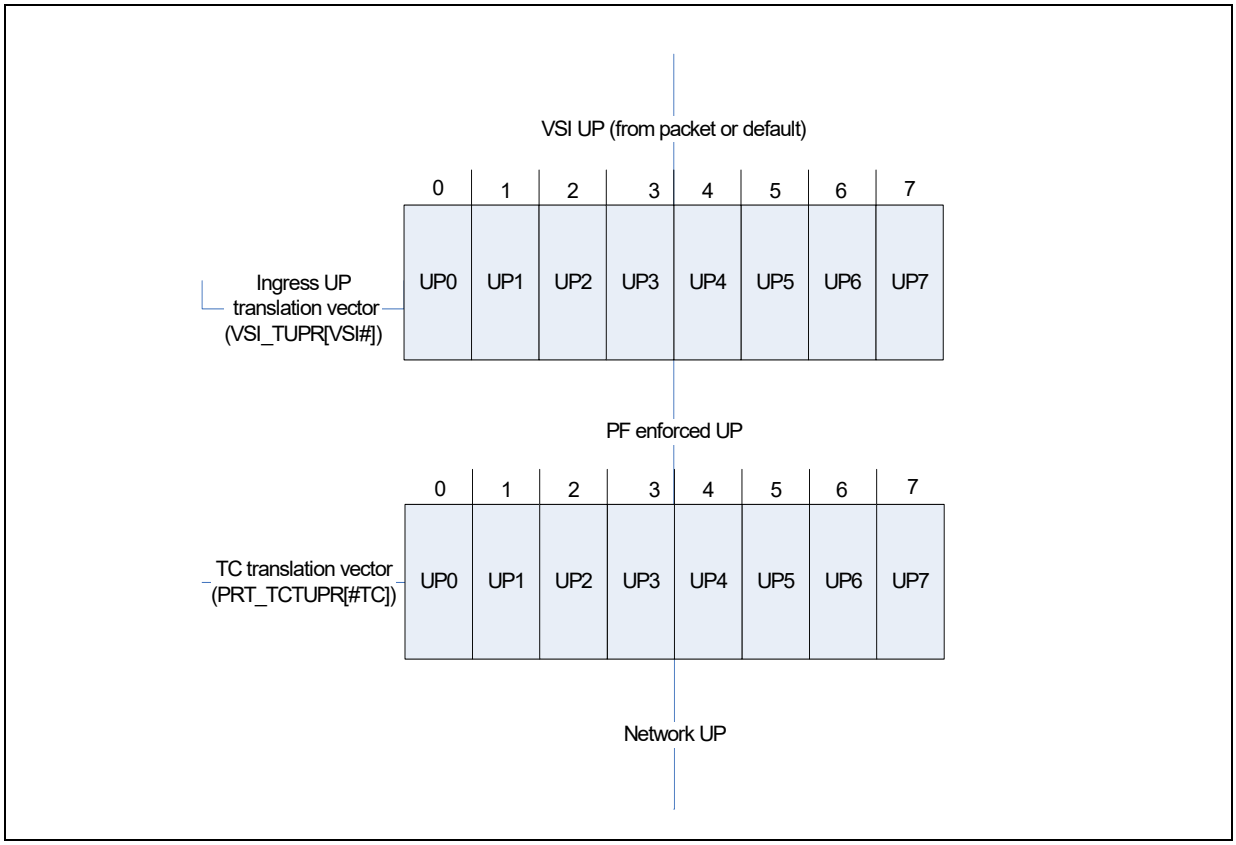
The second vector is specific to a TCID in a physical port. It translates the UP received from the previous translation to a UP matching the TCID to which the packet is associated. These vectors can be configured through the *PRT_TCTUPR* registers.

Note: There is no drop of packets due to a wrong 802.1p tagging, if a wrong tag is requested the tag is replaced. Thus, this functionality supports both UP anti-spoofing and port based UP.

Untagged packets are kept as is. No translation is done on them.

Figure 38-10 shows the algorithm.

Figure 38-10.Example: Non-VLAN Aware, Non-DCB Aware Operating System



Assume a VF that queues LAN traffic and iSCSI traffic to different queues, but is not VLAN aware. In this case, the packet is sent by the software device driver with no VLAN and the default UP set in the port based VLAN (such as 0) is the initial UP of the packet. In this case, the *Ingress UP translation vector* (*VSI_TUPR*) is not relevant and might be for example (0,1,2,3,4,5,6,7) or (0,0,0,0,0,0,0,0). The *TC translation vector* (*PRT_TCTUPR)* for LAN might be (0,1,2,0,0,0,0,0) and the TC table for storage might be (4,4,4,4,5,4,4,4). So a LAN packet goes out with a UP of 0, and a storage packet goes out with a UP of 4.

Example: Non-DCBx tagging operating system

Assume a VF that tags LAN high priority traffic with a UP of 4 and LAN low priority traffic with a UP of 3 and the storage traffic as UP 7. In this case, the VF translation table might be (0,1,2,0,2,4,4) and the TC tables as previously described. The low priority LAN goes out with a UP of 0, the high priority LAN with a UP of 2 and storage with a UP of 4.

The tag on which the transmit UP translation is done is identified by setting the *INNERUP* bit in the matching GL_SWT_L2TAGCTRL register.

The default mapping of both tables is identity mapping (for example, mapping a UP to itself). If UP translation is not needed these mappings should not be changed.

38.21.5.1.1 Transmit Outer Tag User Priority

The transmit outer tag user priority can be handled in three different ways:

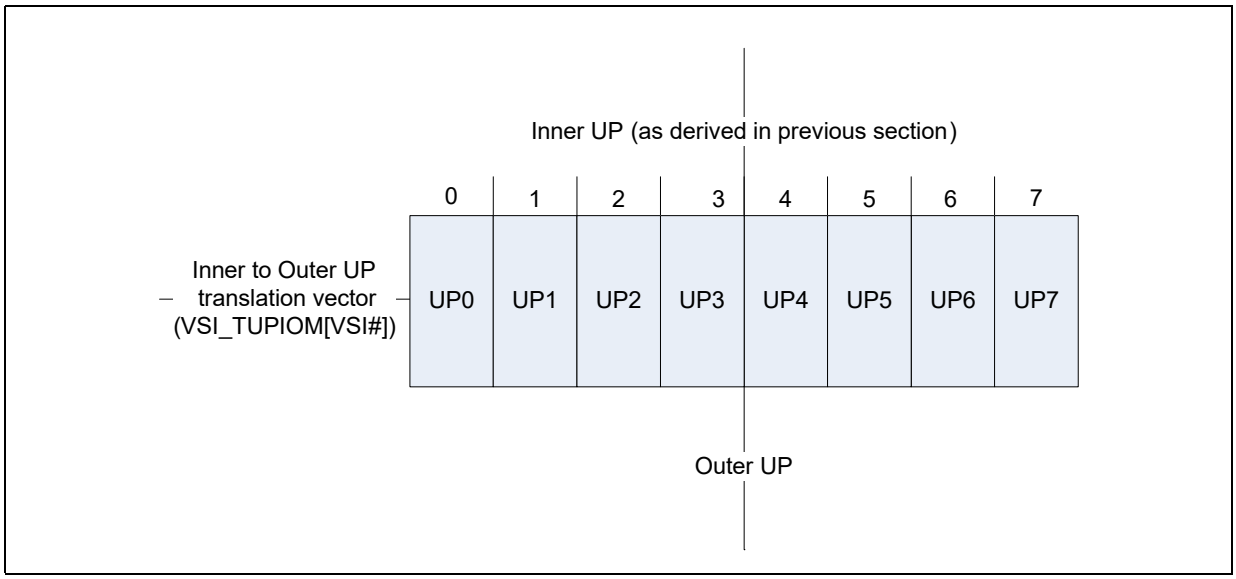
1. If the outer tag is received in the packet or the descriptor (for example, cascaded S-comp), the UP should be part of the packet sent by the software device driver or inserted from the descriptor.
2. If the outer tag is inserted by hardware then the UP might be either:
   a. The UP defined as part of the inserted tag in the VSI_TIR register.
   b. A translation of the regular VLAN UP as defined in the *VSI_TUPIOM* register.

If the outer tag is defined as *OuterUP* in the GL_SWT_L2TAGCTRL register, then the UP is based on the first tag for which the *InnerUP* field is set in the GL_SWT_L2TAGCTRL register (usually the VLAN tag) (option 2.b); otherwise, it is based on the default (option 2.a).

The tag to which translation is applied can be either S-tag or external VLAN.

Figure 38-11 describes the translation process.

Figure 38-11.Tx Inner-to-Outer UP Translation



38.21.5.2 Receive Functionality

38.21.5.2.1 VLAN UP Translation

The priority bits of inner VLAN tag in received packets might be remapped using the *Ingress UP translation table* in the *Add VSI* command. This means the operating system might get packets with 802.1p priority bits reflecting the local configuration and not the link configuration. By default (ingress UP translation section is not valid) the translation is one-to-one, meaning that the received UP is not translated.

38.21.5.2.2 VLAN UP Exposure to Software Device Driver

The exposure of the received UP to operating systems is defined by the awareness of the operating system controlling the VSI as follows:

1. For a monolithic operating system, VMM or for a guest operating system that is VLAN aware, the UP is exposed as part of the VLAN as requested by software device driver (either in the packet or in the descriptor).
2. If the guest operating system is DCB aware but not VLAN aware, it gets the priority bits as part of the receive descriptor in the VLAN tag field. The VLAN ID is zero, but the priority bits are valid.
3. A guest operating system that is not VLAN aware and not DCB aware does not get the user priority bits (UP) at all.

The configuration of the UP removal is part of the port VLAN configuration described in Section 38.21.4.5 and in the *Add VSI* command VLAN handling section.

38.21.6 Frame Formats

This section describes the packet formats supported by the device. It is structured by layers (L2, L3, and L4), providing the details per each layer. Packets that do not conform with the described formats are handled as L2 packets.

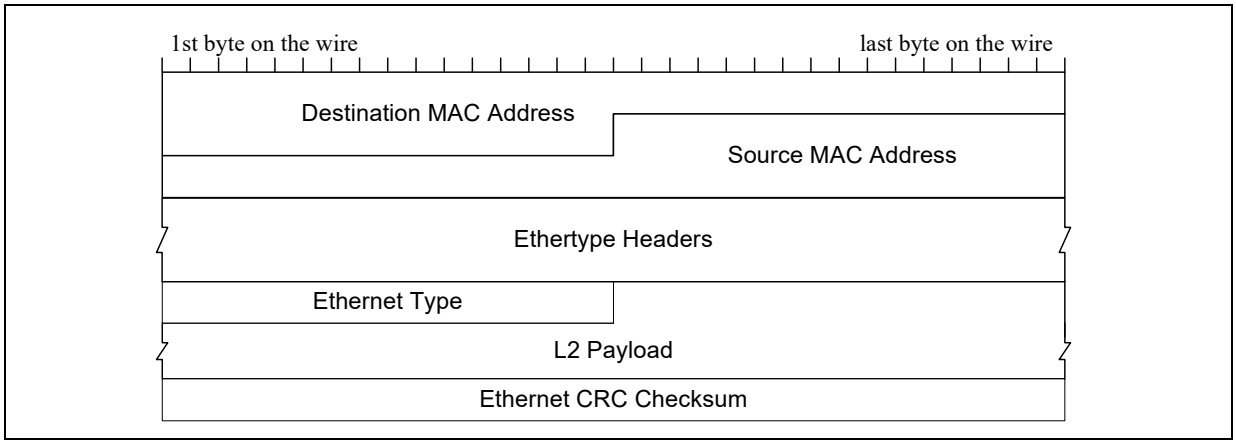
- The following restrictions apply on the lengths of parsed packets: The device parses headers in the first 480 bytes of the packet only. If the header (as defined in this section) extends beyond 510 bytes, the packet is handled as an L2 packet with a packet type of abort (0xFF).
- Each single header is limited to 255 bytes (unless restricted further in the text). If the header (as defined in this section) extends beyond 255 bytes, the packet is handled as an L2 packet with a packet type of abort (0xFF).
- Aborted packets by the parser are handled as L2 packets by the switch: VSI decision is made by the outer MAC, optional VLAN tag in the outer L2 header and an optional S-tag.

38.21.6.1 L2 Packet Formats

IEEE 802.3 and 802.3 SNAP packets are considered and treated as Layer 2 packets. The 10 GbE controller does not identify such packets explicitly in any way.

38.21.6.1.1 Standard L2 Packet Format

Figure 38-12.Standard L2 Frame Format



38.21.6.1.2 Serial EtherTypes (L2-tags)

Table 38-166 lists the supported EtherType headers. The order of EtherType headers in the L2 header are shown in Figure 38-12.

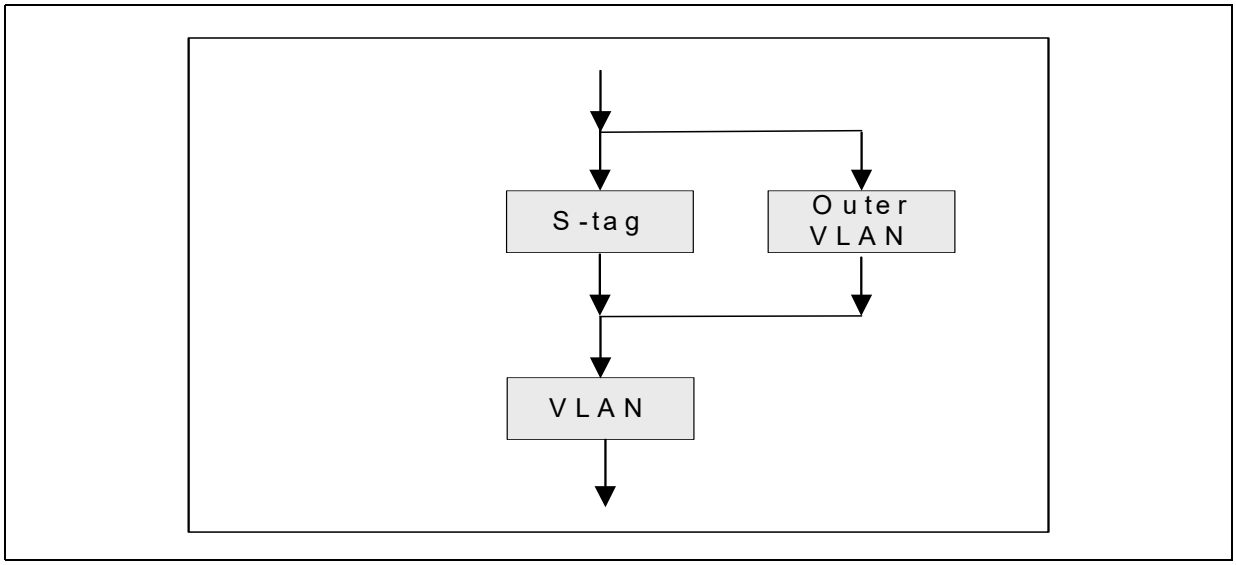
Table 38-166.List of Serial EtherType Headers

EtherType Header	Value	Length[1]	Defined In
802.1Q S-tag	0x88A8	2 words	IEEE 802.1Q clause 9.
Outer VLAN	Loaded from NVM (such as 0x8100, 0x9100, and 0x9200)	2 words	IEEE 802.1Q clause 9.
VLAN	0x8100	2 words	IEEE 802.1Q clause 9.

Notes:

1. EtherType headers are an integer number of words. Length includes the *Type* field.

Figure 38-13.Order of L2 EtherType Headers



38.21.6.1.3 Last EtherType (Upper Layer Protocol)

The Upper-layer Protocol (ULP) following the L2 header is identified by the *Ethernet Type* field (see Figure 38-12).

Table 38-167 lists the EtherTypes supported by the device along with their index value. A value of 00-00 means the entry is reserved.

Table 38-167.List of Ethernet Types (Sheet 1 of 2)

Index	Value	Purpose	Format of ULP
0	88-F7	IEEE 1588 (IEEE 802.1AS)	
1	89-14	Reserved	
2	00-00	Reserved	
3	00-00	Reserved	
4	88-CC	IEEE 802.1ab (LLDP)	IEEE P802.1AB specification.
5	89-40	Reserved	
6	00-00	Reserved	
7	88-47	Reserved	

Table 38-167.List of Ethernet Types (Sheet 2 of 2)

Index	Value	Purpose	Format of ULP
8	88-8E	IEEE 802.1X (Network Access Control)	N/A
9	08-06	ARP	See Section 38.21.6.2.6.
10	00-00	Reserved	
11	00-00	Reserved	
12	08-00	IPv4	See Section 38.21.6.2.1.
13	86-DD	IPv6	See Section 38.21.6.2.2.
14	89-06	Reserved	
15	00-00	Reserved	

38.21.6.1.4 MPLS Header(s)

The 10 GbE controller identifies MPLS headers. It skips these headers processing the rest of the packet and providing all stateless offloads, switch filtering and the classification filters. Table 38-168 lists which fields of the MPLS header are digested by the device and Figure 38-14 shows the structure of the MPLS header.

Packets with MPLS are only supported for LAN traffic. Not supported for RDMA offload.

Identification of Next Protocol

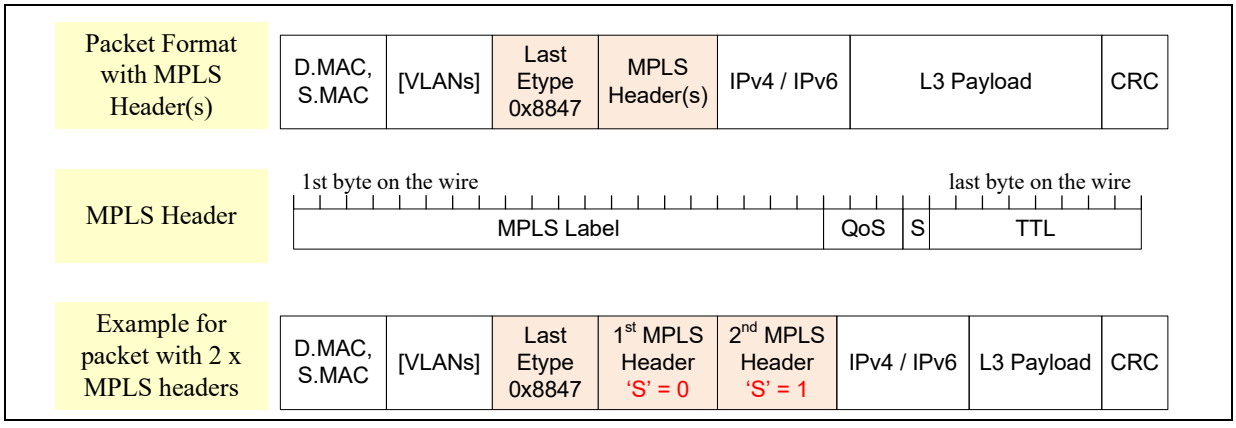
The 10 GbE controller supports IP after MPLS by evaluating the nibble (4 bits) immediately following the last MPLS tag.

- If it equals to 0x4 then the next protocol is identified as IPv4
- If it equals to 0x6 then the next protocol is identified as IPv6
- Else, packet is handled as L2 packet

Table 38-168.MPLS Header Structure

Bit Offset	Size [bits]	Field	Value	Action	Comment
0	20 bit	MPLS Label	Variable	Ignore	
20	3 bit	QoS and ECN	Variable	Ignore	
23	1 bit	S - Bottom of Stack	0 / 1	Check	Identify the last MPLS header when 'S' = 1b
24	8 bit	TTL	Variable	Ignore	

Figure 38-14.MPLS Header



38.21.6.1.5 LLDP Frame Format

The general structure of an LLDP packet is listed in Table 38-169.

Table 38-169.Structure of LLDP Frame

LLDP Header	
LLDP EtherType	LLDPDU

The LLDPDU contains an ordered sequence of three mandatory TLVs followed by zero or more optional TLVs plus an end of LLDPDU TLV, as listed in Table 38-170.

Table 38-170.Structure of LLDP PDU

Mandatory	Mandatory	Mandatory				Mandatory
Chassis ID TLV	Port ID TLV	Time to Live TLV	Optional TLV	...	Optional TLV	End of LLDPDU TLV

Each TLV has the structure shown in Figure 38-171.

Table 38-171.Structure of LLDP TLV

TLV Header		
7 bits	9 bits	0 <= n <= 511 octets.
TLV type	TLV information string length	TLV information string.

The basic format for organizationally specific TLVs is shown in Table 38-172.

Table 38-172.Structure of LLDP Organizationally Specific TLV

TLV Header		TLV information string - 0 <= n <= 511 octets		
7 bits	9 bits	3 octets	1 octet	0 <= n <= 507 octets
TLV type	TLV information string length	Organizationally Unique Identifier (OUI)	Organizationally defined subtype	

The following table lists the applicable organizationally specific TLVs:

Table 38-173.List of Organizationally Specific TLV

TLV	OUI Value	Subtype Value
DCBx	00-80-C2	09, 0A, 0B, 0C

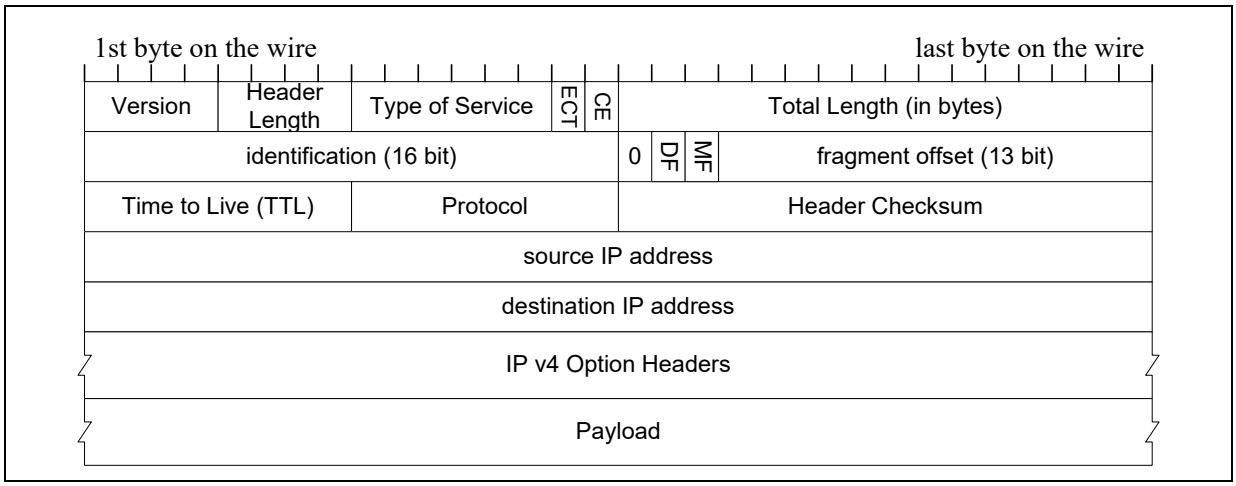
38.21.6.1.6 Other L2 Packets

Refer to the Magic and link control packets sections for more details.

38.21.6.2 L3 Packet Formats

38.21.6.2.1 IPv4 Datagram

Figure 38-15.IPv4 Datagram



Version — The *Version* field is set to 0x4 for IPv4 header.

Header Length — The length of the IP header defined in Dword units.

Type of Service — TOS (6 bits): The T*ype Of Service* field is used to indicate the quality of service with which this datagram is to be delivered by the internetwork routers.

IP Congestion Flags — CE, etc.

Total Length — The total length is the size of the IP datagram (IP header and payload) in byte units.

Identification — ID: The *Identification* field identifies a specific IP packet sent between a source and destination node. The sending host sets the *Identification* field's value, and the field is incremented for successive IP datagrams. The *Identification* field is used to identify multiple fragments of an original IP datagram.

Fragment Parameters — Fragment offset, *More Fragment*(s) flag and *Disable Fragmentatio*n flag.

Time to Live — TTL: The *TTL* field indicates the number of links that this IP datagram can travel before an IP router discards it.

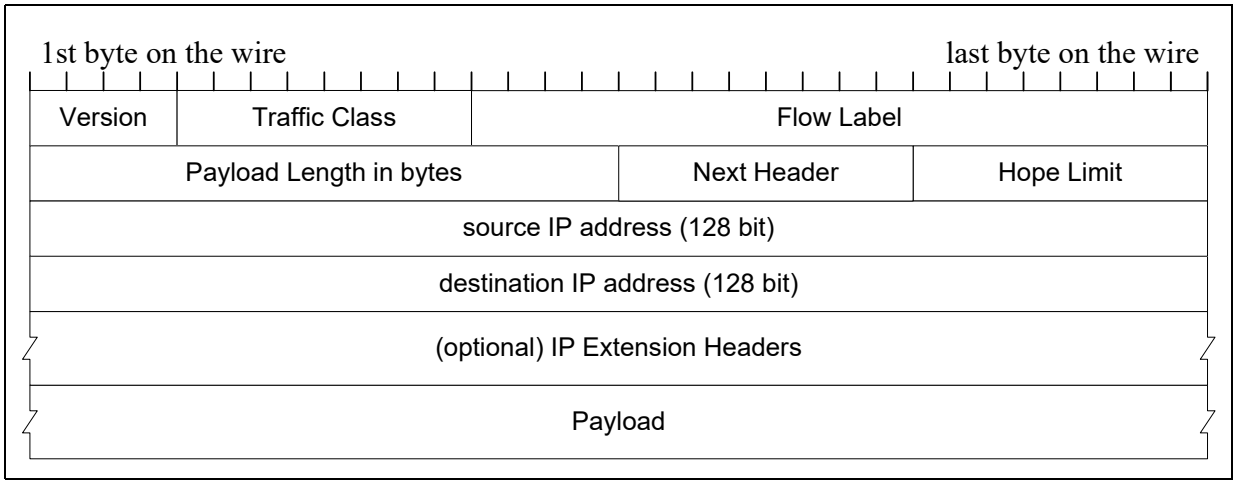
Protocol (next header) — The *Protocol* field indicates the next protocol encapsulated within the IP layer. See Section 38.21.6.2.3 for the list those protocols that can be offloaded by the 10 GbE controller.

Header Checksum — The *Checksum* field is a 16 bit one's complement of the one's complement sum of all 16 bit words in the IP header.

Source and Destination IP Addresses — 2- x 32-bit IP addresses.

38.21.6.2.2 IPv6 Datagram

Figure 38-16.IPv6 Datagram



- Version — The *Version* field is set to 0x6 for IPv6 header.
- Traffic Class and Flow Label — The TC and flow label are used for QOS support.
- Payload Length — The length of the IPv6 payload such as the rest of the packet following the IPv6 header, in byte units. Note that any IPv6 extension headers are considered part of the payload.
- Next Header (Protocol) — The *Next Header* field indicates the next protocol encapsulated within the IP layer. See Section 38.21.6.2.3 for the list those protocols that can be offloaded by the 10 GbE controller.
- Hope Limit (TTL) — The *Hope Limit* field indicates the number of links that this IP datagram can travel before an IP router discards it.
- Source and Destination IP Addresses — 2- x 128-bit IP addresses.

IPv6 Extension Headers

The IPv4 option headers that are included as part of the IP header are replaced in IPv6 by separate extension headers per option. Table 38-174 lists those most used IPv6 extension and their recommended ordering in the packet (shown in the figures that follow). The 10 GbE controller identifies these headers, skips them and then parses the rest of the packet for its processing (unless specified differently in Table 38-174).

Table 38-174.IPv6 Extension Headers and their Recommended Ordering (Sheet 1 of 2)

Header (Protocol)	Next Header Value	Header Length And Header Length Field Offset
Hop-by-Hop Options	0	Variable length field defined in 8-byte units excluding the first 8 bytes.
Destination Options	60	Variable length field defined in 8-byte units excluding the first 8 bytes. Can be located either at this location or before the mobility header.
Routing header	43	Variable length field defined in 8-byte units excluding the first 8 bytes.
Fragment header	44	Length is always 8 bytes. The 10 GbE controller does not continue to parse the rest of the packet.
Authentication header	51	Length Field Is At Byte 1. Define the header length minus 2 in 4-byte units. When this header is found, the 10 GbE controller does not continue to parse the rest of the packet. Applicable for IPv4 as well.

Table 38-174.IPv6 Extension Headers and their Recommended Ordering (Sheet 2 of 2)

Header (Protocol)	Next Header Value	Header Length And Header Length Field Offset
Encapsulating Security Payload	50	Length is 8 bytes plus variable length of initial value plus a trailer. When this header is found, the 10 GbE controller does not continue to parse the rest of the packet. Applicable to IPv4 as well.
Destination Options	60	Variable length field defined in 8-byte units excluding the first 8 bytes. Can be located either at this location or before the routing header.
Mobility Header	135	Variable length field defined in 8-byte units excluding the first 8 bytes.
No Next Header	59	When no next header type is found, the rest of the packet is not processed.

Figure 38-17.IPv6 Hop-by-hop Extension Header

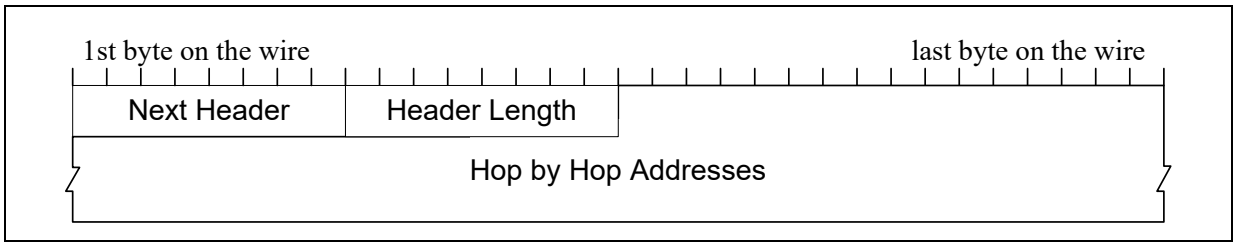


Figure 38-18.IPv6 Routing Header

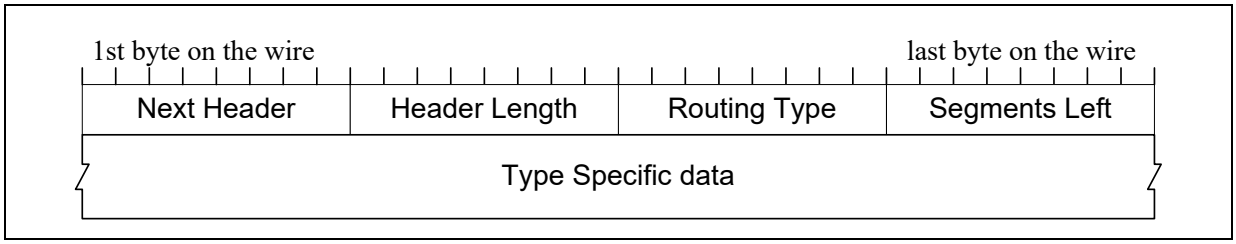


Figure 38-19.IPv6 Fragment Header

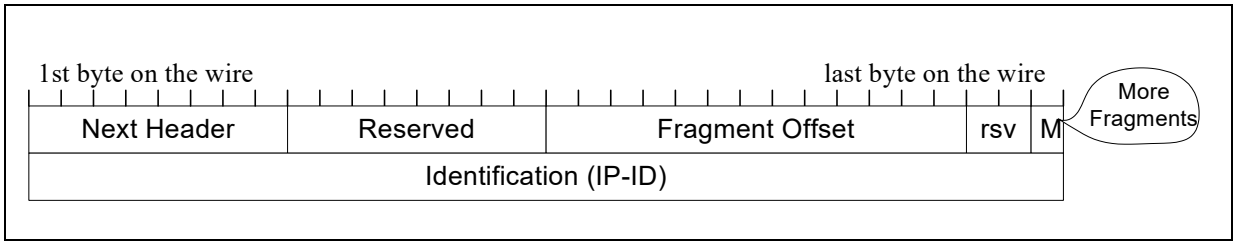


Figure 38-20.IPv6 Destination Options

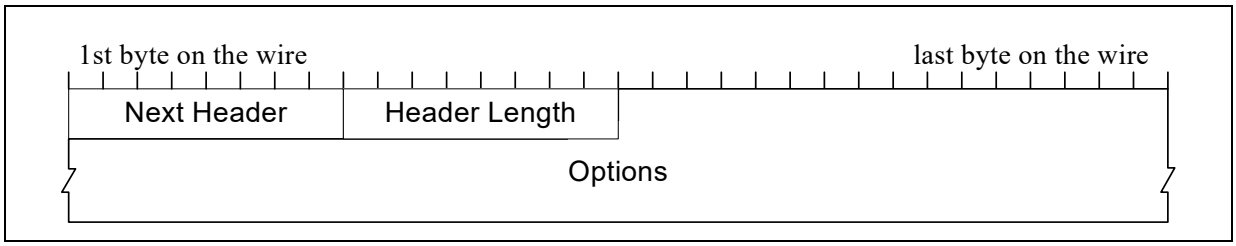
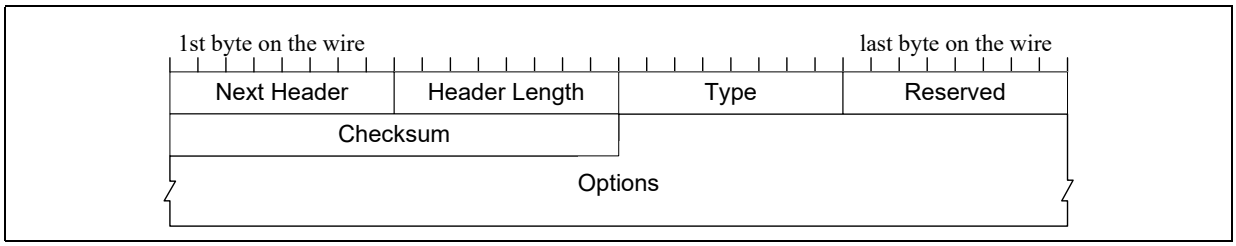


Figure 38-21.IPv6 Mobility Header



38.21.6.2.3 Protocol Headers (Next Header)

Table 38-175 lists the encoding of the *Next Header Type* field and information on determining each header type's length recognized by the 10 GbE controller according to the recommended ordering.

Table 38-175.Header Type Encoding and Lengths

Header (Protocol)	Next Header Value	Header Length And Header Length Field Offset
IPv4	4	Length field is at bits[7:4], defined in 4-byte units.
IPv6	41	Length is always 40 bytes.
Authentication	51	Length field is at byte 1, defines the header length minus 2 in 4-byte units.
Encapsulating Security Payload	50	Length is 8 bytes plus variable length of initial value plus a trailer.
TCP	6	Length field is at byte 12, bits [7:4], defined in 4-byte units.
UDP	17	Length is always 8 bytes.
ICMP	1	Length is always 8 bytes.
ICMPv6	58	Length is always 4 bytes.
SCTP	132	Length is always 12 bytes.
No Next Header	59	When no next header type is found, the rest of the packet is not processed.

38.21.6.2.4 ICMP Datagram

ICMP packets are used as part of the manageability filtering.

Table 38-176.ICMP Packet Format

	MSB..... . . . .LSB 31 24 First byte on the wire	MSB 23	LSB 16	MSB 15	LSB 8	MSB.. 7 Last byte on the wire	... . . LSB 0
0	Type	Code	ICMP Header Checksum				
4	Payload						

Table 38-177 lists the processing done to identify ICMP packets:

Table 38-177.IPv4 Packet Structure and Processing (Sheet 1 of 2)

Offset	# Of Bytes	Field	Value	Action	Comment
0	6	Destination Address		Compare	MAC header. Processed by main address filter.
6	6	Source Address		Skip	
12	T=(0/4)	Possible S-tag		Skip	
12 + T	S=(0/4)	Possible VLAN Tag		Compare	Processed by main address filter.

Table 38-177.IPv4 Packet Structure and Processing (Sheet 2 of 2)

Offset	# Of Bytes	Field	Value	Action	Comment
12 + T + S	2	Type	0x0800	Compare	IPv4
14 + T + S	1	Version/ HDR Length	0x4X	Compare	Check IPv4
15 + T + S	1	Type of Service	-	Ignore	
16 + T + S	2	Packet Length	-	Ignore	
18 + T + S	2	Identification	-	Ignore	
20 + T + S	2	Fragment Info	-	Ignore	
22 + T + S	1	Time To Live	-	Ignore	
23 + T + S	1	Protocol	0x1	Compare	ICMP
24 + T + S	2	Header Checksum	-	Ignore	
26 + T + S	4	Source IP Address	-	Ignore	
30 + T + S	4	Destination IP Address	-	Ignore	
34 + T + S	1	ICMP Type	-	Ignore	
35 + T + S	1	ICMP Code	-	Ignore	
36 + T + S	2	ICMP Header Checksum	-	Ignore	
38 + T + S	F	ICMP Payload	-	Ignore	

The 10 GbE controller does not parse the ICMP header and only detects the IP next protocol as ICMP.

38.21.6.2.5 ICMPv6 Datagram

ICMP packets are used as part of the manageability filtering and as part of proxying capabilities

Table 38-178.ICMPv6 Packet Format

	MSB..... . . . .LSB 31 24 First byte on the wire	MSB 23	LSB 16	MSB 15	LSB 8	MSB ... . . LSB 7 0 Last byte on the wire
0	Type	Code		ICMPv6 Header Checksum		
4	Payload					

The 10 GbE controller supports filtering of the following ICMPv6 packets.

Neighbor discovery packets:

1. 0x86 (134d) — Router Advertisement
2. 0x87 (135d) — Neighbor Solicitation
3. 0x88 (136d) — Neighbor Advertisement
4. 0x89 (137d) — Redirect

MLD packets:

1. 0x82 (130d) — MLD Query
2. 0x83 (131d) — MLDv1 Report
3. 0x84 (132d) — MLD Done
4. 0x8F (143d) — MLDv2 Report

Table 38-179 lists the processing done to identify ICMPv6 packets.

Table 38-179.ICMPv6 Packet Structure and Processing

Offset	# Of Bytes	Field	Value	Action	Comment
0	6	Destination Address		Compare	MAC header. Processed by main address filter.
6	6	Source Address		Skip	
12	T=(0/4/8)	Possible S-tag		Skip	
12 + T	S=(0/4)	Possible VLAN Tag		Compare	Processed by main address filter.
12 + T + S	2	Type	0x86DD	Compare	IPv6
14 + T + S	1	Version/ Priority	0x6X	Compare	Check IPv6.
15 + T + S	3	Flow Label		Ignore	
18 + T + S	2	Payload Length		Ignore	
20 + T + S	1	Next Header	IPv6 Next Header Types or 0x3A	Compare	0x3A - ICMPv6 header type.
21 + T + S	1	Hop Limit	0x1	Ignore	
22 + T + S	16	Source IP Address		Ignore	
38 + T + S	16	Destination IP Address		Ignore	
54 + T+ S	N	Possible IPv6 Next Headers		Ignore	
ICMPv6 Header					
54 + T + S + N	1	Type	0x82, 0x83, ox86, 0x87,0x88, 0x89 or 0x8F	Compare	MLD or neighbor discovery types.
55 + T + S + N	1	Code	0x0	Ignore	
56 + T + S + N	2	Checksum		Check	
58 + T +S + N	F	Message Body		Ignore	

38.21.6.2.6 ARP Packets

ARP packets are used as part of the manageability filtering.

Figure 38-22 and Table 38-180 show/list the ARP packets format and the processing of these packets.

Figure 38-22.ARP Packet Format

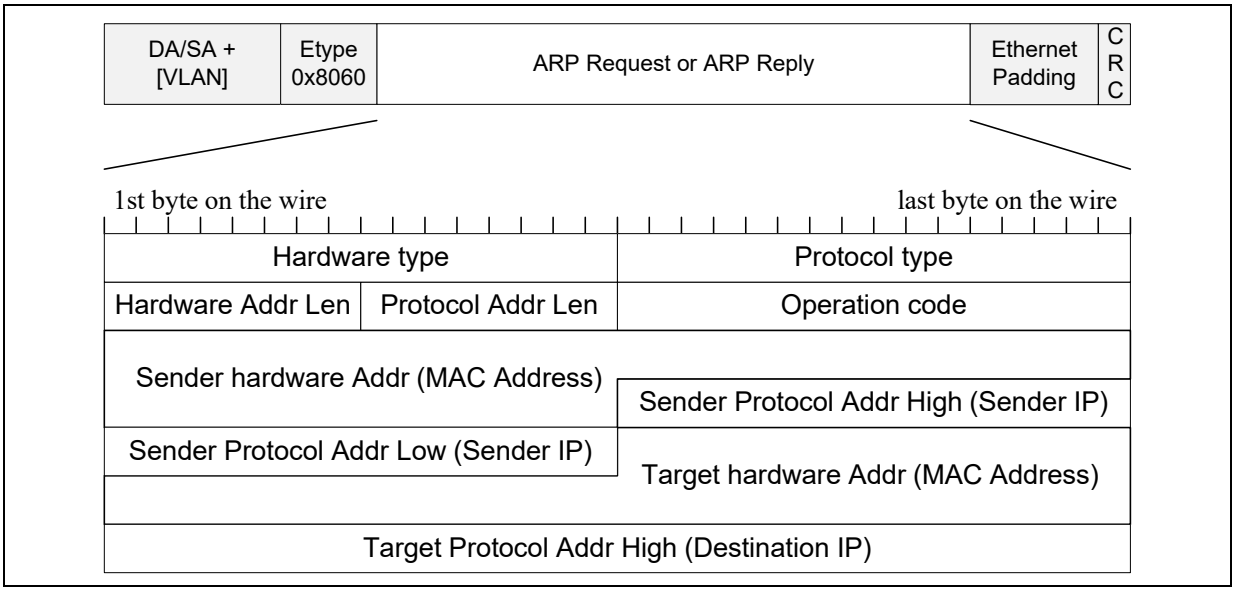


Table 38-180.ARP Packet Structure and Processing

Offset	# Of Bytes	Field	Value	Action	Comment
0	6	Destination Address		Compare	MAC header. Processed by main address filter.
6	6	Source Address		Skip	
12	T=(0/4/8)	Possible S-tag		Skip	
12 + T	S=(0/4)	Possible VLAN Tag		Compare	Processed by main address filter.
12+ T + S	2	Type	0x0806	Compare	ARP.
14 + T + S	2	Hardware Type	0x0001	Compare	Ethernet hardware type.
16 + T + S	2	Protocol Type	0x0800	Compare	IPv4 protocol.
18 + T + S	1	Hardware Size	0x06	Compare	MAC address size in bytes.
19 + T + S	1	Protocol Address Length	0x04	Compare	IPv4 address size in bytes.
20 + T + S	2	Operation	0x0001/ 0x0002	Compare	0x0001 = Request. 0x0002 = Response.
22 + T + S	6	Sender HW Address	-	Ignore	
28+ T + S	4	Sender IP Address	-	Ignore	
32 + T + S	6	Target HW Address	-	Ignore	
38 + T + S	4	Target IP Address	IP4AT	Compare	Use to decide of an IP match of ARP packets.

38.21.6.3 L4 Packet Formats

38.21.6.3.1 UDP Datagram

Figure 38-23.UDP Packet Format

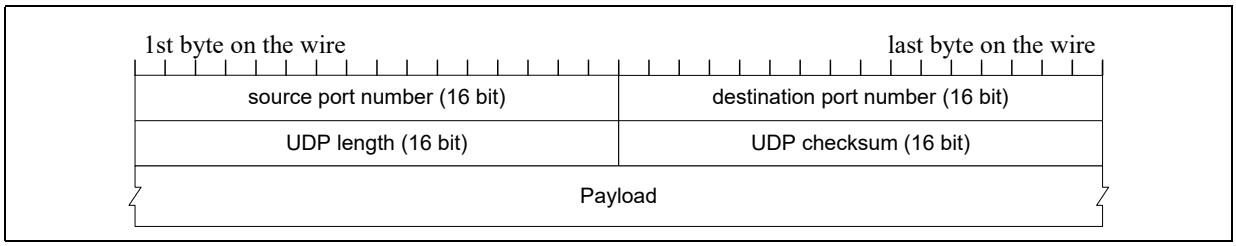


Figure 38-24.IPv4 Pseudo Header for TCP/UDP Checksum

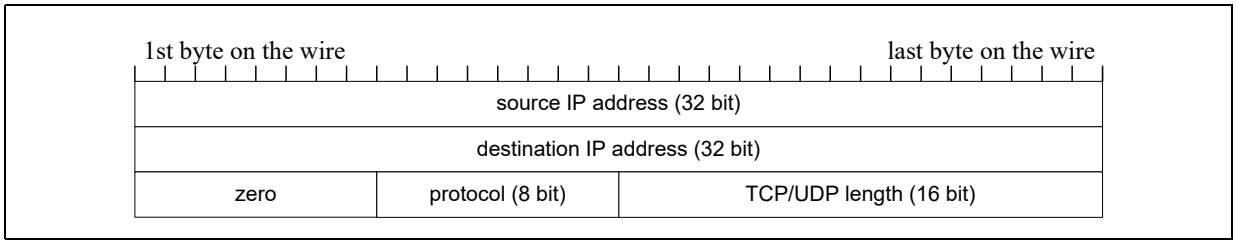
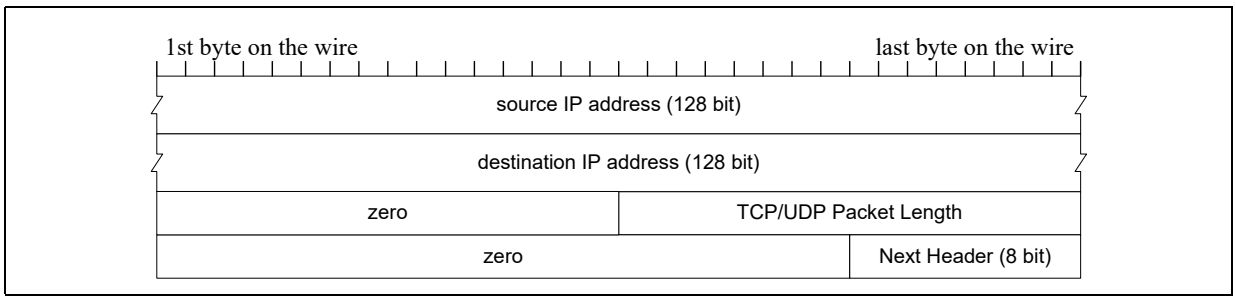


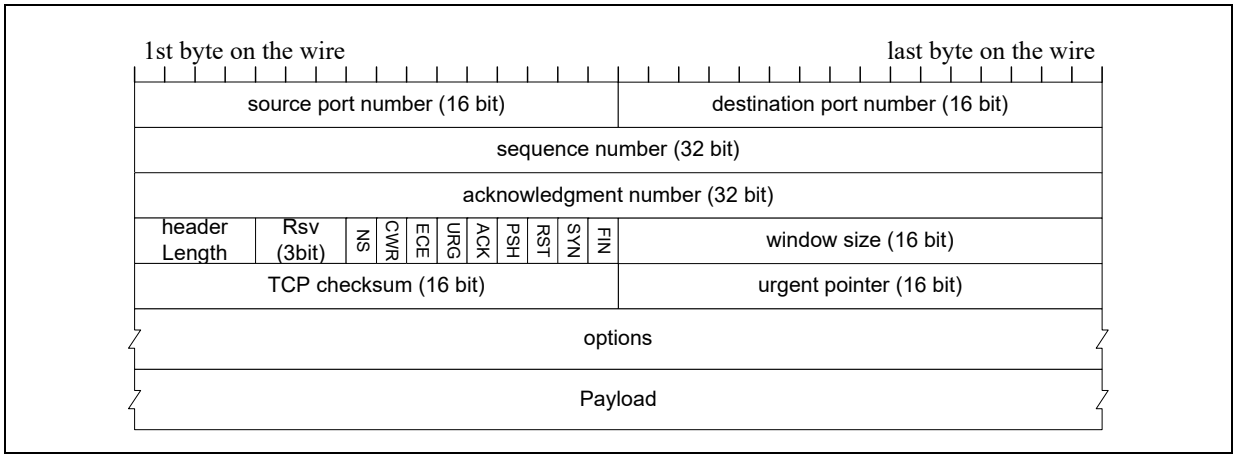
Figure 38-25.IPv6 Pseudo Header for TCP/UDP Checksum



- Source and Destination Port Number — 16-bit port numbers.
- UDP Length — The length of the entire UDP datagram, including both *Header* and *Data* fields defined in byte units.
- UDP Checksum — An optional one's complement of the one's complement sum of all 16-bit words in the header and payload and a pseudo header shown in Figure 38-24 and Figure 38-25. On the transmit flow, the operating system stack provides the pseudo header checksum in the UDP checksum field when requesting from the NIC to offload the checksum calculation.

38.21.6.3.2 TCP Datagram

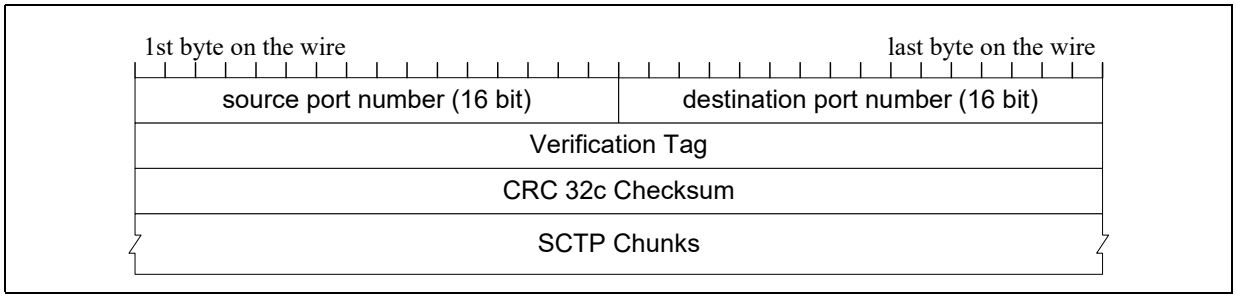
Figure 38-26.TCP Packet Format



- Source and Destination Port Number — 16-bit port numbers.
- Sequence Number — The sequence number of the first data octet in this segment (except when SYN is present). If SYN is present the sequence number is the Initial Sequence Number (ISN) and the first data octet is ISN+1.
- Acknowledge Sequence Number — If the ACK control bit is set this field contains the value of the next sequence number the sender of the segment is expecting to receive. Once a connection is established this is always sent.
- Window Size — The number of data octets beginning with the one indicated in the *Acknowledgment* field that the sender of this segment is willing to accept.
- TCP Flags (9 bits) — FIN, SYN, RST, PSH, ACK, URG, ECE, CWR, NS.
- Header Length (4bit) — The number of Dwords in the TCP header. This indicates where the data begins. The TCP header (including TCP options) is always an integral number of 32 bits long.
- Urgent Pointer — This field communicates the current value of the urgent pointer as a positive offset from the sequence number in this segment. The urgent pointer points to the sequence number of the octet following the urgent data. This field is only be interpreted in segments with the URG control bit set.
- TCP Checksum — The *Checksum* field a the 16-bit one's complement of the one's complement sum of all 16-bit words in the header and payload and the pseudo header shown in the Figure 38-24 and Figure 38-25. On the transmit flow, the operating system stack provides the pseudo header checksum in the TCP checksum field when requesting from the NIC to offload the checksum calculation.

38.21.6.3.3 SCTP Datagram

Figure 38-27.SCTP Packet Format

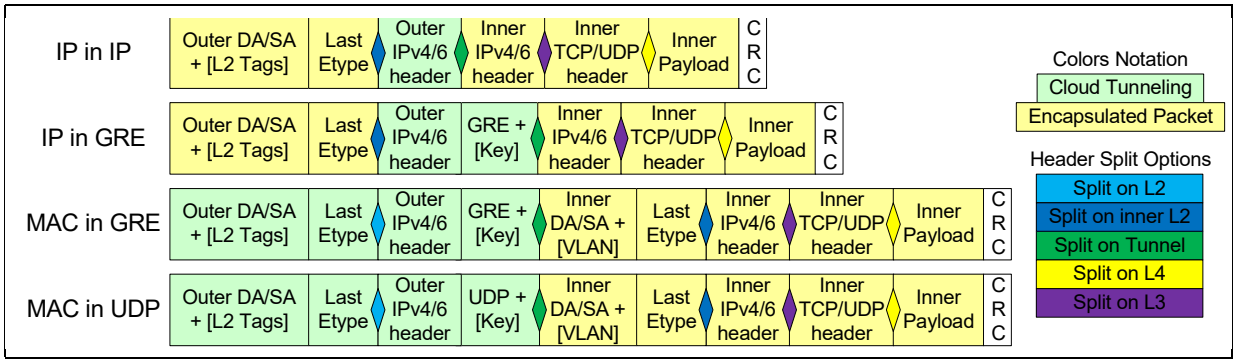


- Source and Destination Port Number — 16-bit port numbers.
- Verification Tag — 32-bit random value selected by each endpoint in an association during setup. It is used to discriminate between two successive associations as well as protection mechanism against blind attackers.
- CRC Integrity — CRC32c integrity checksum covering the entire SCTP packet (SCTP header and all chunks). The CRC32c is the same polynomial used for iSCSI as follows: 1 + x + x2 + x4 + x5 + x7 + x8 + x10 + x11 + x12 + x16 + x22 + x23 + x26 + x32. The CRC bytes are transmitted on the network in big endian ordering while the MS bytes is first on the wire.

38.21.6.4 Supported Tunneled Packet Formats

Supported tunneled packet formats are shown in Figure 38-28.

Figure 38-28.Tunneled Packet Formats



Processing of these packets is described in the LAN Engine chapter, Internal Switch chapter and Receive Classification Filters chapter.

38.21.6.4.1 IP and UDP Tunneling

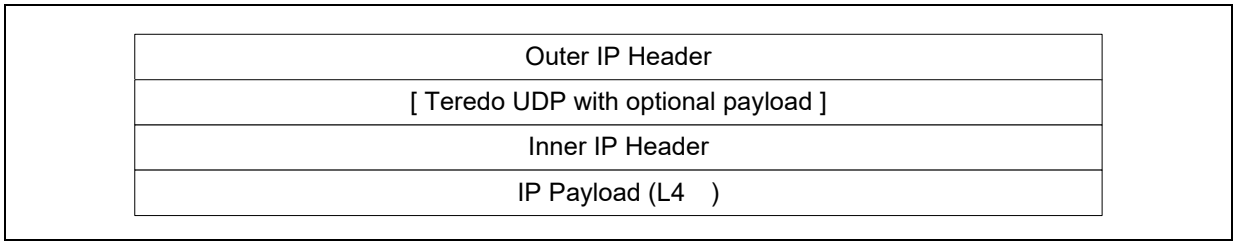
The 10 GbE controller supports the following tunneled packets:

- IPv4 — IPv4.
- IPv4 — IPv6.
- IPv6 — IPv4.
- IPv6 — IPv6.
- IP — UDP teredo - IP // The IP can be IPv4 or IPv6 as previously described.

The IP headers might have options (IPV4) or extended headers (IPv6) described in IPv6 Extension Headers.

UDP Teredo header is identified by its destination port number equals to one of 16 values. The 10 GbE controller does not provide checksum offload for the Teredo header with no reporting.

Figure 38-29.IP / Teredo Tunneling



38.21.6.4.2 GRE Header Version 0 (NVGRE)

1st byte on the wire								last byte on the wire
C	R	K	S	s	Recur	Flags	Version	Protocol Type
Checksum (Optional)								Offset (Optional)
Key (Optional) - Tenant Network ID (TNI)								Reserved
Sequence Number (Optional)								
Routing ::: (Optional)								

The GRE header is indicated by IP protocol equals to 47 (0x2F). The GRE headers supported by the 10 GbE controller can be 1, 2, 3 or 4 Dwords depending on the flags in the first byte.

- C (Checksum Present) — When set, it indicates that the *Checksum* field is present and contains valid information. If either the *Checksum Present* bit or the *Routing Present* bit are set, the *Checksum* and *Offset* fields are both present.
- R (Routing Present) — GRE header with routing header is not supported by the 10 GbE controller. If this flag is found active, the header is not recognized by the device.
- K (Key Present) — If set, then the *Key* field is present and contains valid information.
- S (Sequence Number) — If set, then the *Sequence Number* field is present and contains valid information. The 10 GbE controller ignores this flag other than an indication for the length of this header.
- s (Strict Source Route) — Intel recommends that this bit only be set if all of the routing information consists of strict source routes. The 10 GbE controller ignores this flag.
- Recur (Recursion Control) - 3 bits — Contains the number of additional encapsulations that are permitted. The 10 GbE controller supports only GRE header with no recursion headers (recursion control equals to zero).
- Version - 3 bits — GRE protocol version. Zero value identifies a GRE header version zero. The 10 GbE controller supports only zero value. Note that version 1 is used for PPP protocol.
- Protocol - 16 bits — Contains the protocol type of the payload packet. In general, the value is the Ethernet *Protocol Type* field for the packet. The following values are supported by the 10 GbE controller. Packet parsing that might have other protocol values is undefined.

  - — 0x0800 IPv4 header (indicates an IPv4 in GRE packet format)
  - — 0x86DD IPv6 header (indicates an IPv6 in GRE packet format)
  - — 0x6558 MAC header (indicates a MAC in GRE packet format)
  - — 0x894FNSH header (defined by the *NSH Protocol* field
- Checksum - 16 bits — Contains the IP (one's complement) checksum of the GRE header and the payload packet. This field is not processed by the 10 GbE controller.
- Offset - 16 bits — Indicates the byte offset from the start of the *Routing* field to the first byte of the active source route entry to be examined. This field is not processed by the 10 GbE controller.
- Key - 24 bits — Contains a number that was inserted by encapsulation. It might be used by the receiver to authenticate the source of the packet. The GRE key is used for VSI classification if enabled by the switch filters.
- Sequence Number - 32 bits — Contains a number that is inserted by encapsulation. It might be used by the receiver to establish the order in which packets have been transmitted from encapsulation to the receiver. This field is not processed by the 10 GbE controller.
- Routing - Variable length — This field is a list of SREs and is not supported by the 10 GbE controller.

38.21.6.4.3 UDP Plus its Optional Private Header

UDP Header	Optional UDP Private Header

- UDP Header — The UDP destination port number equals to one of 16 Teredo port numbers (also used for MAC in UDP tunneling).
- Private Header — The private UDP header includes any optional private content that could include a networking key. The length of the private header and the optional networking key are recognized by the device according to the UDP port number as programmed by the Add Teredo Port admin command.

VXLAN Header in UDP Format

VXLAN draft provides the structure of the optional private header previously described and used for VXLAN encapsulation as shown in Figure 38-30 and listed in Table 38-181.

Figure 38-30.VXLAN Header Structure

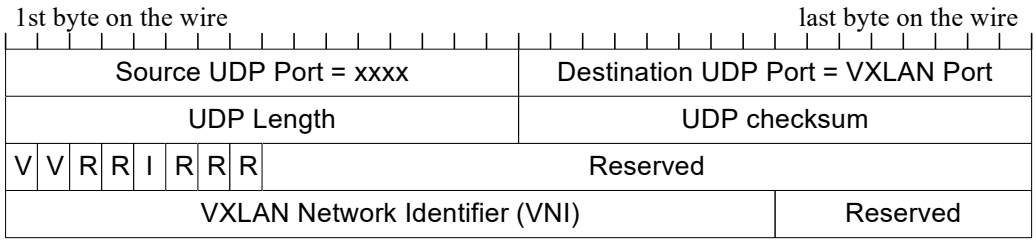


Table 38-181.VXLAN Header Structure

Offset (From The UDP Header)	Size (Byte)	Field	Value	Action	Comment
0	2	Source UDP Port	XXXX	Ignore	Source UDP port number could be any value.
2	2	Destination UDP Port	VXLAN Port	Compare	The reserved VXLAN port number is programmed by the Add Tunneling UDP admin command.
4	2	UDP Length	XXXX	Ignore	Length of the entire datagram including the UDP header.
6	2	UDP Checksum	XXXX	Ignore	Provide an indication if the checksum equals zero (such as no checksum).
8	1	Flags	0x08	Ignore	VV = version (expect zero). I = VNI valid indication.
9	3	Reserved	XXXX	Ignore	Reserved.
12	3	VXLAN Network Identifier	XXXX	Compare	Unique value per tenant.
15	1	Reserved	XX	Ignore	Reserved.

Geneve Header in UDP Format

Geneve draft (Geneve: Generic Network Virtualization Encapsulation draft-gross-geneve-00) provides the structure of the optional private header previously described used for Geneve encapsulation as listed in Table 38-182 and Table 38-183.

Table 38-182.Geneve Header Structure

1st byte on the wire					last byte on the wire
Source UDP Port = xxxx					Destination UDP Port = 0x6081
UDP Length					UDP Checksum
V	Opt Len	O	C	Reserved	Next Protocol
Virtual Network Identifier (VNI)					Reserved
Variable Length Options (defined by Opt Len)					

Table 38-183.Geneve Header Structure (Sheet 1 of 2)

Offset (From The UDP Header)	Size (Byte)	Field	Value	Action	Comment
0	2	Source UDP Port	XXXX	Ignore	Source UDP port number could be any value.
2	2	Destination UDP Port	6081	Compare	The reserved Geneve port number is programmed by the Add Tunneling UDP admin command.
4	2	UDP Length	XXXX	Ignore	Length of the entire datagram including the UDP header.
6	2	UDP checksum	XXXX	Check	Checksum integrity indication.
8.7 - 8.6	2 bits	V V (Version)	00b	Check[1]	
8.5 - 8.0	6 bit	Opt Len (Option Length)	Variable	Check[1]	Defines the length of the options fields in 4-byte units.
9.7	1 bit	O (OAM frame)	0b	Check	Rout the packet to queue zero of the VSI.
9.6	1 bit	C (Critical Options Present)	X	Ignore	Expected to be processed by the software stack.
9.5 - 9.0	6 bit	Reserved	X	Ignore	

Table 38-183.Geneve Header Structure (Sheet 2 of 2)

Offset (From The UDP Header)	Size (Byte)	Field	Value	Action	Comment
10	2	Next Protocol		Check	Next protocol that follows the Geneve header. The 10 GbE controller supports the following values: 0x6558 MAC header (MAC in Geneve). 0x0800 IPv4 header (IPv4 in Geneve). 0x86DD IPv6 header (IPv6 in Geneve). 0x894F NSH header (NSH in Geneve). If other values are found, the packet is identified as Geneve or payload.
12	3	VNI (Network Identifier)	Variable	Compare	Unique value per tenant.
15	1	Reserved	X	Ignore	
16	4 x Opt Len	Variable Length Options	XXXX	Ignore	The length of the *Options* field is defined by the Opt Len field in this header.

Notes:

1. If the total header length is larger than 256 or the version <> 0, the packet is handled as a simple L2 packet type.

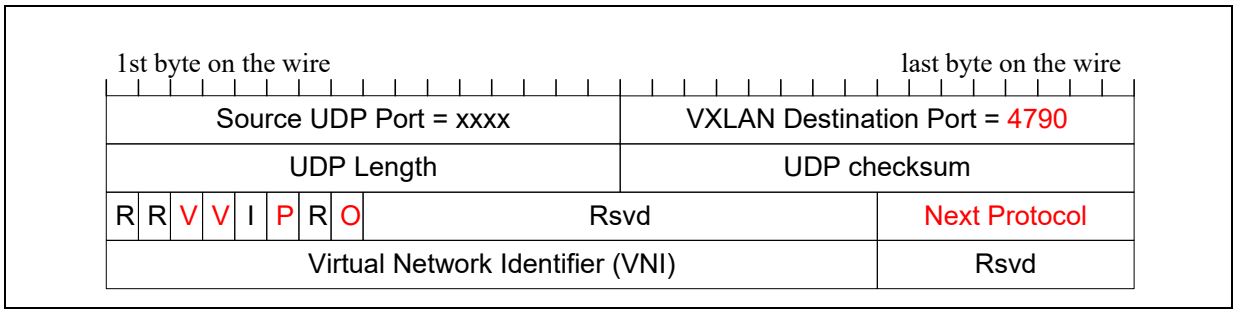
38.21.6.4.4 VXLAN-GPE Header in UDP Format

VXLAN-GPE draft provides the structure of the optional UDP private header previously described used for VXLAN-GPE encapsulation as listed in Table 38-184 and shown in Figure 38-31. The highlighted fields in red color are those fields that are changed in VXLAN-GPE compared to the original VXLAN header.

Table 38-184.VXLAN-GPE Header Structure (Additive Fields on Top of Original VXLAN)

Byte Offset (from the UDP Header)	Size	Field	Value	Action	Comment
2	2 byte	Destination UDP Port	4790	Compare	The reserved VXLAN-GPE port number is programmed by the Add Tunneling UDP Admin command with Tunneling UDP Protocol Type equals to VXLAN-GPE.
8.0	1 bit	O (OAM bit)	0 / 1	Compare	Packets with active OAM flag are identified as VXLAN-GPE-OAM PCTYPE and routed to Rx queue zero of the VSI.
8.2	1 bit	P (Next Protocol Field Valid)	0 / 1	Ignore	The next *Protocol* field is valid only when the *P* bit is set. The 10 GbE controller ignores this flag based on the assumption that if the *P* is cleared, then the *Next Protocol* field equals to zero.
8.3	1 bit	I (VNI Tag Present)	1	Ignore	VNI valid indication must be set in the header.
8.5 - 8.4	2 bits	Version Field	00	Compare	Parsing of packets with non-zero *Version* field is aborted. It is indicated by Abort PTYPE in the Rx descriptor and the switching decision is made by the MAC/VLAN headers.
11	1 byte	Next Protocol	XXXX	Compare	Next protocol that follows the VXLAN-GPE header. Product Name supports the following values: 0x0 or 0x3 = MAC in VXLAN-GPE. 0x1 = IPv4 in VXLAN-GPE. 0x2 = IPv6 in VXLAN-GPE. 0x4 = NSH in VXLAN-GPE. If other values are found, the packet is identified as VXLAN-GPE or payload.
12 - 14	3 bytes	Virtual Network Identifier	Variable	Compare	Unique value per tenant.

Figure 38-31.VXLAN-GPE Header Structure



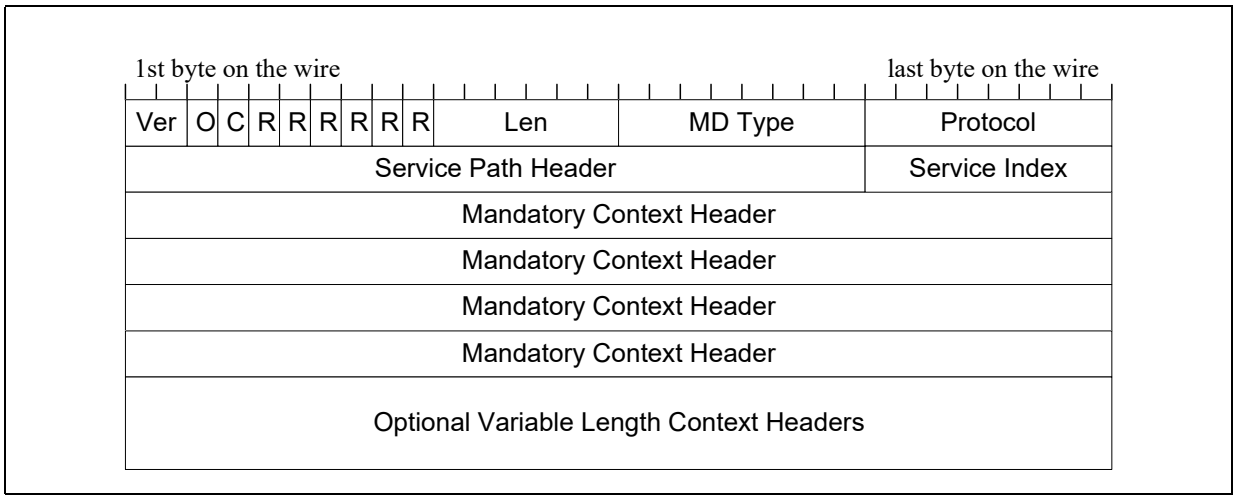
38.21.6.5 NSH Header

NSH is one of the optional Next Protocol in VXLAN-GPE or in GRE or as an L2 protocol (EType = 0x894F). The structure of the NSH header is shown in Table 38-184 and Figure 38-31.

Table 38-185.NSH Header Structure

Byte Offset	Size [byte]	Field	Value	Action	Comment
0.4	1 bit	C (Critical data flag)	0 / 1	Ignore	
0.5	1 bit	O (OAM Frame)	0 / 1	Compare	Packets with active OAM flag are identified as NSH-OAM PCTYPE and routed to Rx queue zero of the VSI.
0.7 - 0.6	2 bits	VV (NSH Version)	00b	Compare	Parsing of packets with non-zero *Version* field is aborted. It is indicated by abort PTYPE in the Rx descriptor and the switching decision is made by the MAC/VLAN headers.
1.5 - 1.0	6 bits	Length	XXXX	Process	Defines the length of the NSH header in 4-byte units.
2	1	MD Type (NSH Meta Data Type)	XXXX	Ignore	
3	1	Protocol	XXXX	Process	Next Protocol can be one of the following: 0x3 = MAC in NSH. 0x1 = IPv4 in NSH. 0x2 =IPv6 in NSH. If other values are found, the packet is identified as NSH or payload.
4 - 7	4	Service Path Header and Index	XXXX	Ignore	
8 - 23	16	Mandatory Context Header	XXXX	Ignore	
24	Variable	Optional Context Headers	XXXX	Ignore	Optional Variable Length Context Headers.

Figure 38-32.NSH Header Structure



38.21.6.5.1 NSH Header Encapsulation Options

The 10 GbE controller supports the following NSH encapsulation options as shown in Figure 38-28:

Native NSH over L2 (L2 EtherType = 0x894F); NSH over GRE and NSH over VXLAN-GPE

Note: NSH support should be considered only as NSH ready. It is enabled but not validated.

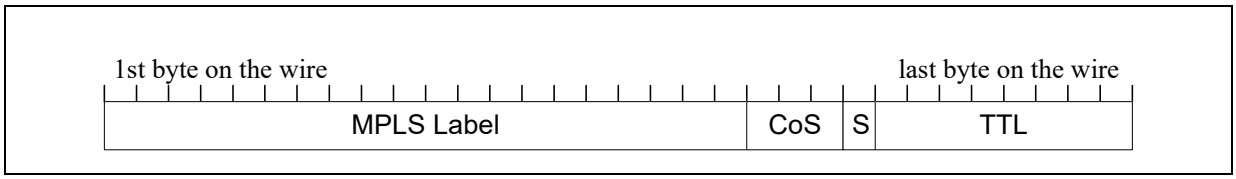
38.21.6.6 Unicast MPLS Header

Multi Protocol Layer Switching (MPLS) is a 2.5 protocol layer tag used to enable IP and non-IP packet routing. The 10 GbE controller ignores the content of the MPLS header and skips it to the L3 protocol layer. The 10 GbE controller supports stacked MPLS headers up to four headers. It identifies the next protocol by the value of its first byte. For IP headers, the first byte indicate if it is an IPv4 or IPv6. Any other value is interpreted as L2 payload. The structure of the MPLS header is listed in Table 38-186 and shown in Figure 38-33.

Table 38-186.MPLS Header Structure

Offset (relative to MPLS header)	Size	Field	Value	Action	Comment
-2	16	EtherType	0x8847	Compare	
0	20 bits	MPLS Label	XXXX	Ignore	
2.4	4 bits	CoS (Class of Service)	XXXX	Ignore	
2.7	1 bit	S (Stack)	0 / 1	Process	0 - Additional MPLS headers to follow 1 - Bottom of the Stack (last header)
3	8 bit	TTL (Time to Live)	XXXX	Ignore	

Figure 38-33.MPLS Header Structure



The following assumptions are taken while handling MPLS packets:

The 10 GbE controller only parses MPLS headers to be able to skip them. It does not use the MPLS header content for any filtering and does not provide any extract or insert offloads.

MPLS is not supported for RDMA packets.

The 10 GbE controller does not support line rate parsing for MPLS packets.

38.21.7 Internal Switch

38.21.7.1 Switch Concept

The 10 GbE controller includes a number of switch elements. Part of these elements are generic L2 switching elements and part are dedicated elements that can implement a specific functionality.

These elements are compatible to the IEEE P802.1Qbg specification. Section 38.21.7.2 describes the possible usage models of the switch as defined in these specifications. Section 38.21.7.3 describes the management protocol used to configure the switch.

The different elements are described in Section 38.21.7.4. Each switch can be connected to different ingress and egress ports in different configurations. The algorithm that defines which switch routes which packet is described in Section 38.21.7.13.

The switches can have VSIs connected through the PCIe interface, a physical port connected via the Ethernet interface and management ports either internal (EMP) or connected to a PCIe function.

VSIs contain a set of queues or flows and are characterized by an L2 identifier. After the packet is forwarded to a specific ingress port, an internal destination inside the port is selected such as LAN traffic (Section 38.21.1) and iWARP (Section 38.36.1).

In addition to forwarding services, the switch also supports some additional services as described in Section 38.21.7.10.

The switch is not a learning switch and it is managed by the host, The programming interface exposed to the operating system is that of a managed switch. Each switch element can be configured either via a driver running on a PF or via the EMP. When a driver programs the switch it uses admin commands processed by the EMP as described in Section 38.21.7.14.

The switch is built from a set of tables that can be allocated to different filtering resources and a set of other fixed resources (VSIs, VEBs, mirroring rules, etc.). The current resource allocation and the way to allocate them to PFs are described in Section 38.21.7.16.1.

38.21.7.2 Switches Configurations

The 10 GbE controller embedded switch can be used to offload the data plane functions of the virtual switching capability supported by many VMMs (bridging and switching are used synonymously in this document).

This section describes the logical models that can be represented by the switching elements. The actual switching elements definition can be found in Section 38.21.7.4.

The 10 GbE controller switch can be configured to operate in different modes as follows:

- Internal switching configurations — Virtual embedded bridging:
    - Software based switching in VMM
    - Hardware based switching for direct connected VMs
    - Local switching between VMs
- External switching configurations:
    - Virtual port aggregator where VM-to-VM switching is performed by an adjacent bridge.
    - Port virtualizer configuration where the VM-to-VM switching is performed by a controlling bridge that could be in the network edge or aggregation layers.

Note: A device might work either in port aggregator or in port virtualizer mode. The mode of operation should be the same for all the ports.

- Combination of internal and external switching configurations:
    - — Internal switching modes, software switching model or direct connect model, can coexist with one of the external switching modes, port aggregator or port virtualizer.

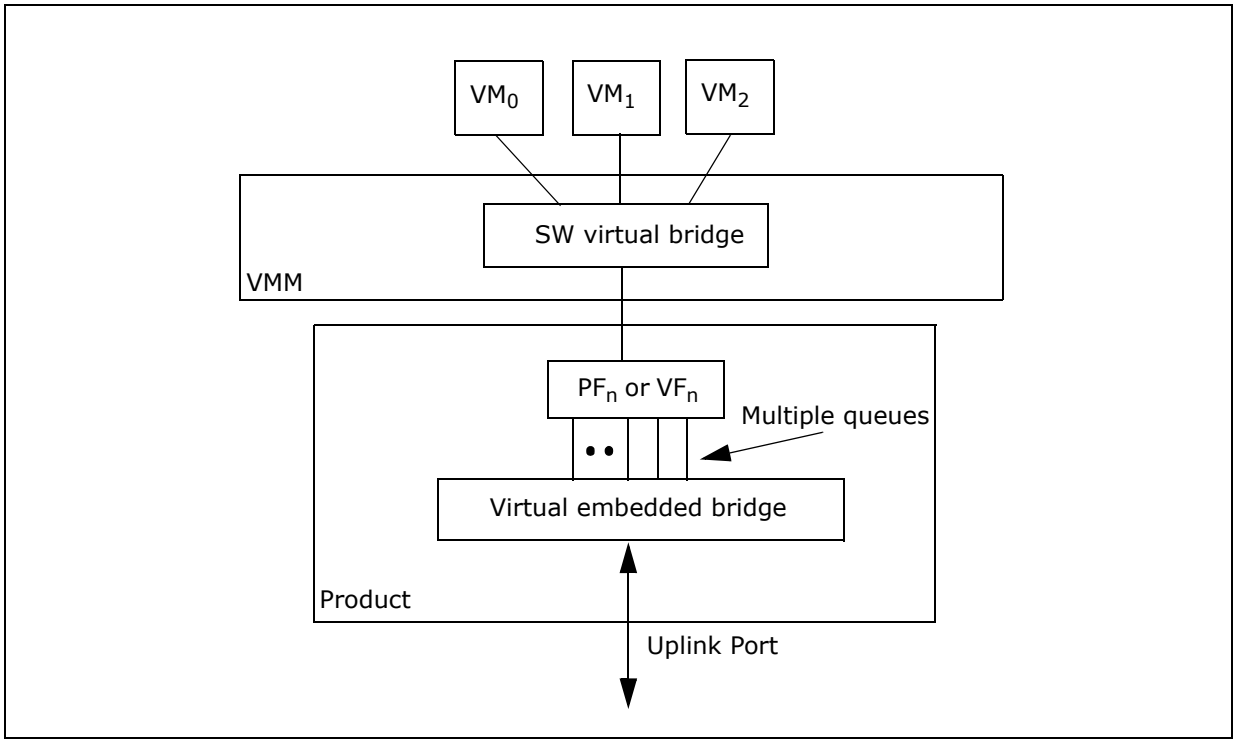
38.21.7.2.1 Internal Switch Configurations

In the internal switching mode, the 10 GbE controller switching element can be configured to offload data plane functions of the VMM-based software switches or can handle data plane switching functions for direct connected VMs (through SR-IOV or multiple PCI functions).

Hardware Offload of Software Virtual Embedded Bridges (VEBs)

The following figure shows the hardware offload of software-based switches.

Figure 38-34.Internal Switching Configuration — Software Virtual Embedded Bridging



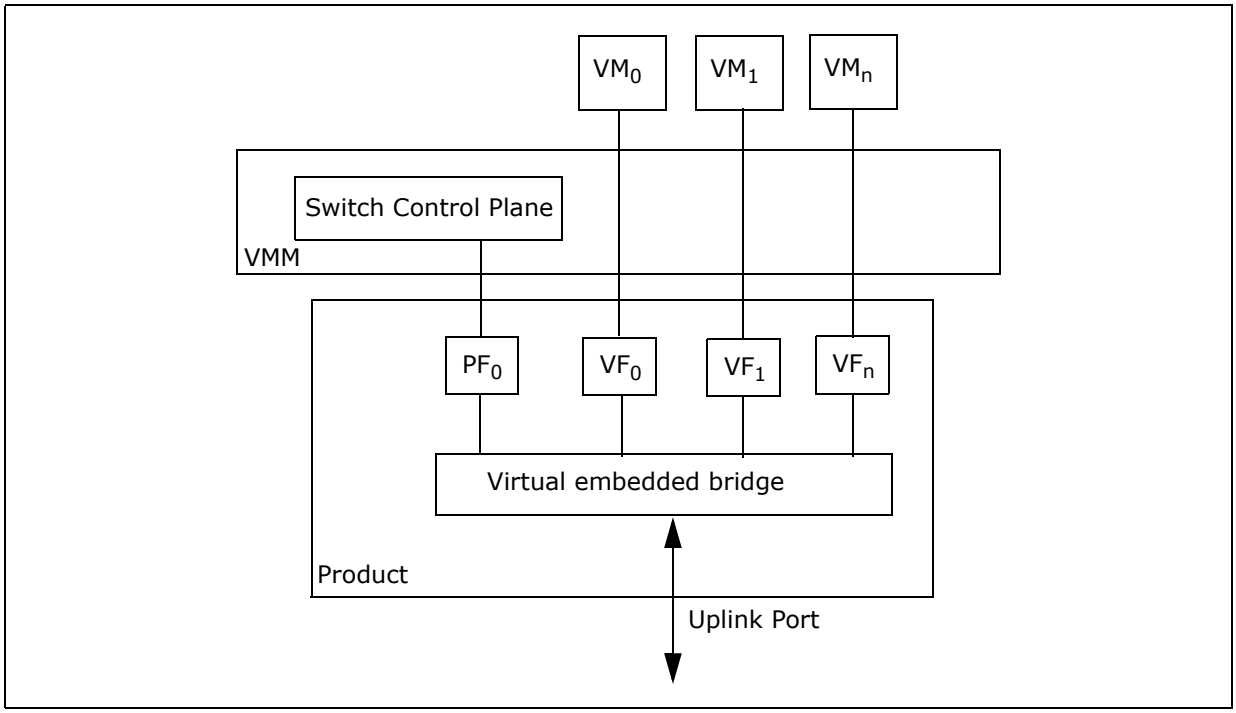
When the 10 GbE controller switching element is configured to operate in this mode, data plane functions such as the filtering and forwarding functions are offloaded to hardware. The switching element can also perform hardware replication of multicast and broadcast frames. In addition, the switch can perform VLAN offload functions and security functions such as MAC address spoofing. In this mode, multiple queues are assigned to the PCI function (PF or VF in case of SR-IOV). Each set of queues represents a VSI in the switching element. The uplink of the switch is connected to one of the external Ethernet MAC ports.

The entire control plane functionality is handled by the software switch in the VMM via a PF. The switch hardware is programmed by a set of switch APIs. The switch control plane software manages the switching elements forwarding database, VLAN membership, bandwidth assignment or other virtual port characteristics. Since the forwarding path is still through the software intermediary layer in the VMM, it is possible for the software switch to examine the packets or perform additional functions on the frames. For example, handling new protocol headers, collecting additional statistics etc.

VEB for Direct Connected VMs

The following figure shows the hardware switching of direct connected VMs.

Figure 38-35.Internal Switching Configuration — Hardware VEB



When the 10 GbE controller switching element is configured to operate in this mode, each of the VSIs are directly connected to PCIe functions (PFs or VFs) that are assigned to the VMs. The software layer is eliminated in the forwarding path hence the entire data plane functionality is handled by the hardware embedded switch. All the virtual switching functions such as VLAN membership, filtering and forwarding, bandwidth management, statistics etc., are handled in hardware.

The entire control plane functionality is handled by the switch management agent in the VMM. Control packets are forwarded to the control plane agent through a management port. The switch hardware is programmed by a set of switch APIs. The switch management software manages the switching elements forwarding database, VLAN membership, bandwidth assignment or other VSI characteristics. The switch management can monitor the traffic by accessing the statistics counters or can also monitor data traffic to VM by enabling mirroring functionality in the embedded switch.

Multiple virtual embedded switching elements can be instantiated in the 10 GbE controller. There can be only one virtual switching element per uplink port. If there are multiple switching elements on a single uplink port, then a multi-channel capable S-component needs to be configured on the uplink. See Service-VLAN Component (S-comp) for more detail. However, a virtual switching element can be configured without an uplink port for local switching between VMs. This is used for configuring internal networks.

38.21.7.2.2 External Switch Configurations

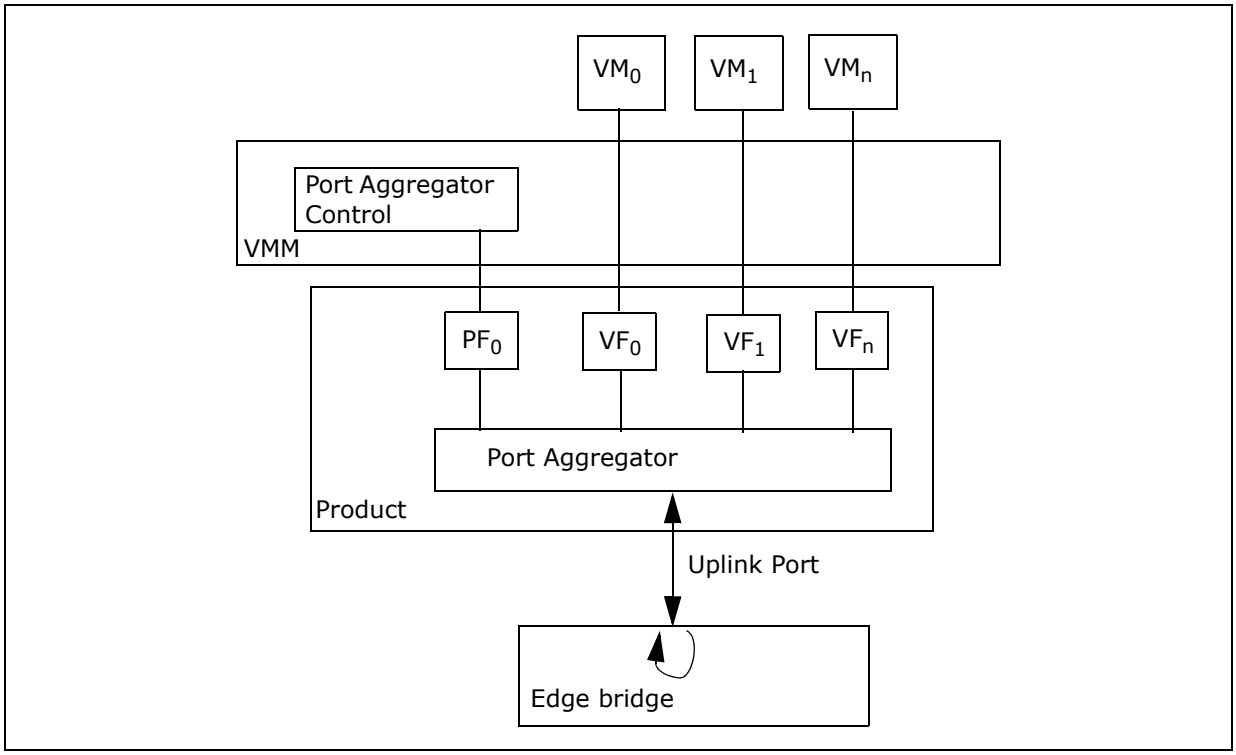
The 10 GbE controller supports external switching modes. In the external switching modes the virtual switching functionality including VM-to-VM, and VM to uplink forwarding is handled by external switches. This enables a single network management domain to manage both enterprise switches and virtual switching. The entire data plane functionality and the control plane functionality is handled by the external switch.

Two protocols are being defined by IEEE to enable virtual bridging in external switches, IEEE P802.1Qbg and IEEE P802.1BR. The 10 GbE controller supports both models. Note that the virtual port aggregator model is very similar to the embedded virtual switching model.

Virtual port Aggregator Model

The following figure shows a port aggregator configuration.

Figure 38-36.External Switching Configuration — Virtual Port Aggregator



In a simple port aggregator mode, there is no change to the Ethernet frame format. During the ingress direction, the operation is similar to VEB, the packets are forwarded to the VMs by querying into the forwarding database in the embedded switch. In the egress direction all traffic including the VM-to-VM traffic is forwarded the external switch. The external switch performs lookup on the packets and turns-around the packet back to the end node (called reflexive relay service or hairpin forwarding) if the packet has to go to another VM residing in the same end station. Since all packets flow through the external switch, all advanced features of the external switch (such as advanced ACLs and flow monitoring features) are available for VM-to-VM traffic. However, the bandwidth of the uplink is shared between the VM-to-VM traffic and the uplink traffic; also there is additional latency for externally switched VM-to-VM traffic.

In port aggregator mode, hardware performs broadcast and multicast replication functions. During multicast and broadcast, the packet is forwarded by the external switch once to the end station and the port aggregator in the 10 GbE controller replicates the packet multiple times to all the multicast member VMs attached to the port aggregator.

The VSIs of a port aggregator can be directly connected to the VMs through PCIe functions or VFs in SR-IOV mode. A cascaded port aggregator configuration is also allowed where a software based port aggregator in VMM can be connected through one of the virtual ports. In this case, the port is designated as a cascaded port (See Section 38.21.7.2.4).

The discovery and configuration of port aggregator mode is handled by a software port aggregator control agent residing in the VMM. The hardware switch element is managed through a set of APIs accessible to the port aggregator control agent.

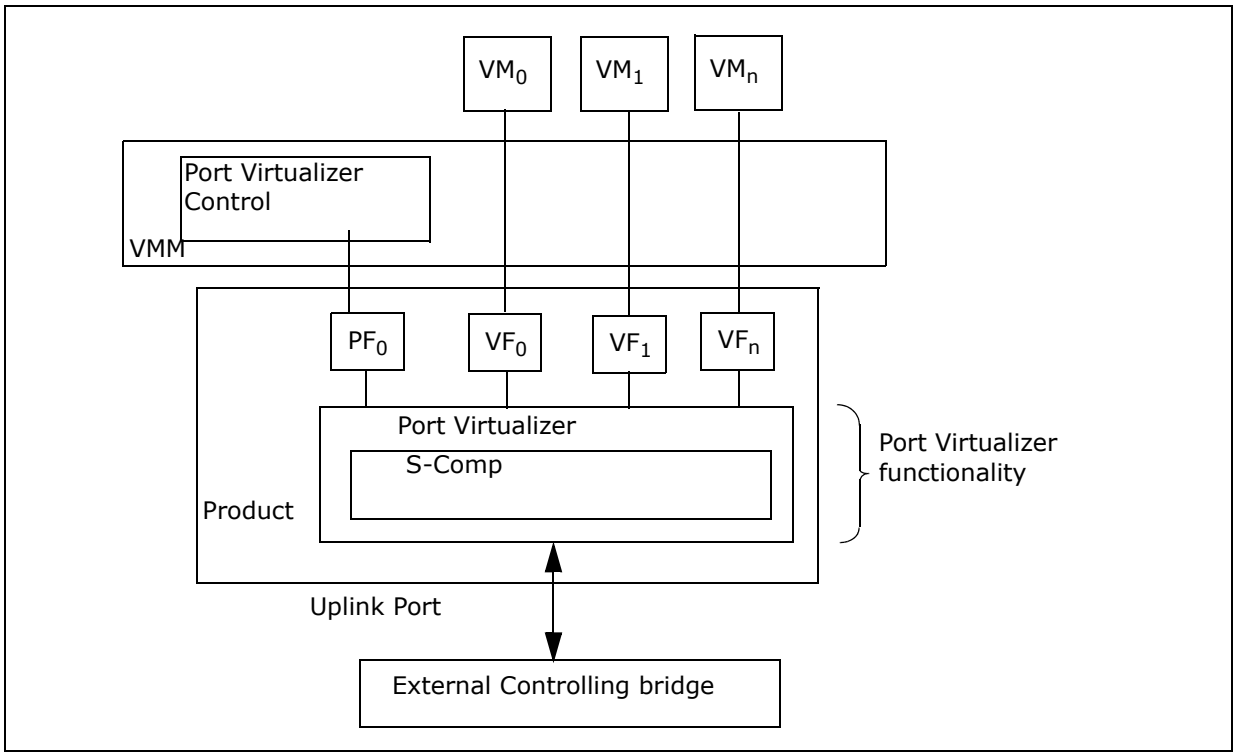
Port Virtualizer Model

There are two ways to divide the link to channels:

- A port mapping S-VLAN component (S-comp) defined in the IEEE 802.1Qbg specification. The S-comp uses S-tags (EtherType = 0x88A8) to define S-channels that do not support replication. See S-comp Model and Service-VLAN Component (S-comp) for more detail.
- A port mapping based on MAC address. This mode is used when a function is dedicated to storage function and uses a small and exclusive set of MAC addresses. This mode supports two channels only. A storage channel and a default channel receive all the other traffic. See Section 38.21.7.3.

The following figure shows the hardware switch element in the 10 GbE controller configured to operate in S-comp configuration.

Figure 38-37.External Switching Configuration — Port Virtualizer



S-comp Model

In an S-comp port virtualizer configuration, the controlling bridge's ports are logically extended to the virtual stations (or VMs). A physical port in the controlling bridge is configured to have multiple logical channels (or logical ports), typically one per VM. Correspondingly the physical port in the end node is also configured to have same number of logical channels (or logical ports) that are associated with VMs. The packets need to explicitly carry a tag to identify the logical channels (or logical ports). An S-component (service VLAN component) is configured to provide this multi-channel capability. The S-tag indicates a logical channel or logical port on the controlling bridge. In this mode of operation, each VSI in the hardware switching element is associated with a logical channel in the S-component.

Connection to the Host

The VSIs of a port virtualizer can be directly connected to the VMs through PCIe functions or VFs in SR-IOV mode. A cascaded port virtualizer configuration is also allowed where a software-based port virtualizer in VMM can be connected through one of the virtual ports. In this case, the port is designated as a cascaded port (See Section 38.21.7.2.4). The discovery and configuration of port virtualizer mode is handled by a software port virtualizer control agent residing in the VMM. The hardware switch element is managed through a set of APIs accessible to the port virtualizer control agent in VMM or IOVM.

38.21.7.2.3 Coexistence of Internal/External Switch Configurations

Both internal and external switching configurations can coexist within an end station accessible through the same uplink port. Legacy VMs or applications that need higher performance for VM-to-VM traffic requires VEB functionality to perform internal switching. In such usage models, the VEB coexists with either a port aggregator or port virtualizer in the same end station.

The following figure shows a combination of VEBs with a port aggregator.

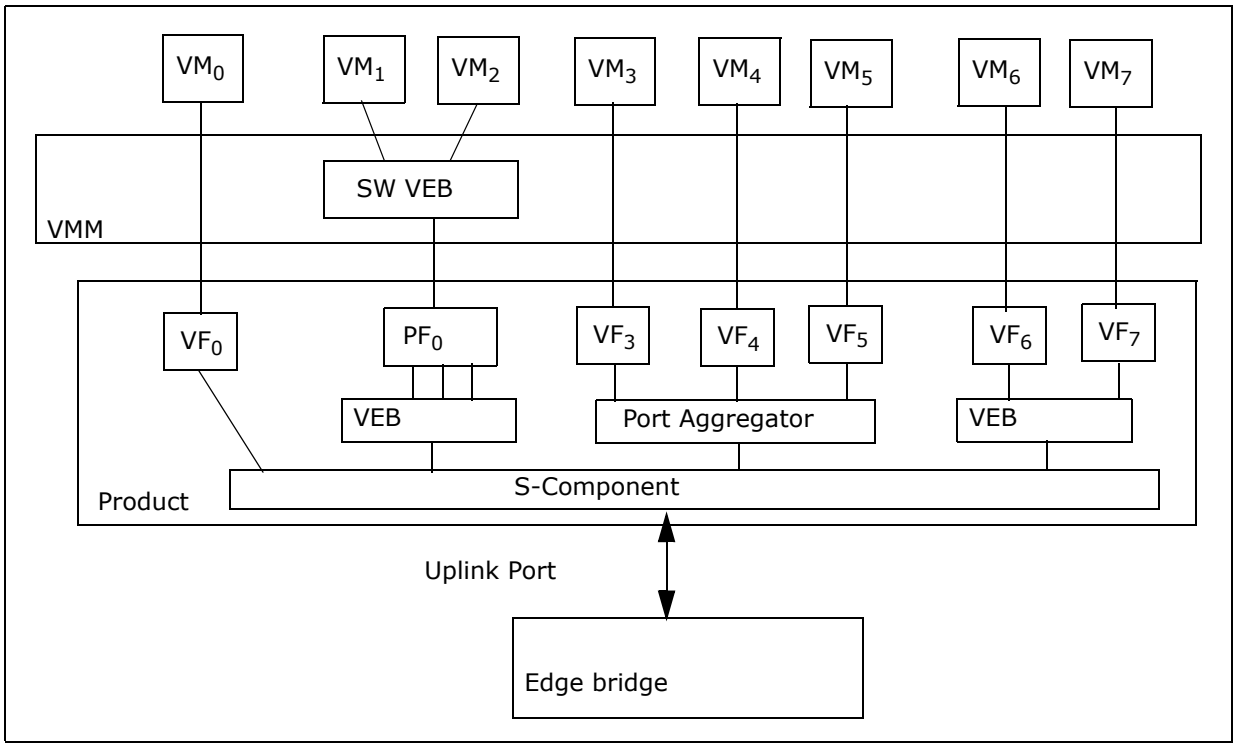
In this configuration, multiple switching elements are instantiated in the 10 GbE controller on the same uplink port. Each instance is accessible through a multi-channel S-component. The S-component is used to mux/demux multiple logical channels (S-channels) from the same physical port. The S-tag identifies the logical channels (S-channels) or logical ports that are extended from the adjacent bridge.

In this example configuration, the legacy VMs are connected through a software-based VEB connected through one of the S-channel ports. A second VM (that hosts a virtual appliance) is directly connected to the external switch through the second S-channel port. In addition, a port aggregator is configured through the third S-channel port that provides external switching functionality to VMs. Certain higher performance applications that require heavy VM-to-VM traffic (such as co-located front end and mid tier application servers) are configured through directly connected VFs to an embedded bridge. This embedded bridge is accessible through fourth S-channel port.

Discovery and configuration of individual switching components is performed through appropriate control agents in the VMM or IOVM.

The configuration of the switch is done using the CDCP protocol as defined in 802.1Qbg specification.

Figure 38-38.Cascaded VEB



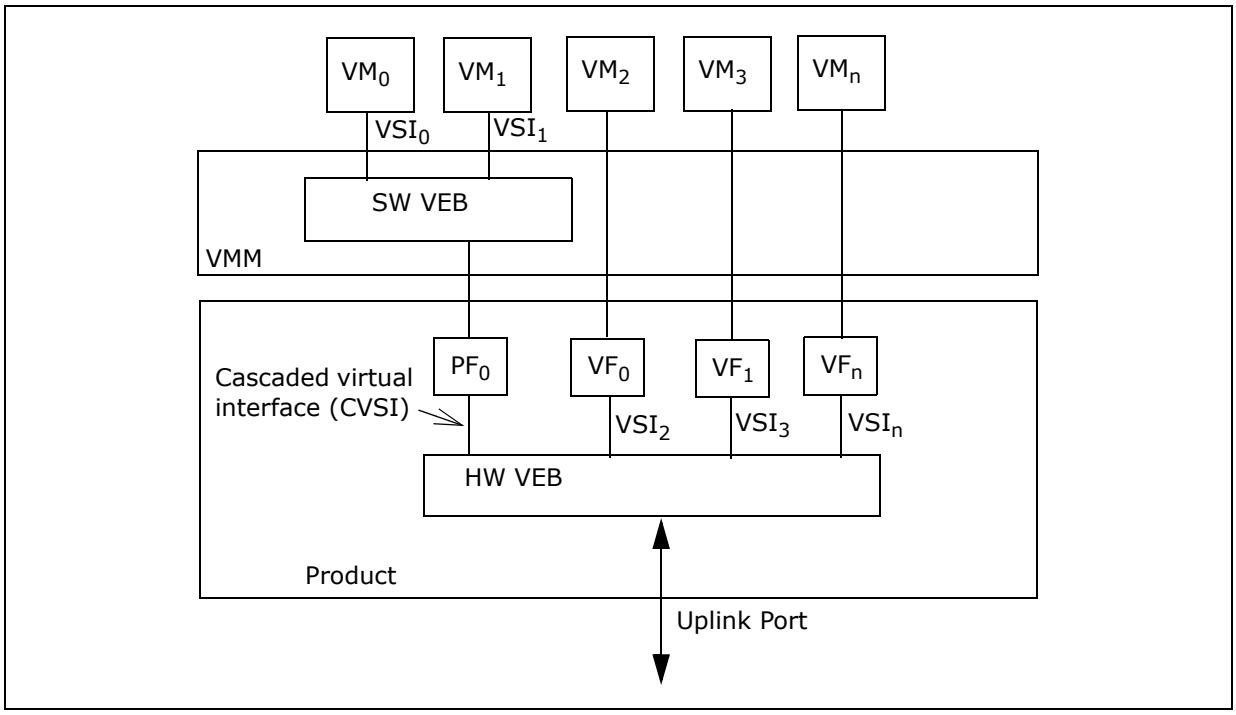
38.21.7.2.4 Cascading of Software VEBs, Port Aggregators and Port Virtualizers

Software-based VEBs, port aggregators and port virtualizers could be cascaded over hardware-based VEBs, PVs and PAs in the NIC. This section provides illustrations of typical cascaded configurations allowed by the 10 GbE controller.

Cascaded Hardware and Software VEBs

The following figure shows a software VEB cascaded over a hardware VEB.

Figure 38-39.Cascaded



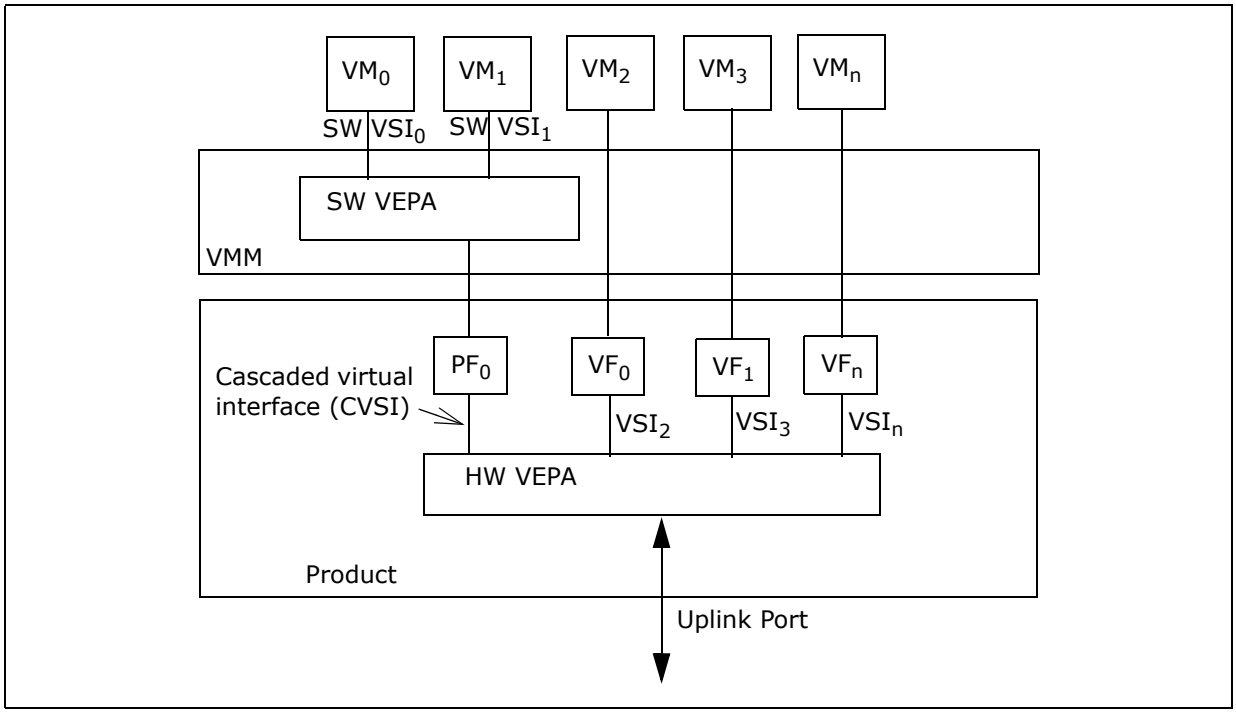
In this configuration, a hardware VEB is instantiated in the 10 GbE controller controller. The VEB is configured with multiple VSIs. A VSI refers to a connection between the VEB and a virtual end station and its associated resources such as MAC address, VLAN, BW/QoS attributes etc. Typically a VSI is configured with a single MAC address and VLAN. A software VEB is configured in the VMM and the connection between the software VEB and a port in the hardware VEB is referred to as a cascaded VSI (or simply a downlink port). The cascaded interface is a special VSI that is used as a downlink to a software VEB. This type of port is referred to as a trunk port in typical Ethernet switches. The downlink ports are typically configured with multiple MAC addresses and multiple VLANs serviced by the software VEB. Security features such as MAC anti spoofing and VLAN ingress checks should not be configured on the downlink ports since those functions are performed at the ingress VSIs of the software VEB.

The 10 GbE controller can be configured to perform VLAN offload functions on the downlink VSIs. In this case, the VLAN information is carried in the transmit descriptor and the 10 GbE controller inserts the VLAN in the transmit direction. The 10 GbE controller can strip the VLAN in the receive direction and store the VLAN tag information in the receive descriptor.

Cascaded Hardware and Software Port Aggregators

The following figure shows a software VEPA cascaded over a hardware VEPA.

Figure 38-40.Cascaded Virtual Ethernet Port Aggregator (VEPA)



In this configuration, a hardware VEPA is instantiated in the 10 GbE controller controller. The hardware VEPA is configured with multiple VSIs. A software VEPA is configured in the VMM and the connection between the software VEPA and a port in the hardware VEPA is referred to as a cascaded VSI (or simply a downlink port). The cascaded interface in this case is a special VSI that is used as a downlink to a software VEPA. The only cascaded configuration allowed over a hardware VEPA is a software VEPA. A software VEB cannot be cascaded over a hardware VEPA because this configuration is not allowed by the IEEE P802.3bg specification.

The downlink ports in a VEPA are similar to a VEB trunk port (see Cascaded Hardware and Software VEBs) except that multicast source address pruning functionality specified by IEEE P802.3bg is not configured on a downlink port to a software VEPA; this functionality is performed at the egress of the software VEPA VSIs connected to the VMs.

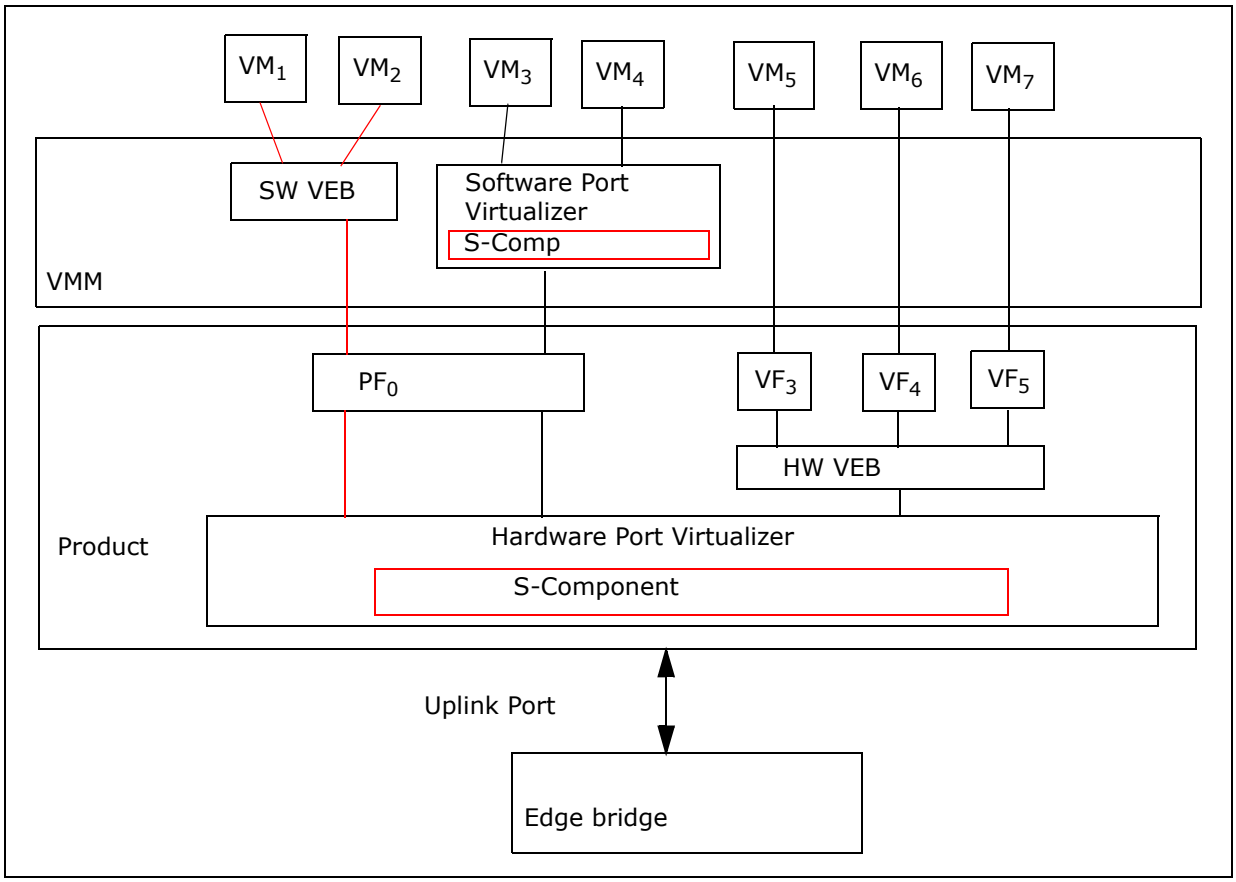
Security features such as MAC anti-spoofing and VLAN ingress checks should not be configured on the downlink ports since those functions are also performed at the ingress VSIs of the software VEPA.

The 10 GbE controller can be configured to perform VLAN offload functions on the downlink VSIs. In this case, the VLAN information is carried in the transmit descriptor and the 10 GbE controller inserts the VLAN in the transmit direction. The 10 GbE controller can strip the VLAN in the receive direction and optionally store the VLAN tag information in the receive descriptor.

Cascaded VEB and Port Virtualizers

The following figure shows a software VEB, hardware VEB and a software port virtualizer cascaded over a hardware port virtualizer. A port virtualizer logically extends the ports of an external bridge to virtual end stations as explained in Port Virtualizer Model. The 10 GbE controller supports a hardware-based port virtualizer for directly connecting virtual end stations to extended bridge ports using VFs in SR-IOV configuration.

Figure 38-41.Cascaded VEB and Port Virtualizer



Cascaded Software Port Virtualizer over a Hardware Port Virtualizer

The 10 GbE controller supports cascading a software-based port virtualizer in VMM to the hardware-based port virtualizer in the network controller. A software port virtualizer is configured in the VMM and the connection between the software port virtualizer and a port in the hardware port virtualizer is referred to as a cascaded VSI (or simply a downlink port). The cascaded interface in this case is a special VSI that is used as a downlink to a software port virtualizer. A VSI connection between a VM and a port virtualizer uses a single S-tag to identify the logical bridge port. However, a cascaded VSI (downlink port) connected to a software-based port virtualizer carries traffic to multiple software-based VSIs serviced by the software port virtualizer. So multiple S-tags need to be configured on this downlink VSI. Receive multicast replication functionality and source pruning functionality should be disabled on these downlink ports.

The 10 GbE controller supports VLAN and S-tag offload functions for software-based port virtualizers connected to the downlink VSI. The 10 GbE controller performs S-tag and VLAN tag insertion functionality on transmit direction and can optionally strip the VLAN tag and S-tags in the receive direction. In this case, the VLAN tag and S-tag information is carried in the transmit descriptor and the 10 GbE controller inserts the VLAN tag and S-tag in the transmit direction. The 10 GbE controller can strip the VLAN tag and S-tag in the receive direction and store the VLAN and S-tag information in the receive descriptor.

The 10 GbE controller can also classify ingress frames based on S-tag and forward to selected queues within the downlink VSI connected to software port virtualizers (similar to VMDq1 mode). The 10 GbE controller offload functions as previously described enhance the performance of software-based port virtualizers. See Section 38.21.7.11.2 for additional details on software port virtualizer offload functionality.

Note: A VSI used as a cascaded port virtualizer should not activate L2 filters on the aggregated S-channels. The VSI should be defined with a *connection type* of *default port*.

Cascaded Software/Hardware VEB over a Hardware Port Extender (Multichannel S-comp)

The 10 GbE controller supports software VEBs cascading to a hardware port virtualizer. In this configuration, a software VEB can be configured in the VMM and the connection between the software VEB and a port in the hardware port virtualizer is referred to as a cascaded VSI (or simply downlink port). The 10 GbE controller supports VLAN offload functions for software VEBs. The software VEB is not aware of the S-tag functionality. The hardware port virtualizer (or multichannel S-component) performs the insertion and stripping of S-tag to the cascaded VSI (downlink port) connected to the VEB. The 10 GbE controller can classify the ingress frames based on (VLAN, MAC address) pair and forward to selected queues within the downlink VSI connected to software VEBs (VMDq1 mode).

The 10 GbE controller supports hardware VEBs to hardware port virtualizer cascading. A hardware VEB is configured in the 10 GbE controller to support direct connected VMs in SR-IOV mode of operation. In this configuration, a hardware port virtualizer (or multichannel S-component) and a hardware VEB can be instantiated in the 10 GbE controller. The hardware VEB can be connected through a cascaded (or downlink) port in the hardware port virtualizer. Logically the hardware VEB's trunk port is connected to the external controlling bridge through a logically extended bridge port identified by an S-tag. The hardware port virtualizer (or multichannel S-component) performs insertion and stripping of S-tags to the cascaded downlink port connected to the hardware VEB.

38.21.7.3 Edge Virtual Bridging (EVB) Management Protocol

This section describes the control and management protocol for discovery and configuration of EVB components between the end stations and external edge switches. The current IEEE 802.1Qbg EVB supports multiple different switching elements: VEB, VEPA and for virtual networking. A control protocol is required to discover the capabilities of the end stations and external switches and configure the appropriate switching elements at both ends.

To enable EVB in the device, all of the following must be set:

- The GLGEN_STAT.EVBEN bit must be set to 1b either from the NVM or through a soft SKU.
- The NVM EVB protocols *Enabled* field should be set to 1b.

- The NVM *Switching Mode* field should be set to the EVB switching value.

EVB is configured and manages via the following stages:

- S-channels are enabled, defined and enumerated between the endpoint (station) and switch (bridge) by exchanging CDCP TLVs over LLDP. The 10 GbE controller sets a default VSI per S-channel.
- Software might then instantiate Edge Relay (ER) bridges, one per each S-channel.
- Support for the relay protocol (VEB/VEPA) of each ER is negotiated through LLDP EVB TLVs. Note that reflective relay might be enabled and common variables for VDP/ECP are negotiated.
- A reliable control protocol, Edge Control Protocol (ECP), is then established.
- A VSI discovery and configuration protocol (VDP) runs over ECP to associate VSIs with bridge ports

38.21.7.3.1 S-tag Allocation Through CDCP TLVs

CDCP TLVs are exchanges over LLDP.

Following initialization of an LLDP agent, and once the conditions for CDCP support are met, a local copy of the CDCP TLV is established and exchanged with the other end.

Once the device receives S-tag setting from the remote CDCP TLVs, it continues with EVB configuration. Specifically, the S-tags extracted from the CDCP MIB are handled as if received through CLP strings or NCSI depends on customer.

Mapping of the SCID/SVID pairs may change as a result of the following events:

- The LLDP agent reverts to the local MIB event. In such a case, all SCID/SVID pairs are deleted
- A CDCP TLV is received with a modified list of SCID/SVID pairs. SCID/SVID pairs may be removed, modified, or added.
- SCID/SVID pair removal: Remove all the VSIs associated with the VEB on this S-channel and remove the optional VEB associated with it.
- SCID change:
  - Remove all filters associated with VSIs on this SCID.
  - Change the switch ID and inserted S-tag for all the VSIs associated with this SCID using Update VSI command.
  - Add back all the removed filters.
- SCID/SVID pair addition: Add a VSI and optionally a VEB on top of this VSI.

38.21.7.3.2 VSI Setup Through VPD TLVs

A new protocol, VSI Discovery Protocol (VDP) is defined by IEEE. VDP protocol is transported over Edge Control Protocol (ECP) that enables an exchange of VSI specific TLVs between the edge switch and the control plane of appropriate switching element (VEB, VEPA or port virtualizer). The TLV transfer protocol is expected to use the same group multicast destination address(es) reserved for LLDP as previously described with a new EtherType (to be defined). A new EtherType enables forwarding the protocol frames to the appropriate control plane of the respective switching element.

VDP protocol uses TLVs to discover and configure the VSIs in the end station. The VDP protocol is used for establishing a new VSI and exchanging VSI attributes such as VSI ID, MAC, VLAN, QoS parameters, etc. The VDP protocol can also be used for

exchanging VSI statistics from end station to edge switches. The 10 GbE controller has the ability to filter for appropriate Edge Virtual Bridge (EVB) control frames and forward to appropriate management ports (VSIs or VSI queues) of the respective switching elements. The control frames (such as LLDP or VDP) are expected to be exchanged by the respective control plane agent in the VMM (or IOVM) and the external switch. LLDP is a link level protocol, so a VM is not expected to directly exchange LLDP control frames with an external switch. For example, the control frames cannot cross the virtual bridge. If a VM is configured to exchange control frames, it does so with the appropriate virtual bridge control agent in the VMM. The 10 GbE controller has a mechanism to either filter and/or forward the control frames from the VSIs to the management port of the switching element to which the VM is connected.

When a VEB or VEPA is instantiated over a port virtualizer, this is the equivalent of a VEB or VEPA uplink directly connected to a logical port in the external bridge. So the external bridge can exchange bridge control frames over the logical link to the VEB or VEPA. In this case the external bridge can send or receive LLDP frames to VEB or VEPA in the end station. These frames are identified using the S-tags that correspond to the extended bridge ports. Any of the bridge control frames can pass through this S-channel (like LACP or a new TLV transfer protocol/VDP). The bridge control frames over the S-channels are not expected to be VLAN tagged, so these control packets are untagged when delivered to the VEB or VEPA by the S-component. The 10 GbE controller can forward those control frames to the respective VEB or VEPA control ports (VSIs).

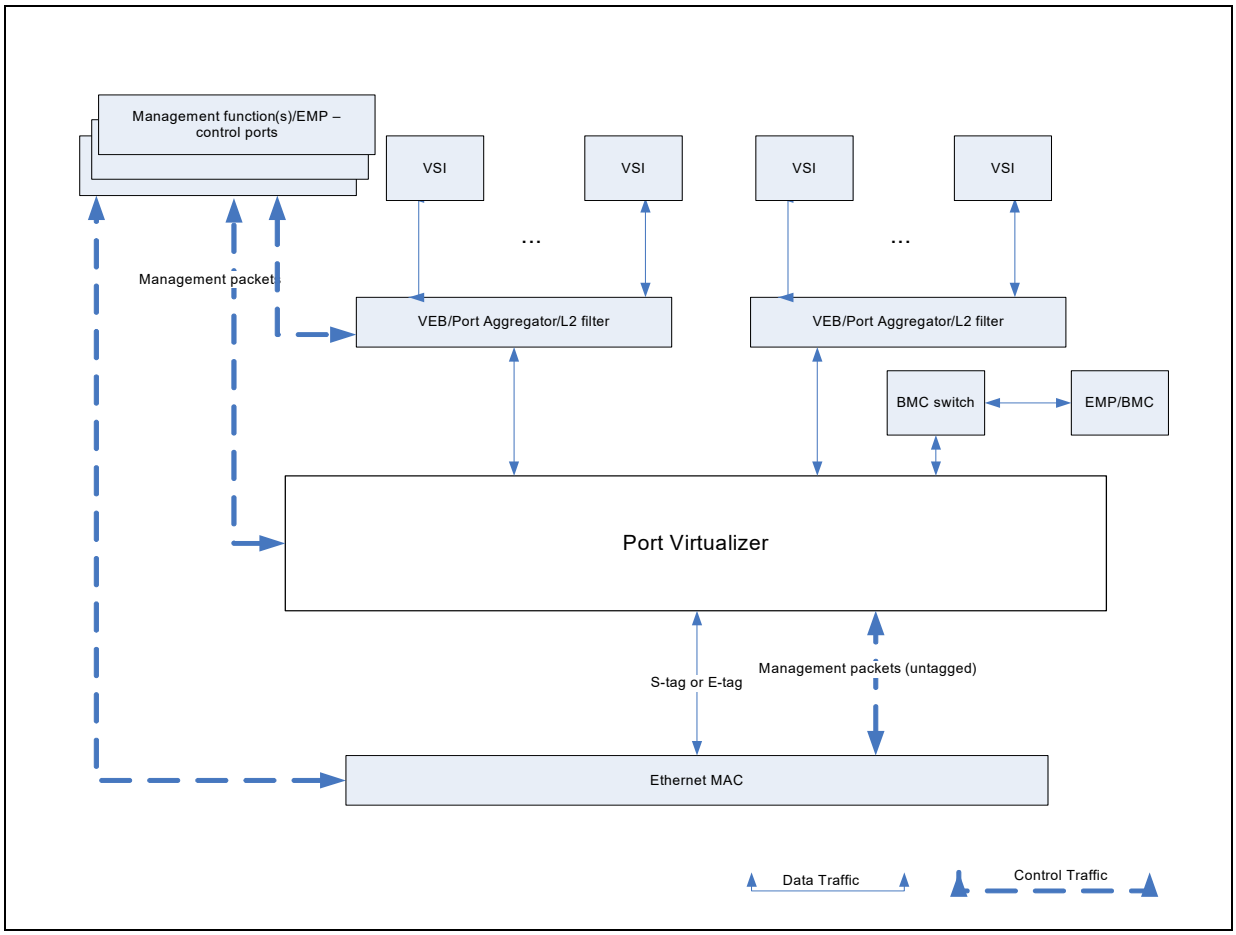
In a multi-channel configuration, there could be multiple LLDP agents (peer-to-peer) established between the external bridge through extended bridge ports (S-channels) and the corresponding VEB, VEPA or port virtualizer components in the end station.

Note: After a link down/link up event, the switch configuration protocol should be run again because the partner (and its capabilities) might have changed.

38.21.7.4 Switch Elements

The following figure shows the possible switching elements on one of the 10 GbE controller ports and the connections between them. Every switching element is optional. In the minimal case, the Ethernet port might be connected to a PCI function (VSI) with no switching capabilities at all. In this minimal case, an L2 filter is used as in regular NICs.

Figure 38-42.Switching Elements Per Port



There can be up to one port virtualizer per physical port. There can be up to 16 concurrent VEB or port multicasting defined in the device. There can be up to one L2 filter per VSI.

38.21.7.4.1 MAC

This section describes the switching decisions of the MAC.

The MAC switching behavior is set according to the following parameters:

- Port Enable
- Link Status
- Save Bad Packets (Stored in the *PRT_SBPVSI.SBP* field)
- Default Port VSI (Stored in the *PRT_SWT_DEFPORTS.DEFAULT_VSI* field)

- Local MAC Address (per LAN port)
- One or more control ports of the MAC. If a port virtualizer or a VEB is directly connected to this MAC, they might share the same control ports. At initialization time, a single control port is connected to the EMP. At a later time, if requested by the software device driver, an additional control port on a VSI of one of the physical function might be added or might replace the EMP control port.

As a rule, the MAC forwards all the packets to the next stage, apart from the following packets:

- Packets with L2 error — These packets are dropped, unless the *Save Bad Packets* attribute of the MAC is set. In this case, the MAC must define a control port and all the error packets are forwarded to this port. The errors included in this category are:
  - — CRC errors
  - — Alignment errors
  - — Length errors (either too big or too small)
- Packets with the MAC local MAC address. If a *Station MAC address* is defined, packets with this address might be forwarded to one of the control ports of the MAC in combination with some EtherType filters. Each control port might require forwarding of these packets using the *Add Control Packet Filter* or the *Add MAC, VLAN pair* admin commands.
- Link local packets without S-tag:
  - — Flow control packets (both 802.3x link flow control and 802.3bd priority flow control frames) are handled by the MAC and are not forwarded.
  - — Any packet defined in the control VSI forwarding rules (see Control VSI).

38.21.7.4.2 Port Virtualizer

An Ethernet port can be connected to a port virtualizer.

A port virtualizer can be either an S-comp, see Service-VLAN Component (S-comp), or a port extender (see Section 38.21.7.4.3). Each port virtualizer should be connected to a control port that receives the control protocol (CDCP or PE-CSP) packets.

There can be one port virtualizer per physical port. These two elements are mutually exclusive and in a given device only one type of port virtualizer can be initiated on all ports.

A port virtualizer can be created using the *Add Port Virtualizer* admin command described in Add Port Virtualizer.

If a port virtualizer is not implemented on a port, S-tagged packets are dropped.

Service-VLAN Component (S-comp)

An S-component is a typical mux/demux component that divides the Ethernet port to a set of S-channels.

Each channel in the S-component can be attached as a physical port to one of the switches. If no switch is needed, an S-comp is used to directly connect S-channels to VSIs.

S-tag Format

The S-tag format is the format defined for S-tags in the IEEE 802.1ad specification as listed in Table 38-187.

Table 38-187.S-tag Format

0	15	16	18	19	20	31
88-a8		PCP		DEI	S-VID	

- 88-a8 — The S-tag EtherType
- PCP — Priority Code Point (the priority bits of the S-tag). These bits can be derived from the VLAN priority bits.
- DEI — Drop Eligible Indicator (this bit is ignored)
- S-VID — the S-tag ID

S-comp Receive Flow

The S-comp acts only on packets with an S-tag received from the uplink. These packets are forwarded to one of multiple S-comp channels according to the S-tag forwarding table and then forwarded directly to VEB switches or VSIs.

Note: If no VEB is defined, then the VSI connected to the S-channel should be put in promiscuous mode using the Set VSI Promiscuous Modes command in order to receive all the traffic with the given S-tag.

Note: If an S-component is defined in the system, packets without S-tag or with an unrecognized S-tag not forwarded to one of the control ports are sent to the default port. If such a port is not defined, they are dropped.

The S-tag is usually removed by the S-component and is not forwarded to the host or to the sideband interface. This functionality is controlled by the *SS-tag extract mode* parameter of the *Add VSI* admin command. For cascaded S-comp ports that represent multiple S-channels, the S-tag can be either left in the packet or removed and stored in the *L2TAG2 (1st)* field of the receive descriptor. The *L2TAG2 (1st)* field of the receive descriptor contains the S-tag if the *S-tag extract mode* parameter of the VSI is set to 10b and is valid if the *L2TAG2P* flag in the descriptor is set. If S-tag extraction to a descriptor is required, the VSI must use 32-byte receive descriptors.

The algorithm used by an S-comp is described in Section 38.21.7.13.1.

S-comp Transmit Flow

S-comp adds the S-tag matching to the S-channel from which the transmit packet is received and forwards it to the physical port. It also forwards packets received from the control ports to the physical port.

Switch-based insertion of the S-tag can be disabled using the *S-tag insert enable* parameter of the *Add VSI* admin command. For cascaded S-comp ports that represents multiple S-channels, S-tag insertion should be disabled. In this case, software can either send untagged packets, ask for S-tag insertion via the transmit descriptor or add it in-line in the packet. This functionality is enabled by the *Accept Tag from host* parameter in the *Add VSI* admin command.

An S-comp only transfers packets from the S-channels to an Ethernet link or vice versa. It does not replicate packets or forward packets between virtual ports or physical ports. The only exception to this rule are local iWARP packets, where the connection is

defined as loopback in which case the packet is not tagged and is sent back to the same VSI. A VEB connected on top of an S-channel can forward packets between VSIs contained by this VEB.

38.21.7.4.3 Port Virtualizer (S-comp) Parameters

A port virtualizer is defined by the following parameters:

1. The port virtualizer connectivity:
   a. Uplink connected.
   b. Control ports that receive link local packets. Note that the control ports might be VSIs or the internal embedded controller or both.
2. Demux rule — The rule used to demux ingress packets. Typically, this rule uses an S-tag as previously described.
3. Default port — This port receives all the traffic not identified as control packets and not forwarded to any of the S-channels. If this port is not defined, such traffic is dropped.
4. Set of channels — Each channel is defined by the following parameters:
   a. Connected VSI or next level switch.
   b. S-tag (S-VID) - see S-tag Format.
   c. Maximum and minimum rate.
   d. S-tag strip enable.
   e. S-tag insert enable.
   f. Report S-tag.
   g. Accept tag from host.

LAN S-comp

The LAN port extender divides the uplink port traffic. The LAN channel can be attached to a switch (for example a VEB).

This S-comp differs from the standard 802.1Qbg port extender in that it uses the MAC address as the channel classifier. The MAC address behaves the same way as the S-tag, and it can translate to a single switch ID. In this case, the traffic is untagged (no S-tag).

Receive Flow

The S-comp filters on unicast and multicast addresses are forwarded to the LAN channel.

Note: The MAC address is used twice in the switch process, first to define the S-channel and second to define the VSI within this S-channel.

Contrary to other port extenders, the MAC address is never stripped and it is always forwarded to the host.

The algorithm used by LAN S-comp is described in Section 38.21.7.13.

Transmit Flow

The S-comp only transfers packets from the S-channels to an Ethernet link or vice versa. It does not replicate packets or forward packets between virtual ports or physical ports. The only exception to this rule are local iWARP packets, where the connection is defined as loopback in which case the packet is not tagged and is sent back to the same VSI.

No tag is added to the transmitted packet.

Port Virtualizer Control Ports

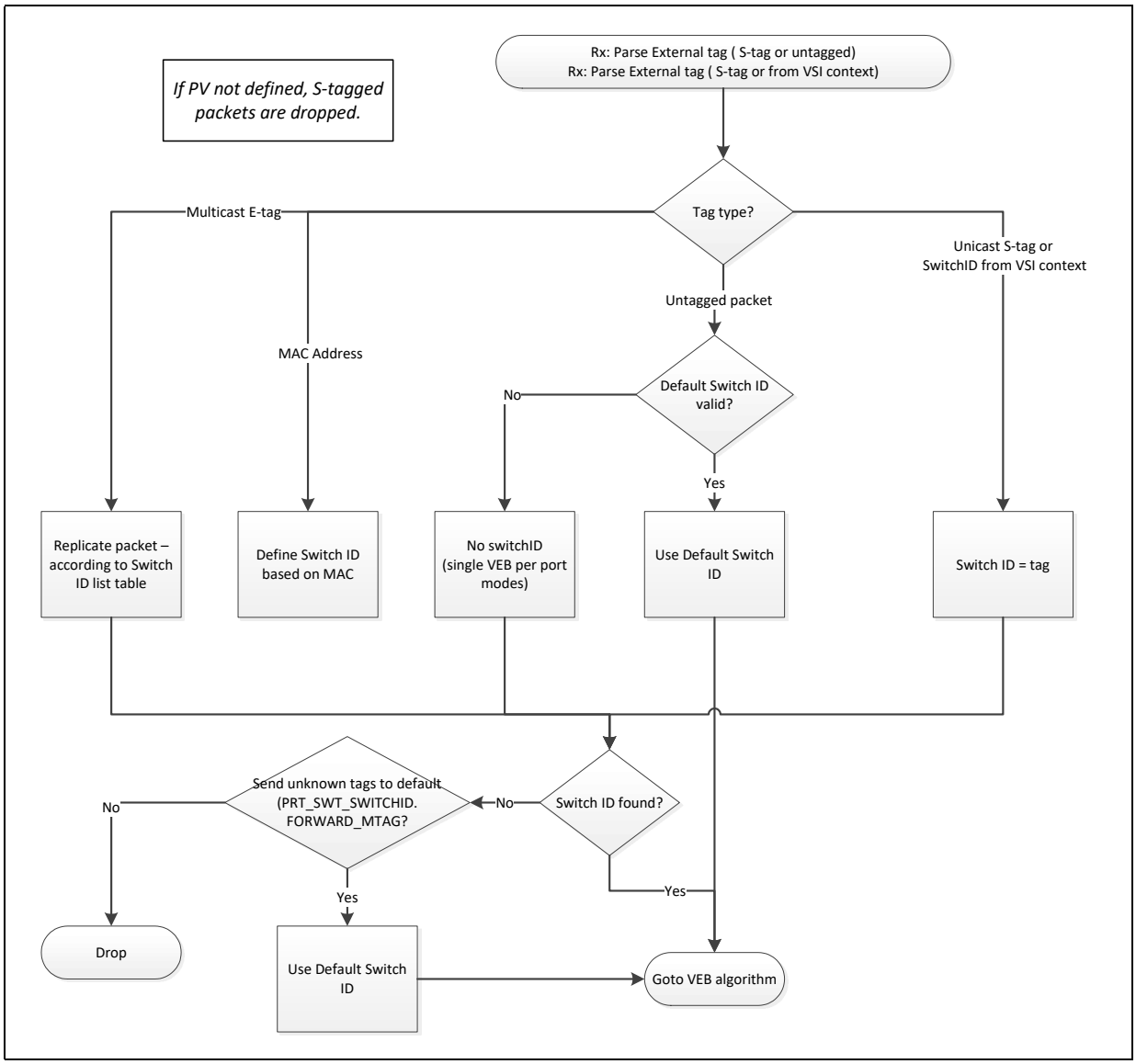
The port virtualizer might forward control packets to the associated control port.

Port virtualizer control packets are untagged LLDP or enhanced LLDP packets combined with specific MAC addresses as described in Section 38.21.7.3. Each control port might require forwarding of specific control packets forwarding using the *Add Control Packet Filter* admin command.

Port Virtualizer Flow Diagram

The following describes the high level flow used to define the switch ID associated with a packet. The details can be found in the algorithm sections (Section 38.21.7.13).

Figure 38-43.Port Virtualizer Flow



38.21.7.4.4 L2 Filters

A VSI directly connected to a physical port, to a port extender or to an S-comp, still needs the ability to filter incoming traffic as done in traditional NICs. The L2 filtering element provides this ability. An L2 filtering element is relevant only for receive or sideband traffic and does not apply to transmit traffic. An L2 element has no control port and is controlled directly by the function.

An L2 filter element is not represented as part of the topology and is implicitly created by adding L2 filters to existing VSIs.

This section describes L2 filters used when no VEB or VEPA element is initiated. If a VEB or VEPA element is added, the L2 filtering is done as part of the VEB forwarding as described in the following sections.

An L2 filtering element includes a set of filters used to define whether a packet is received by the function or is dropped. There are two type of filters: exclusive filters that identify the packet as exclusive to this function and non-exclusive filters that might pass as well as packets that might be forwarded to other ports or functions.

Non-exclusive traffic can be further defined as perfect or imperfect, where perfect means that a packet that passed these filters was requested by the host, and imperfect filters can pass packets the host did not request.

In addition, a packet might be filtered out if it is larger than the maximum size defined by the port.

Note: Even if a packet passed the length filter of the port, it might yet be filtered by the length limitation of the receive queues.

An L2 filter is applicable only to receive traffic and does not refer to transmit traffic. Thus, if there is only an L2 filtering element, transmit traffic is forwarded to the network. The only exception is the case of a local iWARP connection. A packet sent by this connection is a loopback to the sending VSI.

Note: The cloud filtering includes an L2 filter as part of its VEB or VEPA. This L2 filter is described in Section 38.21.7.4.5.

Exclusive Filters

The following exclusive filters are available as part of an L2 filtering element:

- MAC:VLAN Unicast Table — Used to forward to a port packets matching both the MAC and VLAN pair. A VLAN value of zero includes untagged packets. Only unicast addresses are considered as exclusive.
- MAC Unicast Table — Used to forward to a port packets matching MAC addresses ignoring the VLAN tag.
- VLAN Table — Used to define the VLAN to which the port belongs. A VLAN value of zero indicates the port can receive untagged packets.
- EtherType Table — Used to forward to a port packets matching an EtherType ignoring the MAC address and VLAN tag.

Non-Exclusive Filters

The following non-exclusive perfect filters are available as part of an L2 filtering element:

- MAC VLAN Multicast Table — Used to forward to a port packets matching both the MAC and VLAN pair. A VLAN value of zero includes untagged packets. Only unicast addresses are considered as exclusive.
- MAC Multicast Table — Used to forward to a port packets matching MAC addresses ignoring the VLAN tag.
- UPE — Promiscuous Unicast
- MPE — Promiscuous multicast
- BAM — Accept broadcast packets

The following non-exclusive imperfect filters are available as part of an L2 filtering element:

- HashMAC, VLAN, Address Type — Used to accept packets whose multicast/unicast MAC address match a given MAC address and their VLAN match the VLAN tag in the pair. A VLAN value of zero includes untagged packets.
- HashMAC, Address Type — Used to accept packets whose multicast/unicast MAC address match a given MAC address.

The algorithm used by an L2 filter is described in Section 38.21.7.13.3.

38.21.7.4.5 Virtual Ethernet Bridge with Cloud Support (Cloud VEB)

A VEB is used to switch packets between a set of virtual ports, a physical port or an S-channel and a control port. The VEB behaves like a managed 802.1d transparent switch. All the configuration of the switch is done by a software agent. There are no auto configuration capabilities in the VEB. For example, there is no auto-learning of MAC addresses.

A VEB can forward packets received from any of its ports to a list of ports (including, optionally the originating port. The VEB uses the rules described in Cloud VEB Switching Rules to define the egress ports of each packet.

A packet is associated with a VEB using the SwitchID associated to it. A switchID can be either 0, S-tag or a switch ID based on the SwitchID VSI parameter or the Port SWID parameter (*PRT_SWT_SWITCHID.SWID*).

There can be up to 16 VEBs or port aggregators in the 10 GbE controller. They can be assigned dynamically to PFs. All the virtual ports associated with a VEB must belong to a single PF or its VFs.

Note: A VEB can be added only by a PF and cannot be added by a VF or the EMP.

A VEB can be created using the *Add VEB* admin command described in *Add VEB.*

A cloud VEB can also be configured to VEPA mode. The exact algorithm used by a cloud VEB is described in Section 38.21.7.13.4.

VEB with Cloud Support Parameters

A VEB is defined by the following parameters:

1. A control port that receives link local packets (such as 802.1X packets or LLDP packets).
2. An optional default port that can receive packets received from the physical port and not forwarded to any virtual port.
3. Definitions of local and private VLANs.
   a. Local VLANs — A list of VLAN tags defined as local that do not include the physical port.
   b. Private VLANs — A list of VLAN tags defined as private. For each private VLAN tag, define the promiscuous ports list, the community ports list, and the isolated ports list.

Note: Refer to the special VLANs section for more details.

4. A set of VSIs — Each VSI is defined by the following parameters:
   a. Connected PCIe function.
   b. Transmit enable and receive enable.
   c. Set of forwarding tables and parameters rules used to forward packets to the port.
   d. Security features:
      - Port-based VLAN insertion
      - Anti-spoofing enables
   e. Optionally, a maximum and minimum rate.

Cloud VEB Switching Rules

The following parameters are used to define which egress ports (VSIs and LAN) receives a packet received by the VEB.

Forwarding Tables and Parameters

- Priority 1 filters (control filters):
  — EtherType Table — Used to forward to port(s) packets matching an EtherType ignoring the MAC address (usually used for the control VSI).
  — MAC, EtherType Table — Used to forward to port(s) packets whose MAC address match a given MAC address and EtherType match an EtherType (usually used for the control VSI).
- Priority 2 filters (cloud filters):
  — Inner MAC, Inner VLAN (for NVGRE, VXLAN or Geneve packets)
  — Inner MAC, Inner VLAN, Tenant ID (for NVGRE, VXLAN or Geneve packets)
  — Inner MAC, Tenant ID (NVGRE packet or VXLAN/Geneve packets).
  — Inner MAC filter
  — Outer MAC, Tenant ID, Inner MAC filter
  — Inner IP filter
  — Inner Source IP, inner destination MAC filter

Note: The network key is extracted from the GRE or UDP headers in MAC in GRE or MAC in UDP packets.

GRE key is a 4-byte field enabled by the k flag in the GRE header. The key is extracted only if the GRE header includes a key as indicated by active K flag in the header.

The UDP port used for MAC in UDP tunneling is defined by a shared 16-entry table with Tunneling port numbers. The key is extracted if the UDP port number contains a key as defined by the UDP protocol index, which is programmed by the *Add Tunneling Port command*.

- Priority 3 filters (L2 Filters):
  — MAC, VLAN Table — Used to forward to VSI(s) packets matching both the MAC and VLAN pair.
  — MAC Table — Used to forward to a VSI(s) packets matching MAC addresses ignoring the VLAN tag.
  — HashMAC, Packet Type — Used to forward to port(s) packets whose multicast or unicast MAC address match a given MAC address. Packet type is unicast or multicast.

— HashMAC, VLAN, Packet Type — Used to forward to port(s) packets whose Multicast or Unicast MAC address match a given MAC address and their VLAN match the VLAN tag in the pair. Packet Type is unicast or multicast.

— VLAN Table — Used to define the VLANs to which a VSI(s) belongs for egress and ingress checks. This table can point to two lists: one list that is used for ingress VLAN filtering and one used for egress VLAN filtering.

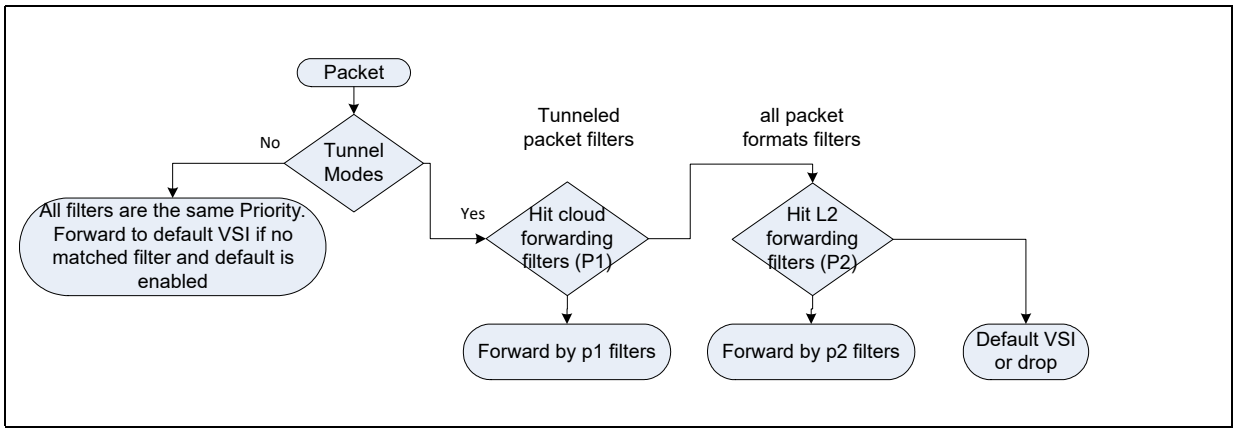
Note: In all previous tables, a VLAN value of zero includes untagged packets.

A port in these tables means either a VSI or the LAN port for transmit packets and a VSI for receive packets.

- Promiscuous modes per VSI:
  - — UPE — Used to forward to a VSI, all unicast packets.
  - — MPE — Used to forward to a VSI, all multicast packets.
  - — BAM — Used to forward to a VSI, broadcast packets.
  - — VPE — Used to forward to a VSI, packets with any VLAN tag.
- Loopback rules:
  - — ALLOWLOOPBACK — Allowed to send packets to other virtual ports
  - — PruneEnable — This mode is useful to enable offload of a software switch connected to this virtual port.
- Anti-spoofing parameters — These parameters define if a packet should be allowed to enter the switch.
  - — MAC anti-spoofing enable
  - — VLAN anti-spoofing enable

The forwarding filters are divided into groups. Those filters that are needed for tunneled packet formats and those ones that are shared for all packet formats (based on the outer MAC or VLAN). Filters that match tunneled packet formats take precedence on those filters that match all packet formats.

Figure 38-44.Cloud Forwarding Filters Priority



The Add Control Packet Filter (0x025A) and Remove Control Packet Filter (0x025B) commands are used to add or remove control filters settings. Each control filter can point to a single VSI. There is no support for packet replication based on control filters. An error is returned if an existing control filter is added to point to another VSI.

The Add Cloud Filters (0x025C) and Remove Cloud Filters (0x025D) commands are used to control cloud-specific filters settings. Each cloud filter can point to a single VSI. There is no support for packet replication based on cloud filters. In addition to those filters, in order for a packet to be received by one of the VSIs, it must also pass the L2 MAC filter. In order to enable L2 filtering, the *Flags.Enable L2 filtering* bit in the *Add VEB* command should be set when creating the cloud VEB.

Note: This L2 filtering is not related to the L2 filtering defined in Section 38.21.7.4.4.

- The Add MAC, VLAN Pair (0x0250), Remove MAC,VLAN Pair (0x0251), Add VLAN (0x0252), and Remove VLAN (0x0253) commands are used to set regular MAC/VLAN filters.

Note: The *Add MAC, VLAN pair* command enables filtering based on a MAC address (any VLAN) or on a MAC, VLAN pair. The *Add VLAN* command is used to define the VLAN membership of ports and is not used to add destination VSIs. Using this method, the VEB can allow MAC-based filtering and promiscuous modes within specific VLANs.

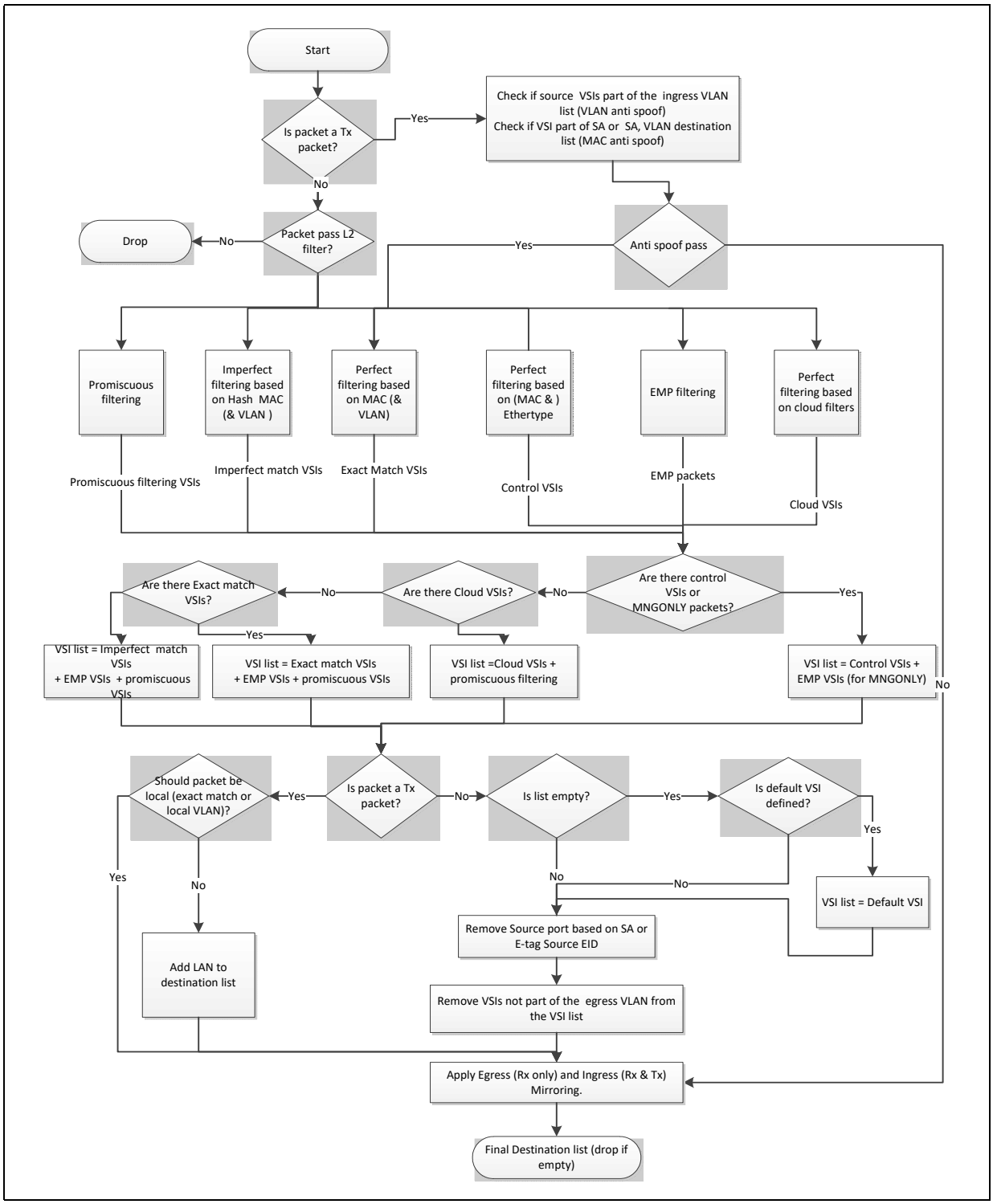
- The Set VSI Promiscuous Modes (0x0254) command is used to set promiscuous filters.
- The Add Mirror Rule (0x0260) and Delete Mirror Rule (0x0261) commands are used to control mirror rules.

Note: It is assumed that manageability packets and control port packets are not encapsulated.

Cloud VEB flow

The following figure shows the generic flow of the cloud VEB element. The exact algorithm implementing this flow is described in Section 38.21.7.13.4.

Figure 38-45.Cloud VEB Flow Diagram



38.21.7.5 Port Aggregator

The port aggregators (VEPA) usage model is described in the "Virtual port Aggregator Model" section.

38.21.7.5.1 Port Aggregator Switching Rules

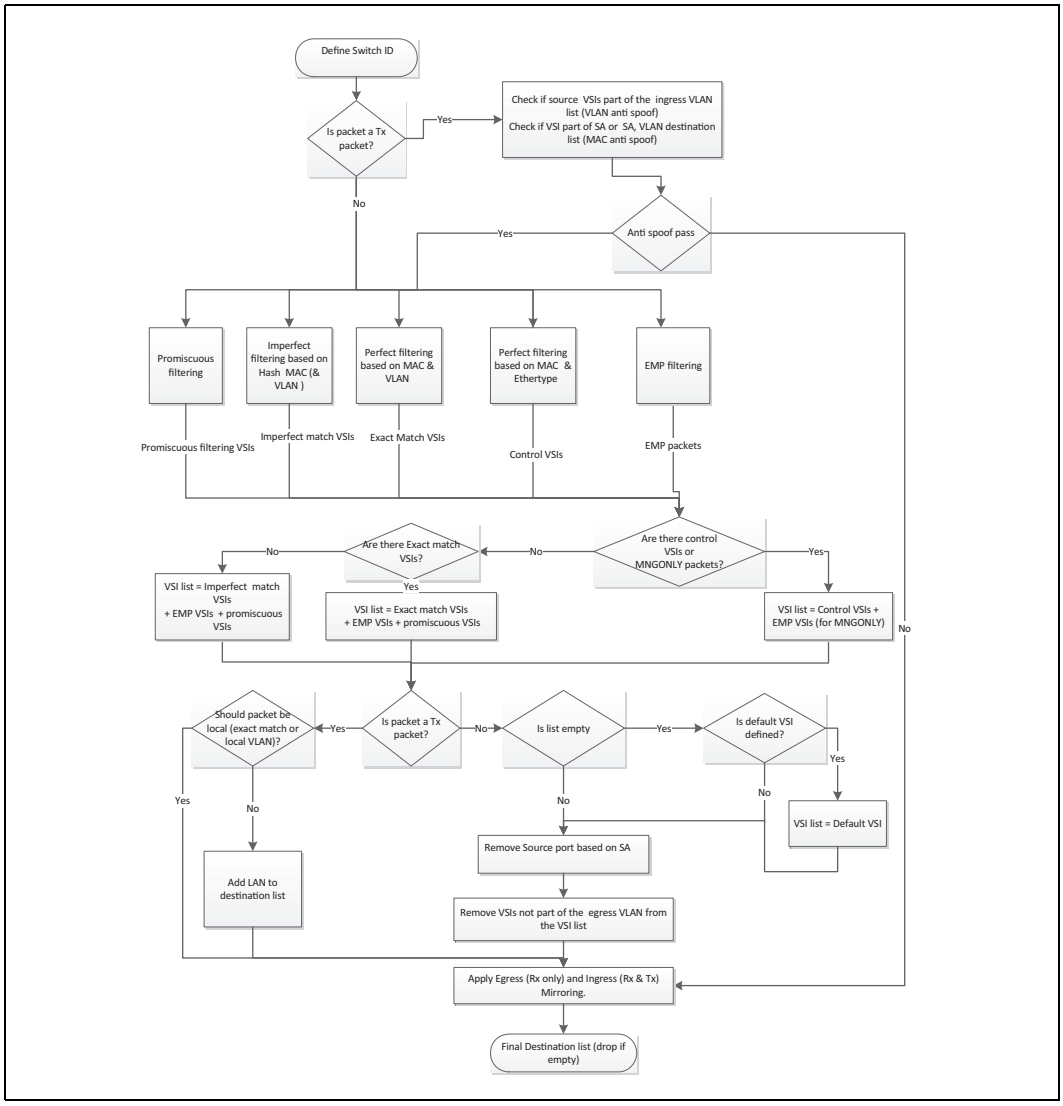
The behavior of a port aggregator is the same as the behavior of a VEB. The only difference is that in a port aggregator, loopback should be disabled for all virtual ports (ALLOWLOOPBACK = 0b).

A VEPA can be created using the *Add VEB* admin command described in *the* "Add VEB" *section* and by disabling loopback in the VSIs connected to this VEB.

38.21.7.5.2 Port Aggregator Flow

Figure 38-46 shows the generic flow of the VEPA element. The exact algorithm implementing this flow is described in Section 38.21.7.13.4.

Figure 38-46.VEPA Flow Diagram



38.21.7.6 Floating VEB

A floating VEB is a VEB not connected to the network, allowing only local traffic between the VSIs members of this VEB. This is useful in case one of the VSIs acts as a gateway to another network for all the other VSIs within the floating VEB. It can be used to isolate a set of VSIs behind a firewall or to implement NFV functionalities.

A floating VEB is created by setting the *floating VEB* flag and setting a *downlink SEID* of zero in the Add VEB command. The AQ response returns the switch ID used for this floating VEB.

Traffic in a floating VEB is identified using a special S-tag inserted and removed by the hardware. Hence the C*ascaded Port Virtualizer section is valid* bit should be cleared in floating VEB VSIs. Firmware internally sets:

- S-tag = Switch ID of the floating VEB
- Switch ID = Switch ID of the floating VEB
- S-tag extract mode = 01b
- S-tag insert enable = 1b
- Accept tag from host = 0b

In addition, the *Allow Loopback* flag should be set.

Note: A VSI in a floating VEB might still bypass the switch if allowed via the SWTCH flag in the Tx descriptor. A packet sent to the LAN via this mechanism goes out on the VEB with the floating VEB internal S-tag. This is not an expected use case, as only trusted VSIs are allowed to use the SWTCH flag.

Note: As VSIs on a floating VEB adds and remove an S-tag, the RXMAX value of the Rx queues in these VSIs should be updated to account for the additional 4 bytes.

Note: Although floating VEB creates an isolated environment for the PF, it cannot completely disconnect the PF from the network. A try to call the Delete Element AQ (0x0243) for VSI, connected directly to port results in an error.

38.21.7.7 Manageability Sideband Switching

The network interface might be shared between the host and a sideband manageability interface. The sideband interface can be either over a 100 Mb/s Ethernet interface using NC-SI, over SMBus using a legacy pass through mode or NC-SI over MCTP transport, or over PCIe using NC-SI over MCTP transport.

The specific filters used to define which packets are forwarded to the sideband interface. The current section only describes the relationships between the sideband and other switching elements.

38.21.7.7.1 Network to BMC Filtering

The relationships between the sideband filtering and other switching elements follow:

- If no switching elements are defined such as no VEB or port virtualizer), the sideband decision filters are applied to the network traffic. Packets which pass these filters are forwarded to the sideband interface.
- If an S-comp is defined on the port, the sideband might be defined as part of one of the channels in the S-comp. In this case, the sideband decision filters are applied only to the traffic with the matching S-tag. The same S-tag is added to packets sent from the sideband interface.

- The sideband switching is never part of a VEB, port aggregator or port virtualizer and is done in parallel to the forwarding decisions of these elements.

38.21.7.7.2 OS to BMC Traffic Filtering

The following rules are used to decide whether OS traffic should be sent to the sideband interface:

- If no switching elements are defined, for each function associated with a given port, the sideband decision filters associated to this port and applicable to host traffic are applied to the function traffic. Packets that passes these filters are forwarded to the sideband interface. If the sideband interface defines these packets as exclusive (*PRT_MNG_MNGONLY* is set for this filter), the packets are not sent to the network. Otherwise, they are sent both to the sideband interface and to the network.
- If an S-comp is defined on the port, the sideband may be defined as part of one of the channels in the S-comp. In this case, the sideband decision filters are applied only to host traffic with the matching S-tag. Traffic from the host to the BMC is sent to the BMC without the S-tag.

Note: It is assumed that traffic sent from an S-channel different than the S-channel of the BMC is forwarded by an external switch.

- If a VEB, port aggregator or Port Virtualizer is defined, they will not receive packets exclusively sent to the sideband interface.

38.21.7.7.3 BMC to OS Traffic Filtering

The following rules are used to decide whether traffic received from the sideband interface should be sent to the host:

- If no switching elements are defined, the port L2 filters are applied to the sideband traffic. Packets that passes these filters are forwarded to the host. Unicast packets that pass exclusive L2 filtering as defined in "Exclusive Filters" are sent only to the host. Multicast or broadcast packets and unicast packets that pass non exclusive filtering as defined in "Non-Exclusive Filters", are sent both to the host and to the network. Other packets are sent only to the network.
- If a VEB or port aggregator are defined, the switching algorithms of the VEB or port aggregator are applied to the sideband traffic. Packets which pass these filters are forwarded to the host. Unicast packets that pass perfect L2 filtering as defined in "Non-Exclusive Filters" are sent only to the host. Multicast or broadcast packets and unicast packets that pass imperfect filtering as defined in "Non-Exclusive Filters", are sent both to the host and to the network. Other packets are sent only to the network.
- If an S-comp is defined on the port, the sideband might be defined as part of one of the channels in the S-comp. In this case, the switching elements relevant to this channel are applied to the sideband traffic. Traffic from the BMC to the host has no S-tag appended.

Note: It is assumed that traffic sent from the S-channel of the BMC to a function on a different S-channel is forwarded by an external switch.

- If a port virtualizer is defined on the port or S-channel of the sideband interface, it does not receive traffic from the BMC directly. It might still receive such traffic through an external switch.

38.21.7.8 Out-of-band Filtering Operation

Part of the type of traffics are expected to be operational also in Sx (equivalent to D3 or Dr state of the device). The following types of network inbound traffic are included in this category:

- Wake-up per PF
- Manageability traffic (two channels)
- LLDP traffic (untagged only).

In order to support this traffic, a separate simplified switch mechanism (PF classifier) is provided that is independent of the regular switch and is used to forward packets to these specific locations.

38.21.7.9 Ports

38.21.7.9.1 Physical Ports

Physical ports are considered as uplink ports and are not part of the L2 forwarding table. In each switch, there can be one physical port only. All the traffic that is not forwarded to one of the other ports is sent to the uplink egress port. In addition, multicast or broadcast traffic not received from the uplink ingress port is always forwarded to the uplink port, unless it is part of a local VLAN or the switch has no connection to an uplink. Local VLANs are defined in Local VLANs.

Ethernet ports

Each of the Ethernet ports can be assigned as an uplink port of a switch if not connected to an S-component. In this case, all the traffic received from this Ethernet port is handed to the switch.

38.21.7.9.2 VSIs

VSIs are the connections from the switch to entities interfacing with the host. It can be either the entire queue set of a PCIe function or part of the queues of a function.

There can be up to 384 VSIs in the 10 GbE controller. All the VSIs are equivalent.

At initialization time, each PF is assigned a VSI. A default VSI can also be used to provide the VMDq1 or control VSIs functionality described in the text that follows.

Other VSIs can be assigned to the PF or to VFs and used for the following purposes:

1. VMDq2
2. VMDq1
3. Control ports
4. iWARP traffic
5. Mirroring

In addition, EMP VSIs can also be defined. The EMP VSIs are used for:

1. Pass through traffic
2. Control ports

The different types of VSIs and their attributes are described in the following sections.

Mapping of queues to VSIs is described in Mapping of Functions and Queues to VSIs.

Host VSIs Types

VMDq2 VSI

The 10 GbE controller supports offloading of a VMM software switch. In this mode, the switching between VMs is done using a VEB element or a port virtualizer element. However, all or part of the ports might not be directly connected to the VEB/port virtualizer as SR-IOV functions (VFs) as they can be hidden behind the VMM or a service VM. These VMs can be represented to the VEB as groups of queues in a single physical function. There can be up to 256 VSIs for VMDq2.

The only difference between a VMDq2 VSI and a regular VSI associated with the PF is the ability to apply the VM reset flow.

VMDq1 Cascaded VEB/S-comp VSI

A VSI might be used for VMDq1 offload. There are two types of VMDq1 offloads:

- A cascaded VEB
- A cascaded S-comp

In a cascaded VEB, a software switch is tied to this VSI and uses the MAC, VLAN pairs to queue packets to the right receive queue.

In a cascaded S-comp, the VMDq1 VSI is used to offload a software-based S-comp. A cascaded S-comp VSI is created by setting the *Cascaded Port Virtualizer* flag in the *Add VSI* command. When an S-channel is used to convey multiple S-tags, it should be connected directly to a VSI; VEB or VEPA elements should not be connected on top of it. Filtering of packets to be sent to a cascaded S-comp VSI should be based only on S-tag and not on MAC, VLAN or other filters.

Control VSI

A single PCI function can be used to control one or multiple switching elements. For example, a single function can be used to control a port virtualizer and a VEB element associated with the function. Designers need to separate the control packets of different elements to different queues in receive and identify the switch element that should process the packets in transmit, a separate control VSI should be defined for each element.

A control VSI can receive part or all the link local traffic received by the controlled element. For example, a port virtualizer control VSI should receive LLDP untagged packets. A VEB control VSI behind a port virtualizer gets LLDP packets tagged with the S-tag associated with this VEB. As the control packet's traffic might be handled by a different control port, each control port driver should request forwarding of the relevant packets using the *Add Control Packet Filter* admin command.

Any VSI can be used as a control VSI as long as the adequate packets are routed to it. A control VSI should have the *Allow Destination Override* flag set to enable it to bypass the switch when sending packets.

At initialization time, the control VSI of the MAC is assigned to the EMP. If at a later stage, one of the PFs decides to take ownership of this control port, it should assign one of its VSI as the control port of the MAC. The EMP should be notified of the change using *Stop LLDP Agent* command and should disconnect the EMP control port. The PF might elect to add a control port in addition to the EMP control port. If a control port is added to a VEB or a VEPA (either connected to the LAN via a port virtualizer or floating), it does not impact the connectivity MAC control port to the EMP. A control port

of a VEB directly connected to the MAC might use the MAC control port or might use a separate VSI. After a port is defined, the owner of the control port (firmware or software device driver) should set the right filters using the *Add Control Packet Filter* admin command.

A typical configuration might be:

- Untagged (no S-tag) LLDP packets with a nearest bridge (01-80-C2-00-00-0E) address are forwarded to the control port of the MAC or of a switch element directly connected to the MAC (port virtualizer or VEB) on the EMP. These packets include DCBx TLVs.
- Untagged (no S-tag) LLDP packets with a nearest non-TPMR (01-80-C2-00-00-03) or nearest customer bridge (01-80-C2-00-00-00) addresses are forwarded to the control port of the switch element directly connected to the MAC (port virtualizer or VEB) on the host. These packets includes CDCP TLVs.
- Untagged ECP packets are forwarded to the control port of the switch element directly connected to the MAC (port virtualizer or VEB) on the host. These packets includes VDP and PE CSP TLVs.
- S-Tagged LLDP packets with a nearest customer bridge (01-80-C2-00-00-00) address are forwarded to the control port of a VEB connected to this S-channel.

Note: A control port can also be used as a default port.

Transmitting Packets from a Control VSI

A control VSI might need to send directed multicast packets. For example, sending an LLDP packet with a link local multicast address to the link partner or to one of the VSIs. According to the regular forwarding rules of the switch, such packets are forwarded back to the control port or dropped. To overcome this, the control VSI should set the *SWTCH* field and optional the *DEST_VSI* field in the transmit context descriptor to indicate the desired destination of the packet.

iWARP VSI

In some operating systems, the iWARP and the LAN functionality uses different MAC addresses. The LAN traffic uses the default VSI and the iWARP traffic (both stateless and stateful) uses a separate VSI. Both VSIs are part of the same PCIe function. There can be up to 16 VSIs for iWARP traffic in PFs and up to 32 VSIs for iWARP traffic over VFs.

An iWARP VSI is not created differently than a data VSI. After the creation of the VSI, iWARP resources should be assigned to this VSI.

Mirroring VSI

A VSI might be used to direct mirror traffic as defined by the mirroring rules in the associated VEB/VEPA. In this case, such a VSI is not expected to receive any traffic apart from the mirrored traffic. Any VSI in the VEB can be set as a mirror port.

A mirror VSI should be set to promiscuous VLAN mode to enable reception packets from any VLAN.

Default VSI

In each port, or each VEB one VSI can be defined as the default VSI. A port default VSI is available only if a port virtualizer is instantiated on this port; otherwise, a VEB default port should be used.

A port default VSI is added by defining the connected VSI port type in the *Add Port Virtualizer* command to Default.

A VEB default VSI is added by defining the downlink VSI port type in the *Add VEB* command to Default.

If a default port is defined, all the received traffic within the port/VEB not matching any forwarding rule is sent to the default port; otherwise, it is dropped.

Transmit traffic is not impacted by the default port. Unmatched transmit traffic is sent to the LAN.

A default port is subject to VLAN filtering rules. In order to allow all the unmatched traffic to be default VSI, it should be set to promiscuous VLAN using the *Set VSI Promiscuous Modes* command.

EMP VSI Types

Pass-through VSI

A pass-through VSI is used to send and receive traffic from an out of band manageability interface. If this VSI is located behind a Port Virtualizer, S-tag insertion and removal for packets sent to or received from the manageability interface is done by the device. VLAN tagging is done by the external BMC.

Control VSI

An EMP control VSI behaves as a host control VSI (see Control VSI). It sends and receive packets without S-tag and can send or receive VLAN-tagged or VLAN-untagged packets.

Mapping of Functions and Queues to VSIs

The function to which each VSI belongs is indicated by the VSI_VSI2F register in the VSI context. This register is initiated by the *Add VSI* admin command. See Add VSI (0x0210) for more detail.

Transmit queues are grouped in queue groups. Each queue group is associated with a VSI, thus creating an association between transmit queues and VSIs. This association is used when applying the VSI policies to transmit traffic.

Receive queues are mapped to VSIs using the *VSILAN_QTABLE* registers. These registers are also initiated by the *Add VSI* admin command.

Mapping can be either contiguous within the function queues or scattered. In case of scattered mapping, up to 16 queues can be associated with a VSI. Both receive and transmit queues should be associated within the queue set of the VF/PF to which this VSI is associated. So for VSIs associated to a VF, up to 16 queues can be associated with the VSI. For VSIs associated with a PF, all the queues of the PF can be associated with a VSI.

A VSI might choose to queue packets to different receive queues either using the RSS and TC information or using some other dedicated filters.

VSI Connections to the Switching Elements

Most VSIs can be connected to the following switching elements as regular ports:

- MAC, VEB, port virtualizer or port aggregator

- Port virtualizer

VSI Context

The context of a VSI contains the following parameters.

Table 38-188.VSI Context (Sheet 1 of 2)

Parameter	Description	Register.Field	Notes
Function pairing			
Function Type	Defines the owner this VSI can be either a PF, a VF or the EMP.	*VSI2_VSI_VSI2F.FUNCTIONTYPE*	00b = VF. 01b = VM. 10b = PF. 11b = EMP.
VF Function Number	The VF number of the function this VSI belongs to.	*VSI_VSI2F.VFVMNUMBER*	
PF Function Number	The PF number of the function this VSI belongs to.	*VSI_VSI2F.PFNUMBER*	This field should be set also for VFs.
VSI_ENABLE	Enables transmit and receive from VSI.	VSI_VSI2F.VS_ENABLE	
Switching parameters			
SwitchID	Defines the Switch to which this VSI belongs. If a VSI connected to a VEB/VEPA/MAC, this should be the same as the uplink switch ID. In case of a VSI connected to a port virtualizer, this is the S-tag associated with the S-channel of this VSI.	*VSI_SRCSWCTRL.SWID[11:0]* *VSI_SRCSWCTRL.ISNSTAG* *VSI_SRCSWCTRL.SWIDVALID*	The ISNSTAG should be set if the switchID is not an S-tag.
Enable VLAN Anti-spoof		*VSI_SRCSWCTRL.VLANAS*	
Enable MAC Anti-spoof		*VSI_SRCSWCTRL.MACAS*	
Allow Destination Override	Enables this VSI to set the destination of a packet.	*VSI_SRCSWCTRL.ALLOWDESTOVERRIDE*	
Enable Local Loopback	Allows forwarding to a VSI of packets sent from this VSI.	*VSI_RXSWCTRL.PRUNEENABLE* (inverse logic)	
Enable Loopback	Allows forwarding of packets from this VSI to local VSIs.	*VSI_SRCSWCTRL.ENABLELLOOPBACK*	
LAN Enable	Defines if the VSI is part of a VEB connected to the network.	*VSI_SRCSWCTRL.LANENABLE*	
Tag insertion and admission parameters			
Tag Accept Mode	Defines which tags to accept.	*VSI_TAR.ACCEPTTAGGED* and *VSI_TAR.ACCEPTUNTAGGED.*	
Port-based Tag	Define the VLAN/S-tag or other tags to insert into packet.	*VSI_TIR.PORT_TAG_ID*	
Port-based Tag Insert	Defines whether to use the tags previously described.	*VSI_L2TAGSTXVALID.PORTBASEDTAGS*	
Tag Strip Policy	Defines for each tag, if it should be left in packet, extracted to descriptor or removed.	*VSI_TSR*	
Queue mapping			
Queue base	The first queue allocated to this VSI.	*VSILAN_QBASE.VSIBASE*	
Number Of Receive Queues Per TC	Defines for each TC how many receive queues are allocated.	*VSIQF_TCREGION*	
Queue Mapping	Maps 16 virtual queues to physical queues.	*VSILAN_QTABLE* and *VSILAN_QBASE.VSIQTABLE_ENA*	

Table 38-188.VSI Context (Sheet 2 of 2)

Parameter	Description	Register.Field	Notes
PE Forwarding Enable	Enable TCP packets by the quad hash filter for the VSI.	VSIQF_CTL.PETCP_EN	
DCB control			
Enabled UPs	Defines the user priorities used by this VSI.	Implemented in scheduler.	

VSI Add and Remove Flows

Adding or removing a VSI can be done only by the PF. The following sections describes the flow a PF driver should use to add or remove a VSI.

VSI Init Flow

In order to initialize a VSI, software should use the *Add VSI* command to create the required VSI context. As part of this command, the queues allocated to the VSI should be specified. After this command runs the queues can be initiated.

VSI Disable Flow

VSIs can be disabled either individually or as part of their function. For example, when a VF is disabled, the VF reset stops the VSIs associated with this VF. There are cases where a PF might need to disable a single VSI. The use case for this flow is when a VSI is used as a VMDq interface to a VM that is disabled or moved to another system. In order to enable a quick disable of a VSI, the following flow should be used.

1. Software should stop scheduling packets to the transmit queues assigned to the VSI(s).
2. Stop the VSI traffic:
   a. For VMDQ VSIs, assert the VM reset using the *VSIGEN_RTRIG.VMSWR* bit and wait until the *VSIGEN_RSTAT.VMRD* bit clears. This step stops the traffic in all the queues associated with this VSI.
   b. For VFs, assert the VF reset using the *VPGEN_VFRTRIG.VFSWR* bit and wait until the *VPGEN_VFRSTAT.VFRD* bit clears. This step stops the traffic in all the queues associated with this VF.
3. Stop the relevant queues.
4. The PF polls the *Transactions Pending* flag of the VM, verifying that there are no transaction pending of the VM, as follows:
   - Set the VSI index in the *PFPCI_VMINDEX* register and then poll the *PFPCI_VMPEND* register.
5. Remove all filters pointing to this VSI/VF.
6. Clear the RSS table and key of the VSI using *Set RSS Key* and *Set RSS LUT* commands with zero values.
7. Send a *Delete Element* admin command with the VSI SEID (for each VSI associated to the VF). This step clears all VSI context values. Remove the VSI from the switch topology and remove it from the scheduler nodes.
8. Wait until the *Delete Element* command completes.
9. Clear the reset bit (*VSIGEN_RTRIG.VMSWR* or *VPGEN_VFRTRIG.VFSWR*).

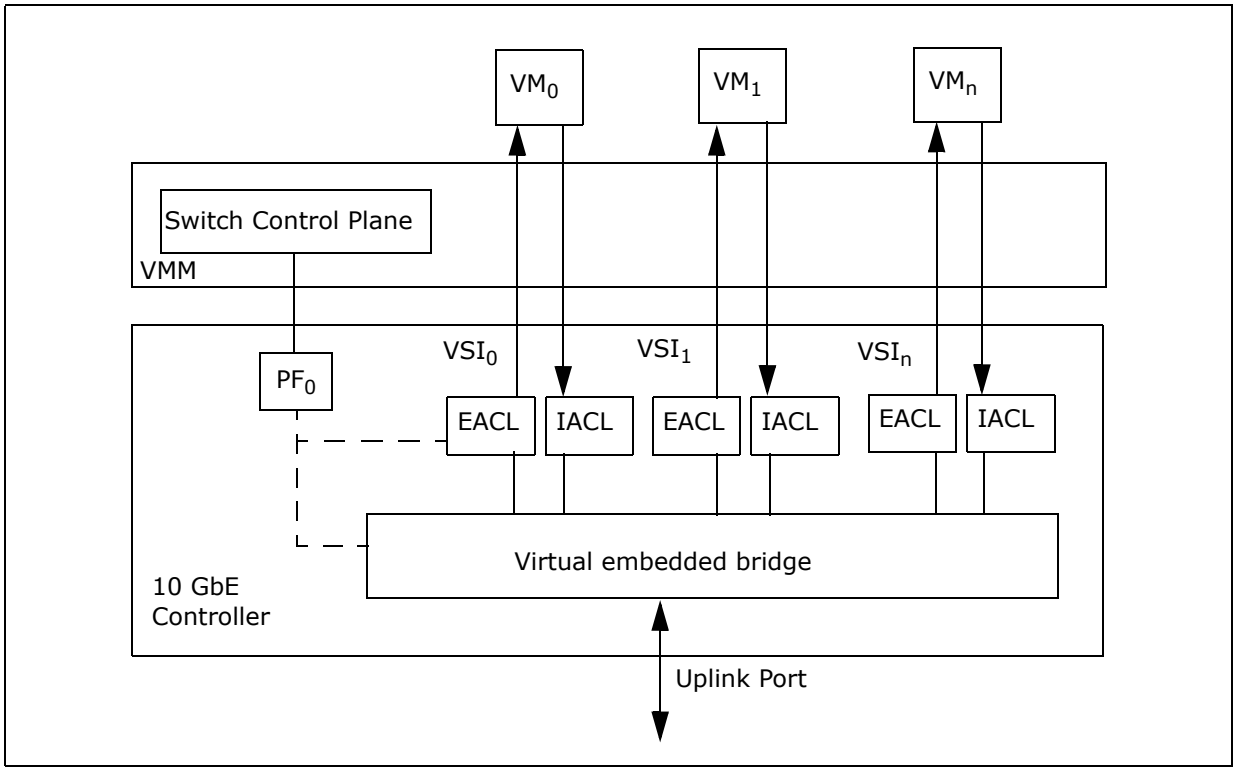
It's assumed that a VMDq VSI is not used for offloaded traffic (iWARP or UDA).

38.21.7.10 Advanced Switching Capabilities

38.21.7.11 Security Features

Security features implemented as part of the L2 tag handling such as port based VLAN and UP handling are described in Port-based VLAN section and Section 38.21.5, respectively.

Figure 38-47.Advanced Switching Capabilities



Anti-Spoofing

The 10 GbE controller supports anti-spoofing functionality. This is a security feature that ensures a VM is sending frames with MAC address and VLAN associated with that VSI. A malicious guest can impersonate as a trusted host by performing MAC address spoofing. If the anti-spoofing feature is turned on then only trusted MAC addresses are allowed on the ingress VSI.

MAC Anti-Spoofing

Trusted MAC address check is performed by performing a MAC address or VLAN, MAC lookup on the forwarding database. Since the forwarding database is populated by the VMM or control plane software when the VM/VSI is created, performing a MAC source address lookup on this database ensures that the VSI is transmitting with a source MAC assigned to that VSI.

MAC anti-spoofing is enabled by the *VSI.Enable MAC anti spoof* field in the VSI context.

VLAN Anti-Spoofing (Ingress Check)

VLAN ingress check is performed on incoming packets from VMs to ensure if the VSI is a member of the VLAN. This check is performed on the ingress VLAN membership table. VLAN ingress check can be enabled or disabled on each VSI. So a VM cannot transmit with a VLAN on which the VSI is not a member. This is applicable for VLAN aware guests. If a port or protocol-based VLAN is configured, a guest is not expected to send packets with 802.1Q tags then the admit untagged/priority tagged entry has to be set in port list.

VLAN ingress filtering is enabled by the *VSI.Enable VLAN anti spoof* field in the VSI context.

Port-Based VLAN

The 10 GbE controller supports a port-based VLAN feature by enabling any VSI to specify the default VLAN that it belongs to. The port-based VLAN association is done when a packet received on the VSI is untagged or priority tagged and protocol VLAN association is not done. Port-based VLANs map packets received on a given VSI with the corresponding port-based VLAN identifier called PVID. The PVID list should be programmed with the port VLAN IDs for each VSI.

The port VLAN list is provided, which is part of the VSI context. Table 38-189 lists the port VLAN parameters in the port VLAN list and the matching parameters in the Add VSI command.

Table 38-189.Port VLAN List and VSI Parameters

Port Based VLAN List	Add VSI Parameter	Notes
VSI Number	VSI number.	Returned by firmware in response.
PVID	PVID + default UP.	
Default UP		
Admit.1Q Tagged Only	VLAN driver insertion mode.	
Admit Untagged/Priority Tagged Only		
Admit All		
Ingress VLAN Check Enable	Enable VLAN anti-spoof.	
Insert PVID	Insert PVID.	
Expose VID And UP Of Received Packets	VLAN and UP expose mode (Rx).	
Expose UP Only Of Received Packets		
Do Not Expose VID Or UP Of Received Packets		
Ingress UP Translation Table	Ingress UP translation table.	see Transmit Functionality.
Egress UP Translation Table	Egress UP translation table.	see Receive Functionality.

The switch should insert the PVID to the packet if the *Insert PVID* parameter is set for the VSI and replace the UP bits according to the algorithm described in Transmit Functionality.

Once the VLAN ID association is made, the packet forwarding is performed using the VLAN, MAC forwarding table and VLAN membership table.

Private VLAN

The 10 GbE controller supports configuration of Private VLANs (PVLAN) in VEB mode. Private VLANs are used to provide Layer 2 level security by isolation of VMs within a VLAN (or IP subnet). Since VLAN defines a broadcast domain, all servers connected to the same VLAN can listen to broadcast packets. So a malicious VM can listen to neighboring virtual servers and launch direct attacks bypassing the security policies enforced by enterprise network switches. PVLAN provides Layer 2 security by creating sub-domains within the same primary VLAN without the need for an L3 router. These sub-domains are called secondary VLANs.

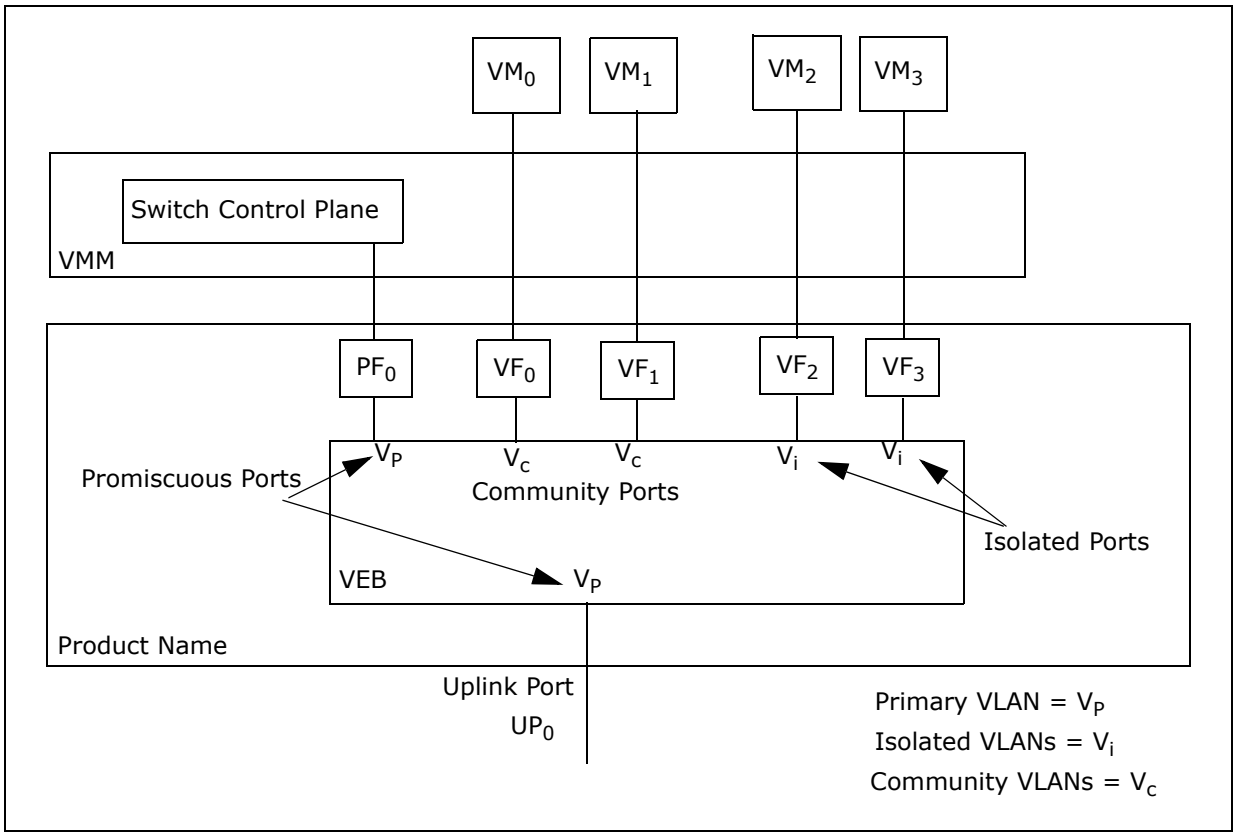
A virtual port in a private VLAN can be configured to one of three modes as follows:

- **Isolated Ports** — These cannot talk to any other virtual ports except to ports that are configured as promiscuous ports in the same primary VLAN. So a VM that only needs access to external network through uplink port has to be connected to isolated ports.
- **Community Ports** — The ports within a PVLAN community can talk to one another within the same community group (community VLAN) and also to the promiscuous ports in the same primary VLAN. There can be one or more PVLAN communities within the same primary VLAN. So a group of VMs that need to communicate with each other and also need external network access can be configured as members to one of PVLAN communities in the primary VLAN.
- **Promiscuous Ports** — These can talk to all of the ports in the same primary VLAN. Typically uplink ports and virtual ports that are connected to service VMs (for management purposes) can be designated as promiscuous ports.

Note: The meaning of promiscuous ports within the context of PVLAN is different from the generic promiscuous definition.

The following figure provides an illustration of PVLAN configuration.

Figure 38-48.Private VLAN Configuration



A primary VLAN V_P is configured on the VEB. Secondary VLANs, V_c is configured as community VLAN, and V_{i1}, V_{i2} are configured as isolated VLANs. The uplink port and the VEB management ports are configured as promiscuous ports. These ports communicate with all ports in the primary VLAN (V_p), including the community VLAN (V_p,V_c), and isolated VLAN (V_p,V_i).

The community VLAN ports (VF_0 and VF_1) can communicate only the between themselves and the uplink port UP_0 (and the management port PF_0). Isolated ports VF_2 and VF_3 can only communicate with uplink UP_0 (and the management port PF_0). The isolated ports cannot communicate with other isolated ports (VF_2 cannot communicate with VF_3) or to ports in community VLAN (VF_0 or VF_1).

VLAN pairing is used to represent secondary VLANs. There can be multiple community VLANs configured within a single primary VLAN (for example Vc1, Vc2, etc.); however, only one VLANID Vi is used to represent isolated VLANs.

The virtual port configuration and associated VLAN IDs are created as VLAN pairs as follows:

- Virtual ports (connected to VMs) are configured with VLAN pairs (primary VLAN, secondary VLAN).
- The distinction between the different roles is done by defining the ingress and egress membership of each VSI for the primary and secondary VLAN as described in Table 38-190.

Table 38-190.Private VLAN Membership

Port Type	Member of Primary Egress List	Member of Primary Ingress List	Member of Secondary Egress List	Member of Secondary Ingress List
Regular VLAN	Yes	Yes	N/A	N/A
Promiscuous	Yes	Yes	Yes	Yes
Community	Yes	Yes	Yes	No
Isolated	No	Yes	Yes	No

Isolated VLAN Configuration and Forwarding

Virtual port VF2, VF3 are configured as isolated ports to be members of both Vp and Vi. Frames from the network to virtual port (ingress) are arriving with primary VLAN ID Vp. However, frames from the virtual port to the network (egress) are always assigned a secondary VLAN ID Vi. Broadcast and multicast frames from the virtual port are not forwarded to other isolated ports in the isolated secondary VLAN Vi; they are only forwarded to uplinks (or ports configured as promiscuous ports in primary VLAN).

The forwarding database has one entry (Vp, MAC2) pairing for virtual port VF2. If a port-based VLAN is assigned to the port, then this is Vi. Otherwise, Vi is the only VLAN ID in the private VLAN to be part of the egress VLAN list for VF2. Packets arriving from a network carry a primary VLAN ID (Vp, MAC2). Only Vp will be part of the ingress VLAN list for this port.

If a unicast packet arrives from network that is designated as isolated VLAN, then it cannot be forwarded to isolated ports. Such frames should be dropped. If a multicast or broadcast packet arrives from the network with VLAN ID Vi, then it cannot be forwarded to isolated ports. It can only be forwarded to promiscuous ports, in this case it is management port PF0.

Community VLAN Configuration and Forwarding

Virtual port VF0, VF1 are configured as community ports to be members of both Vp and Vc. Frames from the network to virtual port (ingress) are arriving with either primary VLAN ID Vp or secondary VLAN ID Vc. However, frames from the virtual port to the network (egress) are always assigned a secondary VLAN ID Vc. Broadcast and multicast frames from the virtual port are only forwarded to other virtual ports that are members of the community VLAN Vc and to uplinks (or ports configured as promiscuous ports in primary VLAN).

The forwarding database has two entries Vp, MAC1 and Vc, MAC1 pairing for virtual port VF1. If a port-based VLAN is assigned to the port, then this is Vc. Otherwise, Vc is the only VLAN ID in the private VLAN to be part of the egress VLAN list for VF2. Ingress packets from a network towards VF2 uses either (Vp, MAC1) or (Vc, MAC2) for forwarding. Both Vp and Vc are part of the ingress VLAN list for this port.

If a unicast packet arrives from network that is designated as community VLAN Vc then it can be forwarded only to community ports that are members to the same community VLAN Vc. If a multicast or broadcast packet arrives from the network with VLAN ID Vc then they can be forwarded to other community ports in the same community VLAN Vc and to promiscuous ports, in this case it is management port PF0.

Primary VLAN, Promiscuous Port Configuration and Forwarding

A primary VLAN Vp is created with all the community ports, isolated ports and promiscuous ports as members. Unicast packets arriving from the network with primary VLAN Vp are forwarded based on (Vp, MACx). Egress check is performed to find out the member ports. Multicast or broadcast packets are forwarded to based on (Vp, Multicast MACx) and forwarded to member ports. Broadcast is forwarded to all members of primary VLAN.

Ports that are configured as promiscuous ports are typically uplink ports (UP0) and switch management ports (PF0). The promiscuous ports are members of primary VLAN Vc and all secondary VLANs, Vc, Vi. So the promiscuous ports can receive broadcast packets from any of the secondary VLANs.

Note: In order to assure proper operation of private VLANs both internal VEB and external access switch need to be configured with the same private VLAN designations.

Local VLANs

A VLAN can be defined as local. In this case, packets sent with this VLAN ID is not forwarded to the network and packets from the network with this VLAN are dropped.

Locality is defined at the VEB level.

A VLAN can be defined as local using the *Add VLAN* admin command with the *LocalVLAN* attribute set.

User Priority Bits (802.1p) Handling

The *UP* bits are used to differentiate between multiple classes of traffic. The 10 GbE controller supports multiple use cases related to the handling of the UP bits:

1. An operating system that is not DCB/traffic type aware and uses a single TCID.
2. An operation system that is traffic type aware, but not DCB aware. It can distribute the traffic to different queues, but cannot tag them with the right UP and iWARP. The operating system is not aware but the software device driver knows the difference and hence can assign the different types of traffic to different flows. There is no DCBX agent so the software device driver does not know the UP or TC for flows.
3. An untrusted operating system that might set *UP* bits of TCs it is not allowed for use.
4. An operating system that can use a single queue per TCID, but would like to use more UPs in order to gain from the differentiated QoS in the network.

The following sections describe the handling of *UP* bits in transmit and receive to support the previous use cases.

Transmit Functionality

Each transmit packet is assigned a UP, either by the software device driver or as part of the port VLAN table.

This UP is translated using two different translation vectors.

The first vector is specific to a VSI and is part of the port VLAN table (*transmit UP translation table - VSI_TUPR*). This vector reflects the mapping of the user priorities as seen by the operating system to the user priorities as seen by the network.

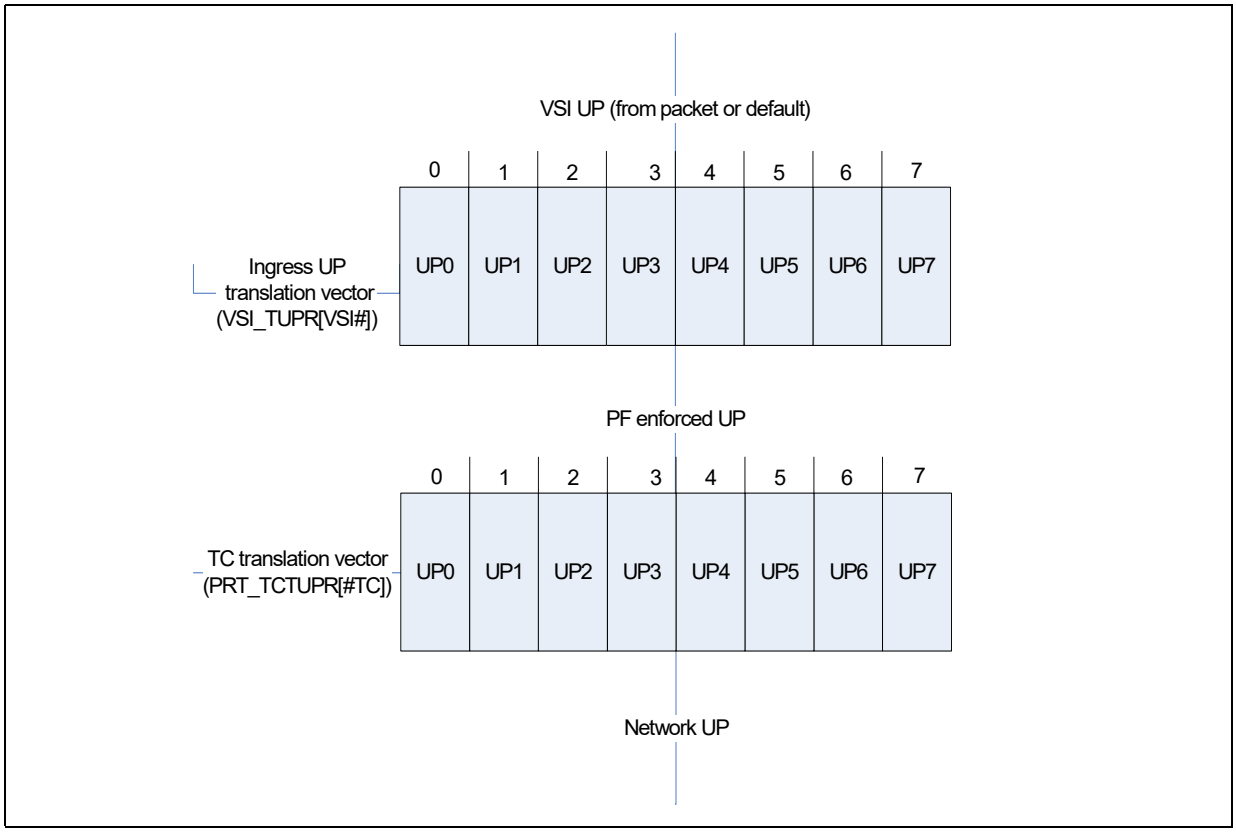
The second vector is specific to a TCID in a physical port. It translates the UP received from the previous translation to a UP matching the TCID to which the packet is associated. These vectors can be configured through the *PRT_TCTUPR* registers.

Note: There is no drop of packets due to incorrect 802.1p tagging. If an incorrect tag is requested the tag is replaced. Thus, this functionality supports both UP anti-spoofing and port based UP.

Untagged packets are kept as is. No translation is done on them.

The following figure shows the algorithm.

Figure 38-49.Example: Non-VLAN Aware, Non-DCB Aware Operating System



Assume a VF that queues LAN traffic and iSCSI traffic to different queues, but is not VLAN aware. In this case, the packet is sent by the software device driver with no VLAN and the default UP set in the port-based VLAN (such as 0) is the initial UP of the packet. In this case, the *Ingress UP translation vector* (*VSI_TUPR*) is not relevant and might be for example {0,1,2,3,4,5,6,7} or {0,0,0,0,0,0,0,0}. The *TC translation vector* (*PRT_TCTUPR)* for LAN might be {0,1,2,0,0,0,0,0} and the TC table for storage might be {4,4,4,4,5,4,4,4}. So a LAN packet goes out with a UP of 0, and a storage packet goes out with a UP of 4.

Example: Non DCBx tagging operating system

Assume a VF that tags LAN high priority traffic with a UP of 4 and LAN low priority traffic with a UP of 3 and the storage traffic as UP 7. In this case, the VF translation table might be {0,1,2,0,2,4,4} and the TC tables as previously described. The low priority LAN goes out with a UP of 0, the high priority LAN with a UP of 2 and storage with a UP of 4.

The tag on which the transmit UP translation is done is identified by setting the *INNERUP* bit in the matching GL_SWT_L2TAGCTRL register.

The default mapping of both tables is identity mapping. For example, mapping a UP to itself. If UP translation is not needed these mappings should not be changed.

Transmit Outer Tag User Priority

The transmit outer tag user priority can be handled in three different ways:

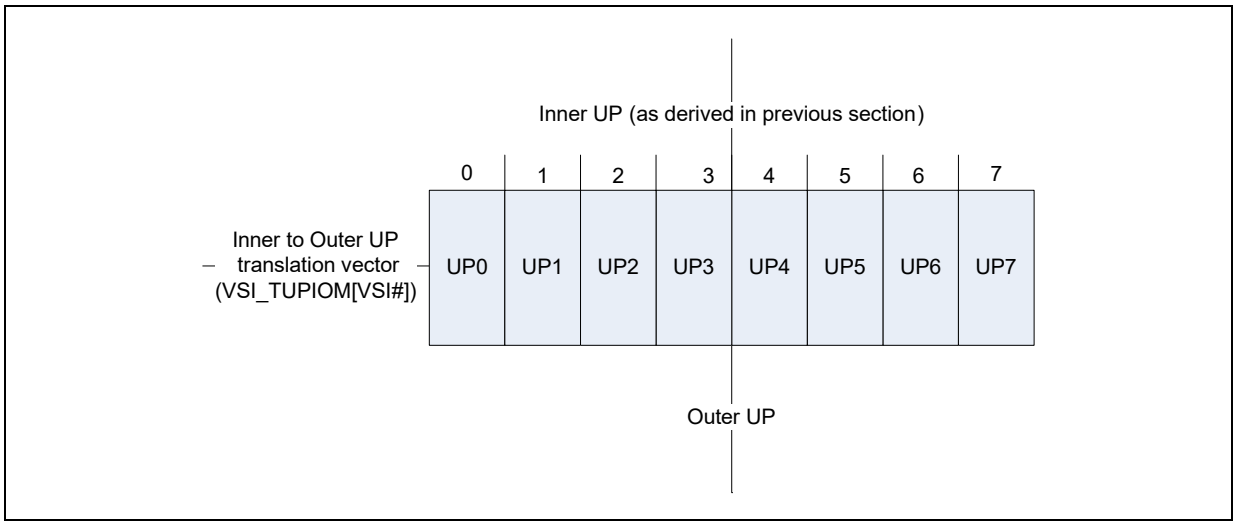
1. If the outer tag is received in the packet or the descriptor (such as cascaded S-comp), the UP should be part of the packet sent by the software device driver or inserted from the descriptor.
2. If the outer tag is inserted by hardware then the UP might be either:
   a. The UP defined as part of the inserted tag in the VSI_TIR register.
   b. A translation of the regular VLAN UP as defined in the *VSI_TUPIOM* register.

If the outer tag is defined as *OuterUP* in the GL_SWT_L2TAGCTRL register, then the UP is based on the first tag for which the *InnerUP* field is set in the GL_SWT_L2TAGCTRL register; usually the VLAN tag (option 2.b); otherwise, it is based on the default (option 2.a).

The tag to which translation is applied can be either S-tag or external VLAN.

The following figure shows the translation process.

Figure 38-50.Tx Inner-to-outer UP Translation



Receive Functionality

VLAN UP Translation

The priority bits of inner VLAN tag in received packets might be remapped using the *Ingress UP translation table* in the *Add VSI* command. This means the operating system might get packets with 802.1p priority bits reflecting the local configuration and not the link configuration. By default (ingress UP translation section is not valid) the translation is one-to-one, meaning that the received UP is not translated.

VLAN UP Exposure to Software Device Driver

The exposure of the received UP to operating systems is defined by the awareness of the operating system controlling the VSI as follow:

1. For a monolithic operating system, VMM or for a guest operating system that is VLAN aware, the UP is exposed as part of the VLAN as requested by the software device driver (either in the packet or in the descriptor).
2. If the guest operating system is DCB aware but not VLAN aware, it gets the priority bits as part of the receive descriptor in the VLAN tag field. The VLAN ID is zero, but the priority bits are valid.
3. A guest operating system that is not VLAN aware and not DCB aware does not get the user priority bits (UP) at all.

The configuration of the UP removal is part of the port VLAN configuration described in Port-Based VLAN and in the Add VSI command VLAN handling section. See Add VSI (0x0210).

38.21.7.11.1 Switch Diagnostics

This section describes the mirroring and loopback features.

Mirroring

The 10 GbE controller supports 64 mirroring rules. Each rule can be coupled to any of the VEB or port aggregator in the device. The destination port of each rule (the mirror port) must be connected to the switching element in which the rule is applied. Packets that match any of the mirroring rules assigned to this element are forwarded to the VSI defined in the mirror rule. The setup of a mirror VSI is described in Mirror VSI Setup.

In order to differentiate packets received according to the different rules, the matched rule is indicated in the Mirror Rule ID *(MIRR)* field in the receive descriptor.

Note: The switching algorithm enables mirroring-to-mirror ports even if the loopback is disabled for this port. The controlling software needs to make sure mirror ports are assigned only to ports that can receive local traffic.

Each rule can is defined by the following parameters:

1. A rule ID — A value in the 0..63 range reflected in the *MIRR* field in the receive descriptor.
2. A mirror port to which packets matching the mirroring conditions are sent.
3. A set of mirrored ingress VSIs — All packets received by these VSIs are also sent to the mirror port.
4. A set of mirrored egress VSIs - All packets sent by these VSIs are also sent to the mirror port.

Additional rules can be defined to create ingress VLAN mirroring — When such a rule is created, all traffic received in a set of given VLANs by any of the virtual ports either from the uplink or from local VMs within a given VEB are forwarded to a mirror port.

Note: Packets forwarded by ingress VLAN mirroring are not identified by the *MIRR* field and the *UMBCAST* field as a mirror packet. The matched mirror rules can be inferred from the VLAN of the packet. Mirroring according to ingress VLAN mirroring of transmit packets might add an additional copy of the packet to the default VSI of the VEB or the port.

Mirroring rules can be applied using the following admin commands:

- Add Mirror rule (Add Mirror Rule (0x0260)).
- Remove Mirror rule (Delete Mirror Rule (0x0261)).

Note: A mirror port cannot be defined in the ingress mirroring list of another mirroring rules.

A single rule can be either ingress VSI, egress VSI or VLAN and cannot include multiple types of rule.

Ingress or egress VSI traffic cannot be mirrored to multiple mirrored ports.

A VSI cannot be part of more than a single port mirror rule (cannot be mirrored twice). This limitation does not include VLAN mirroring.

A control VSI that uses the switch override field (SWTCH) in transmitted packets descriptor cannot be part of an ingress mirror rule.

Ingress Mirroring Flow

Ingress mirroring acts on all the packets that are sent from a given VSI, no matter if the packet is sent to the network or to a local VSI and does not depend on the *Allow Loopback* flag in the VSI context.

Packets dropped due to anti-spoofing are also mirrored.

Egress Mirroring Flow

Egress mirroring acts on all the packets that are received by a given VSI no matter if the packet was received from the network or from a local VLAN.

Mirror VSI Setup

A VSI used to receive mirror traffic should be dedicated to this type of traffic and thus should not be used to send or receive other traffic.

VSI Loopback

Some systems require a local loopback capability at the switch level allowing to receive all the packets sent from a VSI back to the same VSI.

This capability is achieved by using the following steps:

1. When creating the VSI using the *Add VSI* command, set the *Allow destination override* bit.
2. When sending a packet, set in the descriptor the *SWTCH* field to 11b (target VSI) and set the *Segmentation Parameters.VSI* field to the value of this VSI.

Packets sent using descriptors with the attributes previously mentioned are sent back to a receive queue of the same VSI.

38.21.7.11.2 Cascaded Port Virtualizer Offloads

The 10 GbE controller supports VSIs used as cascaded port virtualizers as described in Cascaded VEB and Port Virtualizers. Such a VSI provides some special offloads to support this mode.

Transmit Offloads

S-tag insertion — As this VSI supports multiple S-tags, the S-tag cannot be inserted by the 10 GbE controller using the port-based tagging mechanism. The software device driver might indicate the S-tag to insert in the *L2TAG2* field in the transmit descriptor and set the *IL2TAG2* bit. Hardware then inserts the right tag to the packet.

The ability of a VSI to control the S-tag from the descriptor or from the packet is set by the *Accept tag from host* in the *Add VSI* command.

Note: If a packet is sent by the software device driver with the S-tag as part of the packet, it must also include the VLAN.

If an S-tag needs to be inserted, a 32-byte transmit descriptors must be used.

Receive Offloads

- S-tag extraction — This VSI supports multiple S-tags so the software device driver must receive the S-tag in order to process the packet. These tags can be extracted from the packet and stored in the receive descriptor in the *L2TAG2 (1st)* field. The presence of these tags is indicated by the *L2TAG2P* bit. This offload is activated via the *Report S-tag* field in the *Add VSI* command.

Note: This capability is orthogonal to the VLAN tag extraction and might be requested with or without VLAN extraction.

- Source Pruning — This capability should be disabled in a cascaded S-comp by clearing the *Prune based on ingress E-PID* enable field in the VSI context.
- VMDq1 Queueing — A cascaded port virtualizer might request to queue packets with different tag values to different queues, similar to the VMDq1 mode supported for cascaded VEBs. In this mode, packets with an unrecognized S-tag is forwarded to the default queue of the VSI.

38.21.7.11.3 DCB and Rate Limit Support

Each VSI can be associated with a scheduling element and with a set of user priorities and TCs. Each user priority is represented by a set of queue pairs and a scheduler element. If multiple UPs are associated with the same TC, a scheduler element might be added later to group those UPs into a single TC. The VSI scheduling element can be used to define an SLA for this VSI. The UP and TC scheduling elements are used to set an ETS policy for this VSI. Each S-channel created is also associated with a scheduler element allowing a different SLA for each channel.

Assigning scheduling elements is done internally when an *Add VSI* or an *Add VEB* admin command is received. The dis-association is done when a *Delete Element* admin command is received.

The TCs created are set according to the *Enabled UPs* field in the *Add VSI* command. The handles to the TC rate limiters are returned as part of the *Add VSI* response. The addition of TC scheduler elements can be done after the VSI exists using the *Configure VSI Bandwidth per Traffic Class* admin command).

In order to control the rate limiter behavior, the scheduler admin commands should be used. When accessing a switching element (VSI or S-channel) scheduler element, the SEID is used as the handle to the node. When accessing a TC the handles returned in the *Add VSI* response are used.

38.21.7.12 Flows

This section provides a summary of the processing across all the switching elements in Tx and Rx.

38.21.7.12.1 Tx Flow

The following steps are taken once a packet is accepted for transmission by the switching engine. This flow assumes any state offload needed had been applied to the packet:

1. **Anti-spoofing —** If enabled in the Add VSI command VSI sending the packet is part of those addresses destination, the packet is allowed to proceed.
2. **VLAN insertion —** If needed, insert the VLAN tag to the packet, either as a host request or as a port-based VLAN.
3. **S-tag insertion —** If the VEB or the function is connected to an S-comp, add the S-tag of the channel to the packet.
4. **UP replacement —** Replace the *UP* bits.
5. **VEB filtering —** If the port/queue that sends the packet is connected to a VEB, the algorithm described in Section 38.21.7.13.4 is applied to create a list of destination ports. Otherwise, the list of destination ports is simply the network. The VEB filtering for Tx packets includes the following steps:
   a. Defining the list of VSIs that are candidate for reception of the packet. These VSIs are defined according to the filters set using the *Add MAC, VLAN pair*, *Add Cloud filters* and *Add EtherType filters* commands (Section 38.21.7.18.8).
   b. Add an EMP-owned VSI that should receive the packet. This includes forwarding to the MC channel using decision filters and control packets.
   c. Potentially removing from this list the VSIs that do not belong to the VLAN group as defined by the *Add VLAN* command. See Add VLAN (0x0252) for more detail.
   d. Potentially removing from this list the VSI that sent the packet. This removal can be based on comparison of the SA of the packet to the MAC table.
   e. Add ingress mirror VSI as defined by the *Add Mirror Rules* admin command.
6. For a copy of the packet sent to the sideband interface follow the steps in step 8. For copies of the packet sent to local functions follow the steps in step 9
7. **Processing of packets sent to the network:**
   a. **MAC processing** — If it applies, insert a CRC on the packet.
   b. Forward to the network.
8. **Processing of packets sent to the sideband interface**:
   a. Forward the packet to the sideband interface.
9. **Processing of packets sent to local functions**:
   a. **Security and MAC processing** — If it applies, encrypt or decrypt the packet using IPSec.
   b. **Forward to function** — Forward the packet to the selected function for further processing.

38.21.7.12.2 Rx Flow

The following steps are taken for each packet received from the network by the switching engine:

1. **MAC processing** — Check the CRC and other error conditions. If *Pass Bad Packets* is set (PRT_SBPVSI.SBP), add the bad frames VSI (*PRT_SBPVSI.BAD_FRAMES_VSI*) to the VSI list. Identify flow control packets and priority flow control packets and react to them.
2. **VEB/port aggregator/L2 filter** — According to the configured switching element connected to the S-channel/physical port, apply the algorithms defined in Section 38.21.7.13.2, or Section 38.21.7.13.4 to define a list of destination ports. This includes the following steps:
   a. Defining the list of VSIs that are a candidate for receiving the packet. These VSIs are defined according to the filters set using the *Add MAC, VLAN pair*, *Add Cloud filters* and *Add EtherType filters* commands (Section 38.21.7.18.8).
   b. Add an EMP-owned VSI that should receive the packet. This includes forwarding to the MC channel using decision filters and control packets.
   c. Potentially removing from this list the VSIs that do not belong to the VLAN group as defined by the *Add VLAN* command (Add VLAN (0x0252)).
   d. Potentially removing from this list the VSI that sent the packet. This removal can be based on comparison of the SA of the packet to the MAC table.
   e. Add egress mirror VSIs as defined by the *Add Mirror Rules* admin command.
3. The resulting list of destination ports can include local ports or sideband interface.
4. For a copy of the packet sent to the sideband interface follow the steps in step 6. For copies of the packet sent to local functions follow the steps in step 8.
5. **Processing of packets sent to local functions**:
   a. **S-tag and VLAN removal** — If it applies, remove the S-tag and VLAN tag from the packet and optionally report them in the Rx descriptor.
   b. **UP replacement** — Replace the *UP* bits as described in Receive Functionality.
   c. **Forward to function** — Forward the packet to the selected function for further processing including queuing and iWARP offload.
6. **Processing of packets sent to the sideband interface or to the EMP**:
   a. Forward the packet to the sideband interface or to the EMP. The VSI context defines the destination within the EMP VSI (one of the sideband interfaces or the EMP) either using the regular switch or the packet classifier.

38.21.7.13 Switching Algorithms

38.21.7.13.1 LAN Port Extender Algorithm

This algorithm translates a destination MAC address to a switch ID. After this algorithm is applied, the regular per switch forwarding algorithm should be applied to each of the switch IDs. The algorithm is relevant only to receive traffic.

```
// Global parameters

Port.DefaultStag; Can be Null or an S-tag.

// Define Lookup tables

DA Table: Array of {DA ,LAN , Switch ID}

EComp_function(Packet) {
```

```
// Variables

STag = Port.DefaultStag

SourceStagFlag = FALSE // SourceStagFlag is not used by this algorithm.

//Define packet parameters

DA = Packet.DA: the DA of the packet.

Source Port // The port from where the packet was received.

// Switching algorithm

// Selection of S-tag using DA

If ({DA, source port} match entry e in DA table): DSTag = e.S-tag;

Return DStag, SourceStagFlag;

}
```

38.21.7.13.2 S-comp Forwarding Algorithm

The following algorithm refers only to S-comp receive traffic. As previously described, transmit traffic is always sent to the LAN apart from the iWARP loopback case.

Packets with unmatched S-tag are forwarded to a default VSI of the port, if enabled.

Packets forwarded to the default VSI cannot be subject to L2 filtering.

This algorithm defines a switch ID that identify the S-channel to which the packet should be forwarded. The switch ID is used to identify a potential VEB connected to this S-channel. The forwarding inside this channel is defined according to the L2 forwarding algorithm (Section 38.21.7.13.4) or VEB/VEPA algorithm (Section 38.21.7.13.3).

If no VEB is defined, then the VSI connected to the S-channel should be put in promiscuous mode using the Set VSI Promiscuous Modes command in order to receive all the traffic with the given S-tag.

If an S-comp is not defined on the port, then all the packets with an S-tag are dropped.

```
// Global parameters

Port.DefaultVSI; Can be Null or a VSI

Port.Default_Switch_ID; // The switch ID to use for untagged.

DefaultVSIValid - per port// If set, packets with an S-tag that do not match any of

these programmed S-tags or an untagged packet that do not match the expected

ethertypes will be sent to the default VSI, otherwise the packet is dropped.

// Define Lookup tables

StagTable: Array of {STAG}

// Control Port forwarding rules

SComp_function(Packet) {

// Variables

Dest_VSI - Null; // Destination VSI in case of default VSI;
```

```
Switch_ID = Null;
//Define packet parameters
STag = Packet.Stag: the S-Tag of the packet.
Untagged // True if packet has no S-tag.
// Switching algortihm
// control port forwarding
// Note - the usual case of a control VSI is to receive LLDP packets or other
packets based on special Ethertypes. However, the device allows any L2 forwarding
to the control VSI assuming the right forwarding rule is added. See
Section 10.21.4.8.4 for details of the available rules.
If (Untagged) { Switch_ID = Port.Default_Switch_ID;}
// S-tag forwarding
If(Stag match entry e in StagTable) {
Switch_ID = S-tag
} else if (DefaultVSIValid)
Dest_VSI = Port.DefaultVSI
else drop packet.
Return Switch_ID, Dest_VSI;
}
```

38.21.7.13.3 L2 Filtering Algorithm

The following pseudo code describes the algorithm used to determine if a packet passes the L2 filtering element.

```
// Global parameters
// Define Lookup tables
MacVlan Table: Array of {MAC (MAC Address), VLAN (VLAN tag)}
Mac Table: Array of {MAC (MAC Address)}
VLAN table: Array of {VLAN (VLAN tag)}
HashMacVlan Table: Array of {HashMAC (Hash Values), VLAN (VLAN tag), AddressType}
HashMac Table: Array of {HashMAC (Hash Values) , AddressType}
EtherType Table : Array of {Etype (Ethertypes Values)}
MacEtherType Table: Array of {MAC (MAC Address), Etype (Ethertype value)}
// Define Virtual ports modes
Port.PUE // Promiscuous Unicast Enable
Port.PME // Promiscuous Multicast Enable
Port.BAM // Broadcast Enable
```

```
Port.VPE // Promiscuous VLAN enable

Port.MaxSize: Max Packet size

L2_function(Packet)

// Variables

MFilter: = False // MAC Filtering

VFilter: = False // VLAN Filtering

EFilter: = False // Exclusive Filtering

NEPMFilter = False // Non Exclusive Perfect MAC Filtering

NEPFilter = False // Non Exclusive Perfect Filtering

NEIPMFilter = False // Non Exclusive Imperfect MAC Filtering

NEIPFilter = False // Non Exclusive Imperfect Filtering

Pass = False // Final decision.

//Define packet parameters

DA = Packet.DA //Destination Address of the packet

VID = Packet.VLAN ID // Vlan tag of the packet

Etype = Packet.Ethertype // Ethertype of the packet

AddressType //Type of address of the DA. Can be Unicast, Multicast or Broadcast.

HDA = HashFunction(DA).

PSize: Packet size. This do not include any tag or other header removed by previous
stages or to be added by following stages.

// Exclusive Filters

For Each entry e in MacVlan Table

    If (DA == e.MAC and (VID == e.VLAN or (VID == NULL and e.VLAN == 0)) MFilter =
True;

For Each entry e in Mac Table

    If (DA == e.MAC) MFilter = True;

For Each entry e in Ethertype Table

    If (Etype == e.Etype) MFilter = True;

For Each entry e in MacEthertype Table

    If (DA == e.MAC and Etype == e.Etype) MFilter = True;

For Each entry e in Vlan Table

    If (VID == e.VLAN or (VID == NULL and e.VLAN == 0)) VFilter = True;

// VLAN filters are ANDed with the previous filters unless promiscuous VLAN is
enabled

EFilter = MFilter and (VFilter or p.VPE} and AddressType == Unicast;

// Non Exclusive Perfect Filters
```

```
If (AddressType == Unicast and p.PUE == 1) NEPMFilter = True;

If (AddressType == Multicast and (p.PME == 1 or MFilter)) NEPMFilter = True;

If (AddressType == Broadcast and (p.BAM == 1 or MFilter)) NEPMFilter = True;

// VLAN filters are ANDed with the previous filters unless promiscuous VLAN is
enabled

NEPFilter = NEPMFilter and (VFilter or p.VPE};
```

// Non Exclusive Imperfect Filters

```
For Each entry e in HashMACVlan Table

    If (HDA == e.HashMAC and (VID == e.VLAN or (VID == NULL and e.VLAN == 0)) and
AddressType == e.AddressType) NEIPMFilter = True;

For Each entry e in HashMAC Table

    If (HDA == e.HashMAC and AddressType == e.AddressType) NEIPMFilter = True;

// VLAN filters are ANDed with the previous filters unless promiscuous VLAN is
enabled

NIEPFilter = NIEPMFilter and (VFilter or p.VPE};

// Packet size filtering is done at the queue level, so it is not part of the switch
algorithm

Pass = (EFilter or NEPFilter or NEIPFilter)

Return Pass;

}

HashFunction(MAC){

Hash = MAC[6:0] // Use the 7 least significant bits of the MAC address (last bits
on the wire).

Return Hash

}

LocaliWARP // The packet is part of a local iWARP connection and should be sent to
the source VSI.If this bit is set the packet is sent only to the origin VSI and any
anti spoof check may be skipped. This bit is set only in offloaded packets (either
by the SWPE bit in descriptor or by the PE).

DoNotLoopback // An indication that may come from the Protocol Engine, indicating
that a packet should not be loopback.

if ((Destination == UpLink and SVSI.AllowOverride) {Specific_Port = LAN};

except for local iWARP packets and
```

38.21.7.13.4 Cloud VEB Algorithm

// Global parameters

// Define Lookup tables

// L2 forwarding tables

MacVlan Table: Array of {MAC , VLAN , DestinationVSIList };

MAC Table: Array of {MAC , DestinationVSIList }

VLAN table: Array of {VLAN, IngressVSIList, EgressVSIList, Local}

HashMacVlan Table: Array of {HashMAC (Hash Values), VLAN, AddressType (unicast/ Multicast), DestinationVSIList }

HashMac Table: Array of {HashMAC (Hash Values), AddressType (unicast/multicast), DestinationVSIList }

// Control packets forwarding tables

MacEthertype Table : Array of {MAC , Etype , Source, DestinationVSIList, Drop}

Ethertype Table : Array of {Etype , Source, DestinationVSIList, Drop}

// Cloud Forwarding tables

OuterIP: Array of {Outer IP, DestinationVSIList}; // Outer IP is valid only for encapsulated packets (Filter type = 0x1).

InnerMAC_InnerVLAN: Array of {Inner MAC, InnerVLAN, DestinationVSIList}; // This table can be matched by packets with NVGRE, VXLAN or Geneve tunneling (Filter type = 0x3).

InnerMAC_InnerVLAN_Key: Array of {Inner MAC, InnerVLAN, Key, EncapsulationType, DestinationVSIList}; // This table can be matched by packets with NVGRE, VXLAN or Geneve tunneling (Filter type = 0x4).

InnerMAC_Key: Array of {Inner MAC, Key, EncapsulationType, DestinationVSIList}; // This table can be matched by packets with NVGRE, VXLAN or Geneve tunneling (Filter type = 0x6).

InnerMAC: Array of {Inner MAC, DestinationVSIList}; // This table can be matched by packets with NVGRE, VXLAN or Geneve tunneling (Filter type = 0xA).

InnerMAC_OuterMAC_Key: Array of {Inner MAC, OuterMAC, Key, EncapsulationType, DestinationVSIList}; // This table can be matched by packets with NVGRE, VXLAN or Geneve tunneling (Filter type = 0xB).

InnerIP: Array of {Inner IP, DestinationVSIList}; // Inner IP is valid for all packets. In non encapsulated packets, it is the sole IP (Filter type = 0xC).

Promiscuous_List: Array of {VLAN, Address_type DestinationVSIList}; // Promiscuous modes per VLAN

// Define VSI modes

VSI[i].PUE: Promiscuous Unicast Enable per VSI (promiscuous VLAN only)// Target VSI

VSI[i].PME: Promiscuous Multicast Enable per VSI (promiscuous VLAN only)// Target VSI

VSI[i].BAM: Broadcast Enable per VSI (promiscuous VLAN only)// Target VSI

VSI[i].VLANPruneEnable // Target VSI - Should be cleared in case of promiscuous enable

VSI[i].AllowLoopback: Loopback Enable per VSI // Source VSI

```
VSI[i].MACPruneEnable: // Defines if the source VSI (identified either by SA,VLAN
or by SA or source VSI) should be removed from the list of destination VSI.

VSI[i].EnableMACAntiSpoof: // Source VSI

VSI[i].EnableVLANAntiSpoof: // Source VSI

VSI[i].LANEnable // Source VSI - should be set in each VSI tied to a switch that is
connected to the LAN. (VSI_SRCSWCTRL.LANENABLE). This bit is cleared if the VSI is
added to a floating VEB.

VSI[i].AllowOverride // Allows override of the destination - usually set for
control VSIs.

// Switch generic parameters

DefaultVSI; Can be Null or one of the VSIs. // Defined as part of the S-tag table.

ControlVSI; Can be one of the VSIs or the embedded controller.

// Mirroring rules

Int n_mirror: Number of mirror rules

enum {ALL_EGRESS, ALL_INGRESS, VSI_EGRESS, VSI_INGRESS, VLAN_MIRROR} RULE_TYPE;

Mirror rule[1 .. n_mirror] = {RULE_TYPE type, IngressVSIList (List of VSI),
EgressVSIList (List of VSI), VLANList ( List of VID), MirrorPort};

Dport Switch_function(SVSI : Source VSI, Packet, FromLAN, FromHost)

// Variables

CDport: List of VSI = {} // Cloud Filtering - empty list at startup.

PFDPort: List of VSI = {} // Perfect Filtering - empty list at startup.

IFDport: List of VSI = {} // Imperfect Filtering - empty list at startup.

PMDport: List of VSI = {} // Promiscuous Filtering - empty list at startup.

OFDPort: List of VSI = {} // Override VSI Filtering - empty list at startup.

VFDPort: List of VSI = {} // VLAN membership VSI Filtering - empty list at startup.

MNGPort: List of VSI = {MDEF filtering result}

MNGOnly: Bool = {MNGonly result}

DPort: List of VSI = {} // Final List - empty list at startup. Can include also the
LAN port for Tx packets.

PassAntiSpoof = False;

PassMACAntiSpoof = False;

PassVLANAntiSpoof = False;

//Define packet parameters

DA = Packet.DA: Destination Address of the packet

SA = Packet.SA: Source Address of the packet

VID = Packet.VLAN ID: Vlan tag of the packet - equal to zero if untagged packet.

Etype = Packet.Ethertype: Ethertype of the packet

AddressType: Type of address of the DA. Can be Unicast, Multicast or Broadcast.
```

```
HDA = HashFunction(DA).

OuterIP = Packet.OuterIP // Relevant only for encapsulated packets.

InnerIP = Packet.InnerIP // Inner IP is valid for all packets. In non encapsulated
packets, it is the sole IP.

InnerMAC = Packet.InnerMAC;

InnerVLAN = Packet.InnerVLAN;

Key = Packet.NVGRE TNI key or Packet.VXLAN VNI key or Packet.Geneve VNI Key;

EncapsulationType = NVGRE, VXLAN, Geneve;

Source VSI = VSI that sent the packet (valid only for Tx packets);

PSize: Packet size. This do not include any tag or other header removed by previous
stages or to be added by following stages.

LocalVLAN = False: Define if the packet is part of a local VLAN. See below for
calculation.

Destination: // Can be either uplink (send only to LAN), local (switch decides on
destination, but do not send to LAN), switchBased (regular switching algorithm) or
a TargetVSI.  This is a descriptor bit for packets received from a control port
allowing override of the switching decision. This may be useful when sending switch
generated packets like DCBX packets. Relevant only for transmit packets

Source = Tx or Rx // defines if this is a packet sent by the 10 GbE controller or
received by the 10 GbE controller.

DoNotLoopback // An indication that may come from the Protocol Engine, indicating
that a packet should not be loopback.

Bool Promiscuous_Joins_Exact;//Set by the Set Switch Configuration AQC.

// Combine override destination option.

// This logic creates the following indications to be used later

Specific_Port = NULL; // A specific port defined as the sole destination of the
packet.

SwitchToLAN = TRUE; // Allow forwarding to LAN

SwitchToLocal = TRUE; // Allow forwarding to local VSI.

else if (Destination == TargetVSI and SVSI.AllowOverride) {Specific_Port = Target
VSI from Descriptor};

if ((Destination == UpLink and SVSI.AllowOverride) or DoNotLoopback or
SVSI.ALLOWLOOPBACK == FALSE) SwitchToLocal = FALSE;

if ((Destination == UpLink and SVSI.AllowOverride) {Specific_Port = LAN};

if ((Destination == Local and SVSI.AllowOverride) or SVSI.LANDisable == TRUE)
SwitchToLAN = FALSE;

If (FromLan) SPort = LAN else SPort = SVSI.

// Anti Spoofing

// Local VLAN calculation

For each v in VLAN Table {if (v.VLAN == VID and v.LocalVLAN == True) LocalVLAN =
TRUE}

// Need to check if the packet can enter the switch before applying all the rules
```

```
If (SVSI.EnableMACAntiSpoof == True and FromHost) {

// Only perfect match filters are used for anti spoofing checks

        For Each entry e in MacVlan Table

        if (SA == e.MAC and (VID == e.VLAN or (VID == NULL and e.VLAN == 0)) and
SVSI in e.DPortList ) PassMACAntiSpoof = True

    For Each entry e in Mac Table

        if (SA == e.MAC and SPort in e.DestinationVSIList ) PassMACAntiSpoof = True

    } else PassMACAntiSpoof = True;

// The step below (VLAN anti spoof) is really ingress VLAN filtering

If (SPort.EnableVlanAntiSpoof == True and FromHost) {

    For Each entry e in  VLAN Table

    // Note - an untagged packet will be compared with an entry of the table with
VLAN = 0.

        if (VID == e.VLAN and SVSI in e.IngressVSIList) PassVLANAntiSpoof = True;

} else if (FromLAN and LocalVLAN == True) {

    PassVLANAntiSpoof = FALSE;

} else PassVLANAntiSpoof = True;

    PassAntiSpoof = PassVLANAntiSpoof and PassMACAntiSpoof;

If (PassAntiSpoof == False)  {DPort = null; Goto Mirroring Rules }// Drop Packet as
DPort is empty;

// Switching algorithm

// Perfect Filters

For Each entry e in MacVlan Table {

    If (DA == e.MAC and (VID == e.VLAN or (VID == NULL and e.VLAN == 0)))

    PFDPort = PFDPort + e.DestinationVSIList; // Add VSI matching the MAC, VLAN
pair.

}

For Each entry e in Mac Table { :

    If ((DA == e.MAC )

        PFDPort = PFDPort + e.DestinationVSIList; // Add ports matching the MAC
only entry.

}

// Imperfect Filters

    For Each entry e in HashMACVlan Table where (HDA == e.HashMAC and (VID ==
e.VLAN or (VID == NULL and e.VLAN == 0 and AddressType == e.AddressType))):

        IFDPort = IFDPort + e.DestinationVSIList; // Add VSIs matching the HashMAC,
VLAN pair.

    For Each entry e in HashMAC Table where (HDA == e.HashMAC and AddressType ==
e.AddressType):
```

```
        IFDPort = IFDPort + e.DestinationVSIList;// Add VSIs matching the HashMAC.

}

// Apply promiscuous modes

For each port p in VEB {

    If (AddressType == Unicast and p.PUE == 1) PMDPort = PMDPort + p;

    If (AddressType == Multicast and p.PME == 1) PMDPort = PMDPort + p;

    If (AddressType == Broadcast and p.BAM == 1) PMDPort = PMDPort + p;

}

// Apply per VLAN promiscuous modes

For Each entry e in Promiscuous_List Table where (VID == e.VLAN and AddressType ==
e.AddressType):

        PMDPort = PMDPort + e.DestinationVSIList; // Add VSIs matching the
promiscuous list.


// Cloud filters

// Application Source-IP, Inner MAC filtering. Filter 0xD in Add Cloud Filters
command

For Each entry e in SourceIpInnerMAC Table where (InnerMAC == e.InnerMAC and
SourceIP == e.SourceIP):

    CDPort = CDPort + e.DestinationVSI;

// Inner MAC/VLAN/Key filtering. Filter 0x4 in Add Cloud Filters command

For Each entry e in InnerMAC_InnerVLAN_Key Table where (InnerMAC == e. InnerMAC and
InnerVLAN == e.InnerVLAN and Key == e.Key_Key Low and EncapsulationType =
e.EncapsulationType):

        CDPort = CDPort + e.DestinationVSI;

// Inner MAC filtering. Filter 0xA in Add Cloud Filters command

For Each entry e in InnerMAC Table where (InnerMAC == e. InnerMAC):

        CPort = CPort + e.DestinationVSI;

// Inner MAC/Key filtering. Filter 0x6 in Add Cloud Filters command

For Each entry e in InnerMAC_VLAN_Key Table where (InnerMAC == e. InnerMAC and Key
== e.Key and EncapsulationType = e.EncapsulationType):

        CDPort = CDPort + e.DestinationVSI;

// Inner MAC/VLAN filtering. Filter 0x3 in Add Cloud Filters command

For Each entry e in InnerMAC_VLAN Table where (InnerMAC == e. InnerMAC and
InnerVLAN == e.InnerVLAN):

        CPort = CPort + e.DestinationVSI;

// Outer MAC/Key/Inner MAC filtering. Filter 0xB in Add Cloud Filters command

For Each entry e in InnerMAC_OuterMAC_Key Table where (InnerMAC == e.InnerMAC and
OuterMAC == e.MAC and Key == e.Key_Key Low and EncapsulationType =
e.EncapsulationType):
```

```
        CPort = CPort + e.DestinationVSI;

// Inner IP filtering. Filter 0xC in Add Cloud Filters command

For Each entry e in InnerIP Table where (InnerIP == e.InnerIP):

        CDPort = CDPort + e.DestinationVSI;

// Application Source-IP, Inner MAC filtering. Filter 0xD in Add Cloud Filters
command

For Each entry e in SourceIpInnerMAC Table where (InnerMAC == e.InnerMAC and
SourceIP == e.SourceIP):

        CDPort = CDPort + e.DestinationVSI;

// Application Source-IP, Inner MAC filtering. Filter 0xD in Add Cloud Filters
command

For Each entry e in SourceIpInnerMAC Table where (InnerMAC == e.InnerMAC and
SourceIP == e.SourceIP):

        CDPort = CDPort + e.DestinationVSI;

// L2 filtering rule

// Checks that the packet source MAC address was added by one of the MAC address
based filters (Either initial PF MAC address, Add MAC VLAN AQ command, Add
Ethertype AQ command or filer 9 in Add Cloud Filters.

if (DA does not match any entry in MAC table and L2 filtering is enabled) CDPort =
NULL;


// Override filters

For Each entry e in Ethertype Table where (Etype == e.Etype and Source ==
e.Source):

    OFDPort = OFDPort + e.DPortList; // Control port filtering.

For Each entry e in MacEthertype Table where (DA == e.MAC and Etype == e.Etype and
Source == e.Source):

    OFDPort = OFDPort + e.DPortList; // Control port filtering.

// Create unified list

If (MngOnly) DPort = MNGPort;

Else If (OFDPort not empty) Dport = OFDPort + MNGPort; // Override

Else if (CPort not empty and Promiscuous_Joins_Exact) DPort = CPort + PMDPort; //
cloud & promiscuous

Else if (PFDPort not empty and !Promiscuous_Joins_Exact)) DPort = IFDPort +
MNGPort; // exact only

Else if (IFDPort not empty and Promiscuous_Joins_Exact) DPort = IFDPort + PMDPort +
MNGPort; // imperfect & promiscuous

Else if (IFDPort not empty and !Promiscuous_Joins_Exact) DPort = IFDPort + MNGPort;
// imperfect only

else DPort = PMDPort + MNGPort; // promiscuous

Dport_for_mirroring = Dport;
```

```
if (DPort is empty and FromLAN and DefaultVSI != Null)

    DPort = DefaultVSI;

// Fall back to PV default port if no match to any filter.

if (Dport is empty and PortDefaultVSI != Null and FromLAN) DPort = PortDefaultVSI;

// VLAN egress Filtering

// Note - to use VLAN only fitering (ignore all MAC addresses), the UPE, MPE and
BAM bits should be set in all ports.

For each entry p in DPort {

    if (p.VLANPruneEnable == True) {

        If (VID match VLAN in VLAN table) {

            for each e in Egress Vlan Table where (VID == e.VLAN or (VID == NULL and
e.VLAN == 0))

                if (p NOT in e.EgressVSIList) DPort = DPort - p; // prune ports
which are not member of the VLAN;

        }

        Else Dport = Dport - p; // Remove all ports which are not in Promiscuous
VLAN if VLAN not found in VLAN table.

    }

}

// Add external ports if needed

Bool May_loopback_for_mirror = FALSE;

// Note: In VEPA mode, all Tx packets should go to LAN only (VSI[i].AllowLoopback
== 0)

if (

    (PFDport is empty or AddressType == Multicast or AddressType == Broadcast or
VSI[i].AllowLoopback == 0) and (CPort is empty or VSI[i].AllowLoopback == 0) and
OFDPort is empty and FromHost and LocalVLAN == FALSE and SVSI.EnableLAN) {

    DPort = Dport + LAN port;

    If (DPort = LAN Port) May_loopback_for_mirror = TRUE;

}

// Prunning Rules

// Control Packets drop rules.

PacketDropped = FALSE

For Each entry e in Ethertype Table where (Etype == e.Etype and Source ==
e.Source):

        if (e.Drop == True) { DPort = Null;DPort_for_mirroring = NULL;
PacketDropped = True}

    For Each entry e in MacEthertype Table where (DA == e.MAC and Etype == e.Etype
and Source == e.Source):

        if (e.Drop == True) { DPort = Null;DPort_for_mirroring = NULL;
```

```
PacketDropped = True}

// Find the source port based on the source MAC address.

// For Each entry e in Mac Table not pointing to a list // unicast adresses are
pointing to a single VSI

        if (SA == e.MAC ) SourcePort = e.MAC;

For Each entry e in MacVLAN Table not pointing to a list // unicast adresses are
pointing to a single VSI

        if (SA == e.MAC and VLAN = e.VLAN ) SourcePorts = e.MAC;

if (DoNotLoopback or SVSI.ALLOWLOOPBACK == FALSE) SwitchToLocal = FALSE;

// Note - switch to local may be also disabled via a specific port setting to LAN,
but this is handled below in the Specific_port handling.

For Each port p in Dport {

        // Remove ports from list if loopback disabled for this port except for
local iWARP packets

    if (From Host and SwitchToLocal == FALSE and p != LAN Port) {

        DPort = Dport - p;

        DPort_for_mirroring = DPort_for_mirroring - p;

    }

    // Remove uplinks  from list if requested by descriptor

    if (p == LAN Port and SwitchToLAN == FALSE) {

        Dport = Dport - p ;

        DPort_for_mirroring = DPort_for_mirroring - p;

    }

    // Prune Source port

if (p.MACPruneEnable== True and ( p = SourcePort or p == Source VSI)

    {

        DPort_for_mirroring = DPort_for_mirroring - p;

    }

// Prune by size is done at the queue level, so this is not handled in the switch.

}

    }

}

// Mirroring rules

MatchedRuleID = Null;

for ( i = 1 to n_Mirror) { // Apply all mirroring rules

// Note - a port can be member only in a single VSI based mirror rule.

    Bool Mirror = FALSE;
```

```
    Bool VLAN_Mirror = FALSE;

If(FromHost) {

    Case Mirror rule[i].type {

     ALL_EGRESS {

     If (DPort not empty) Mirror = True;

    };

    ALL_INGRESS {

         If (Specific_port == NULL or Specific_port == LAN) Mirror =
        True;

    };

    VSI_INGRESS {

    // This rule is applied even if packet is dropped.

    if (SPort in Mirror rule[i].IngressPortList and (Specific_port == NULL or
Specific_port == LAN)) Mirror = True;};

    VSI_EGRESS {

        For each p in Dport {

            if (p in Mirror rule[i].EgressPortList) Mirror = True;

        }

    };

};

If (FromLan) {

  Case Mirror rule[i].type{

   ALL_EGRESS {

    Mirror = True;

   };

   VSI_EGRESS {

    For each p in Dport_for_mirroring {

            if (p in Mirror rule[i].EgressPortList) Mirror = True;

        }

    };

};

    if (Mirror rule.type = VLAN_MIRROR and VID in Mirror rule.VLANList and and
Specific_port == NULL and PacketDroped = FALSE and (SVSI.ALLOWLOOPBACK or FromLAN))
VLAN_Mirror = True;

    if (Mirror) {

        DPort = Dport + Mirror rule[i].MirrorPort;

        MatchedRuleID = i; // Only a single non VLAN rule should match
```

```
    }
if (VLAN_Mirror) {
    DPort = Dport + Mirror rule[i] .MirrorPort;
    If (May_loopback_for_mirror) DPort = Dport + DefaultVSI/PortDefaultVSI;
    }
}
Return Dport // Note - the Dport list, shuld include a single copy of each VSI


HashFunction(MAC){
Hash = MAC[6:0] // Use the 7 least significant bits of the MAC address (last bits
on the wire).
Return Hash
}
```

38.21.7.14 Switch Programming Model

38.21.7.14.1 Switching Structure Representation

This section describes the model used to represent the switching elements.

The objects represented in the models are identified by a 10-bit device-wide Switch Element Identifier (SEID). The SEID of an element is returned after creating the element and should be referenced when creating ties between elements. The SEID of elements created automatically can be retrieved using the *Describe Structure* command.

The objects represented can be either external or internal to the switch. Possible external elements are listed in Table 38-191.

Table 38-191.External Elements

External Element	Abbreviation	Element Type	Description
Physical Port MAC	MAC	1	This element is always present. An SEID is allocated at power on for each enabled port.
Physical Functions	PF	2	PF elements is created at PCIe reset for each enabled PF.
Virtual Functions	VF	3	VF elements are created when SR-IOV is enabled on a physical function.
Embedded Micro Processor	EMP	4	The embedded micro processor can be used as a control port for the different switching elements. All the ties to the EMP should be explicitly created when creating an element. An EMP can contain multiple EMP queues that are only used to represent logical connections to the switch. This is used to enable multiple control ports in the EMP. This element includes the MC interface used to enable pass-through traffic to an external MC via an out-of-band connection.

The internal elements are elements that are used as part of the switching decision. The following elements are listed in Table 38-192.

Table 38-192.Internal Elements

Internal Element	Abbreviation	Element Type	Description	Related Commands	Described In Section
Port Virtualizer (S-comp or Port Extender)	PV	16	Defines an S-comp or a port extender. Together with a PV, create a set of S-channels elements	Add port virtualizer. Update Port Virtualizer parameters. Get Port Virtualizer parameters.	38.21.7.4.2
S-channel	SC	N/A	An S-channel created as part of an S-comp/M-comp element. There is no separate S-channel element. An S-channel connected directly to an external element (PF/VF/EMP) is considered a VSI. An S-channel connected to a VEB or a VEPA is considered as part of the VEB/VEPA.	N/A	38.21.7.4.2
L2 Filter	L2	N/A	An L2 filter element. Such an element is created by default to connect a MAC and the first PF tied to this MAC. An L2 is not created as part of the topology. Rather it is implemented using the filtering options to the VSI tied to it.	N/A	38.21.7.4.4
VEB/Port Aggregator	VEB/VEPA	17	A VEB or a virtual port aggregator.	Add VEB Get VEB parameters	
Virtual Station Interface	VSI	19	A VSI can be part of a VEB, VEPA or PV element and is connected to an host external element (PF, VF or EMP).	Add VSI. Update VSI parameters. Get VSI parameters. Delete element.	38.21.7.9.2

The SEIDs are allocated according to firmware. Note that a fixed mapping should not be assumed.

Table 38-193 lists the representation of the switching hierarchy and should be used by embedded firmware and the host software as a common database.

Table 38-193.Switching Structure Representation

Entry in Table	Description	Notes
SEID	The handle to the element.	
Element Type	As listed in Table 38-191 and Table 38-192.	
Up Link Connection(s)	Indicates the SEID of the switching element connected directly below the current element. Below means towards the network in this case.	For sub-elements of a composed switch element, the uplink is the containing element.
Down Link Connection(s)	Indicates the SEID of the switching element connected directly above the current element. Above means towards the host in this case.	The downlink is provided only for elements connected to PFs, VFs or to the EMP. In other cases, the value is zero.
Control Port	For elements with a control port (points to the control port).	
Scheduler Node	The scheduler node associated with this switch element.	

Table 38-194 lists an example of the representation of a switching configuration as shown in Figure 38-51.

Figure 38-51.Switching Structure Example

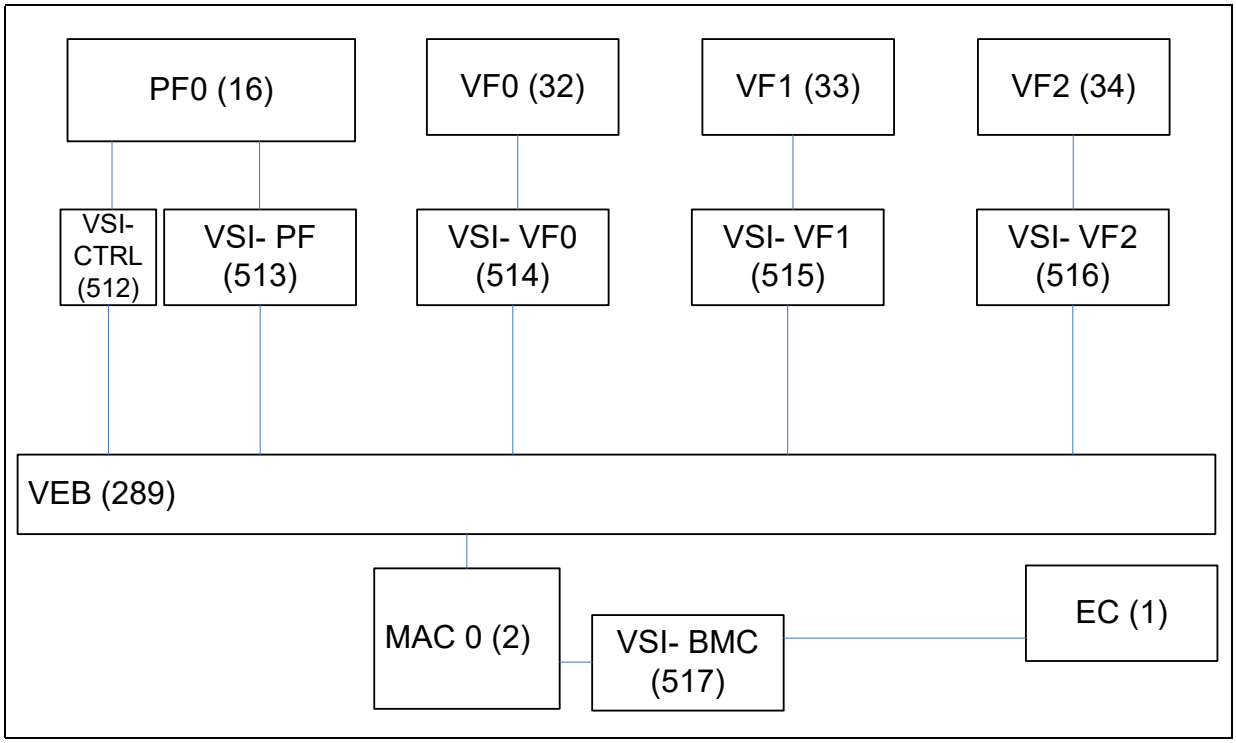


Table 38-194.Switching Structure Representation - Examples

SEID	Description	Element Type	Uplink	Downlink
0	Null	Null	0	0
1	Embedded Controller	EMP	517	0
2	Port 0 MAC Port	MAC	0	0
16	Physical Function 0	PF	512	0
32	VF0	VF	514	0
33	VF1	VF	515	0
34	VF2	VF	516	0
289	VEB	VEB	2	0
512	VSI (PF)	VSI	289	16
513	VSI - Control VSI for VEB	VSI	289	16
514	VSI (VF0)	VSI	289	32
515	VSI (VF1)	VSI	289	33
516	VSI (VF2)	VSI	289	34
517	VSI (EMP)	VSI	2	1

38.21.7.15 Supported Switch Profiles

This section describes the supported switching profiles.

The profiles described are per port.

The following profiles are supported by the 10 GbE controller:

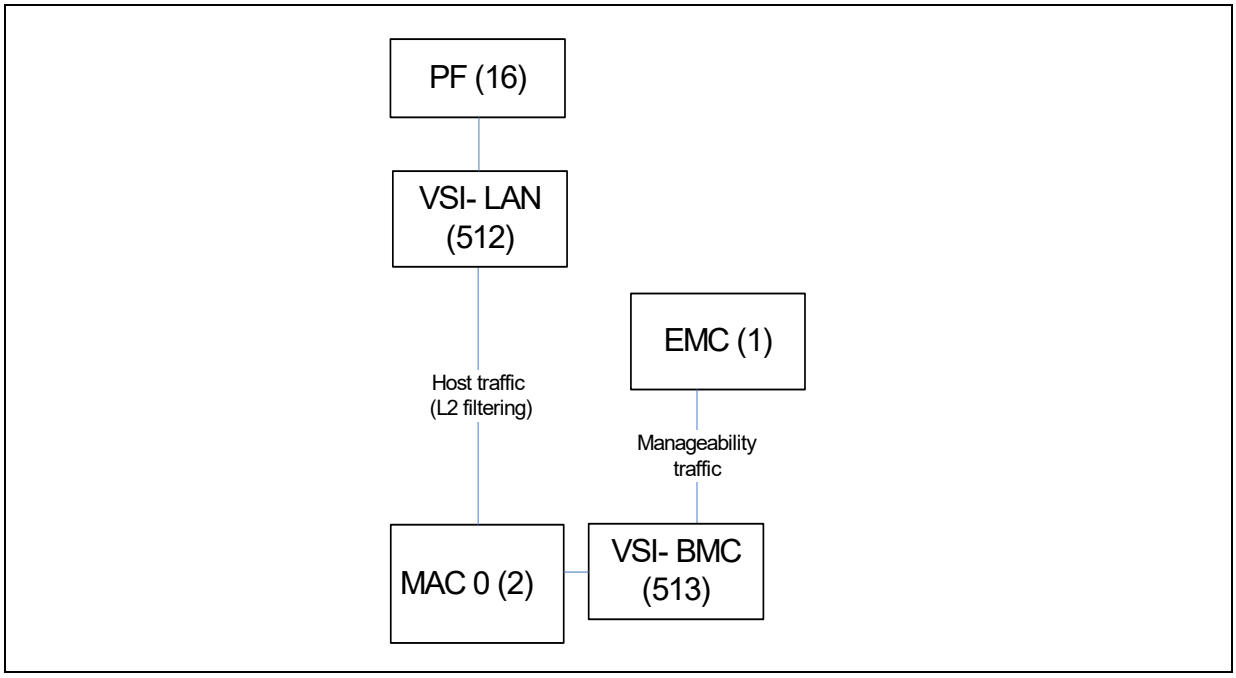
1. Basic L2 (default configuration).
2. Basic L2 + iWARP.
3. VEB/VEPA only (with or without iWARP).
4. Port Virtualizer.
5. Port Virtualizer + VEB/VEPA.

Note: There is no central location in which a profile is set. These profiles are examples of possible configuration that are expected in the 10 GbE controller. The configuration of each profile is done using the admin commands described in Section 38.21.7.17.2

38.21.7.15.1 Basic L2

A basic L2 configuration is used for monolithic OSes (non virtualized) systems where an iWARP function is not using a separate MAC, VLAN pair and the network is not distributed to virtual channels. In this case, the switch configuration is as follow:

Figure 38-52.Switch Configuration (Basic L2)



Basic L2 (Resource Allocation)

In this profile, all resources allocated in the NVM to the port are available to this single function.

38.21.7.15.2 Basic L2 + iWARP

A basic L2 configuration is used for monolithic operating systems (non-virtualized) where an iWARP function is exposed and the network is not distributed to virtual channels.

Note: The control port of the VEB can be either a separate VSI or part of the regular VSI of the PF.

Basic L2 + iWARP (Resource Allocation)

In this mode, a single queue of the PF is assigned for iWARP stateless traffic. All the other queues are used by the LAN function.

38.21.7.15.3 VEB/VEPA

A VEB configuration is used for virtualized operating systems where the network is not distributed to virtual channels. The switch configuration for the VEB/VEPA case is described in the following figure.

The use cases for VEB and VEPA are described in VEB for Direct Connected VMs and Virtual port Aggregator Model, respectively.

Note: The following figure shows both VFs with a separate VSI for iWARP traffic and VFs with a single VSI.

The control port of the VEB can be either a separate VSI or part of the regular VSI of the PF.

VEB (Resource Allocation)

There is no limitation on the number VSIs on a single VEB (apart from the total numbers of VSIs supported) and up to 4 VEBs connected to a single PF.

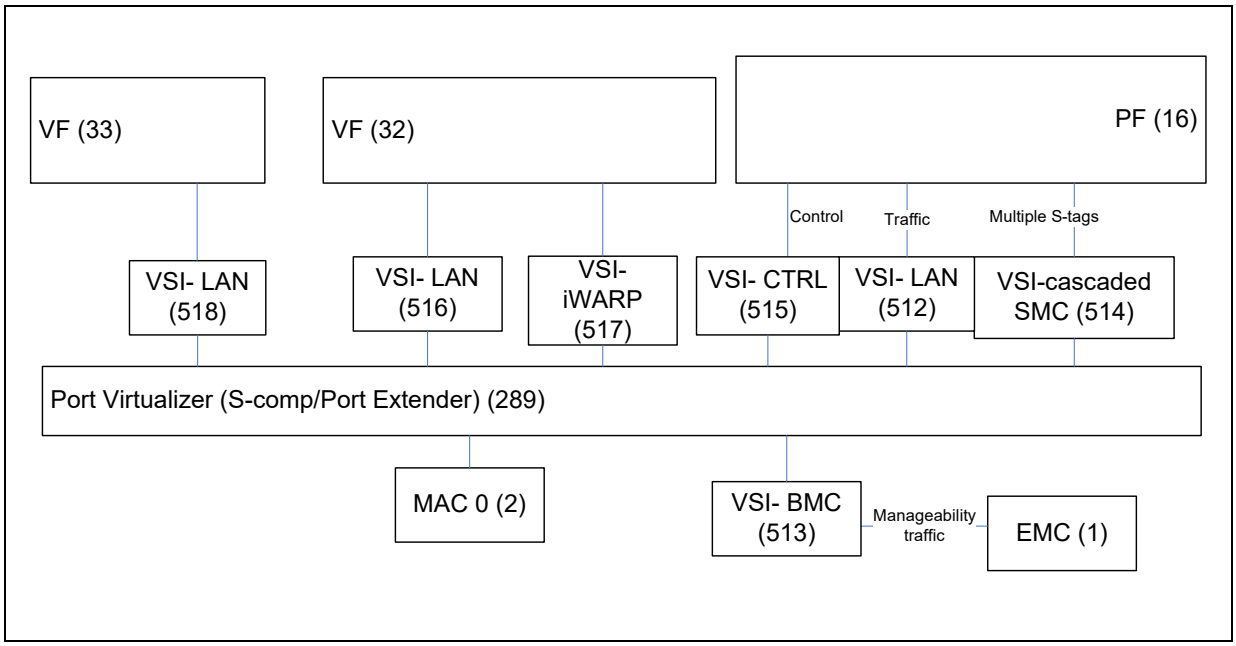
The switching resources are allocated to the PF. The PF can decide to allocate the resources to any of the VEBs or to any of the VSIs according to its own policy.

38.21.7.15.4 Port Virtualizer

This mode is used when all the switching decisions are done in an external switch and each VSI is connected directly to one or more virtual channels. The usage models of the port virtualizer profile are described in Port Virtualizer Model. The switch configuration for the port virtualizer case is shown in the following figure. In the case of a VF requiring multiple VSIs, it might request a new VSI using the *Add VSI* command. This command is forwarded to the PF for completion.

Note: The control port of the port virtualizer can be either a separate VSI or part of the regular VSI of the PF. In this case, the tagging of packets is controlled by the host. Packets sent from the control port should be non-tagged.

Figure 38-54.Port Virtualizer



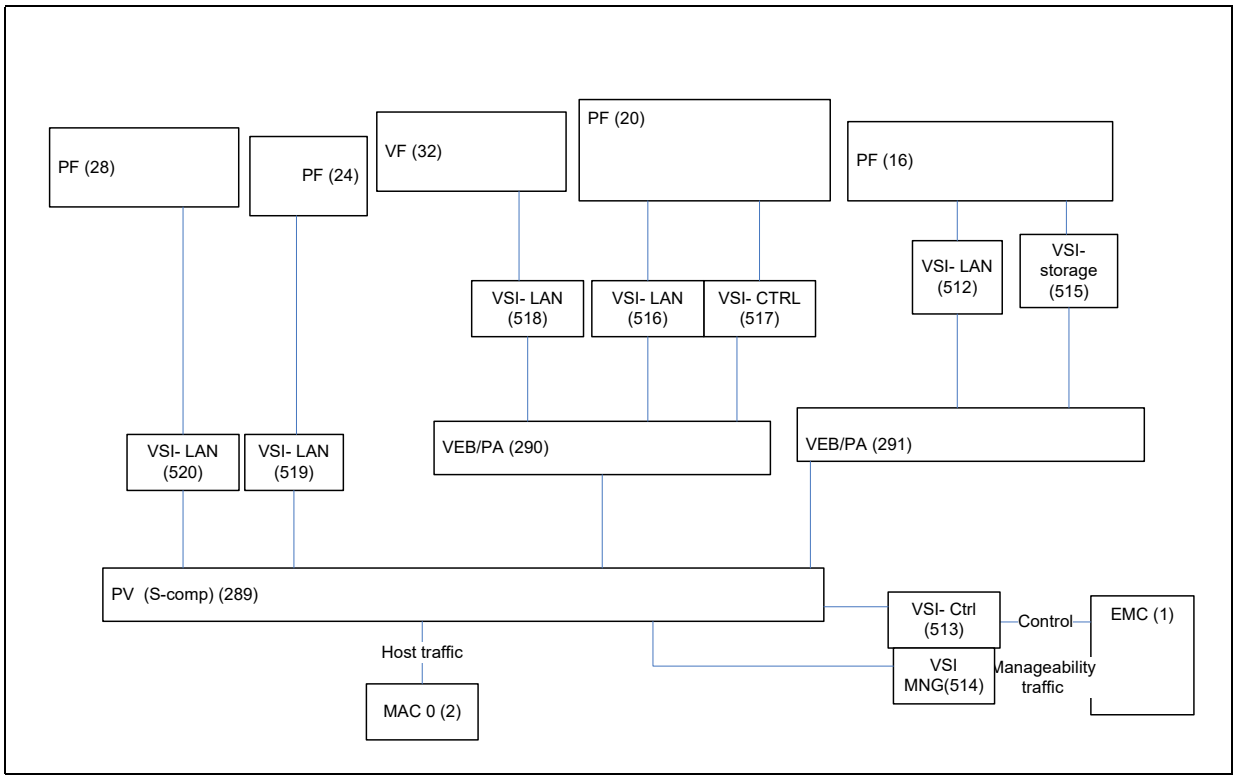
38.21.7.15.5 Port Virtualizer + VEB/VEPA

A port virtualizer +VEB/VEPA configuration is used for virtualized operating systems where the network is distributed to virtual channels. The switch configuration for the port virtualizer +VEB/VEPA case is shown in the following figure. For a VF requiring multiple VSIs, it might request a new VSI using the *Add VSI* command. This command is forwarded to the PF for completion.

Note: The control port of the port virtualizer in this mode is the EMP, as there are multiple PFs connected to the same port virtualizer.

A special case of this mode is the mode where the link is divided to virtual channels and each channel is connected to a PF, but none of these are virtualized. In this case, there are no VEB/PAs in the device (unless needed for VLAN and iWARP traffic as described in Section 38.21.7.15.2).

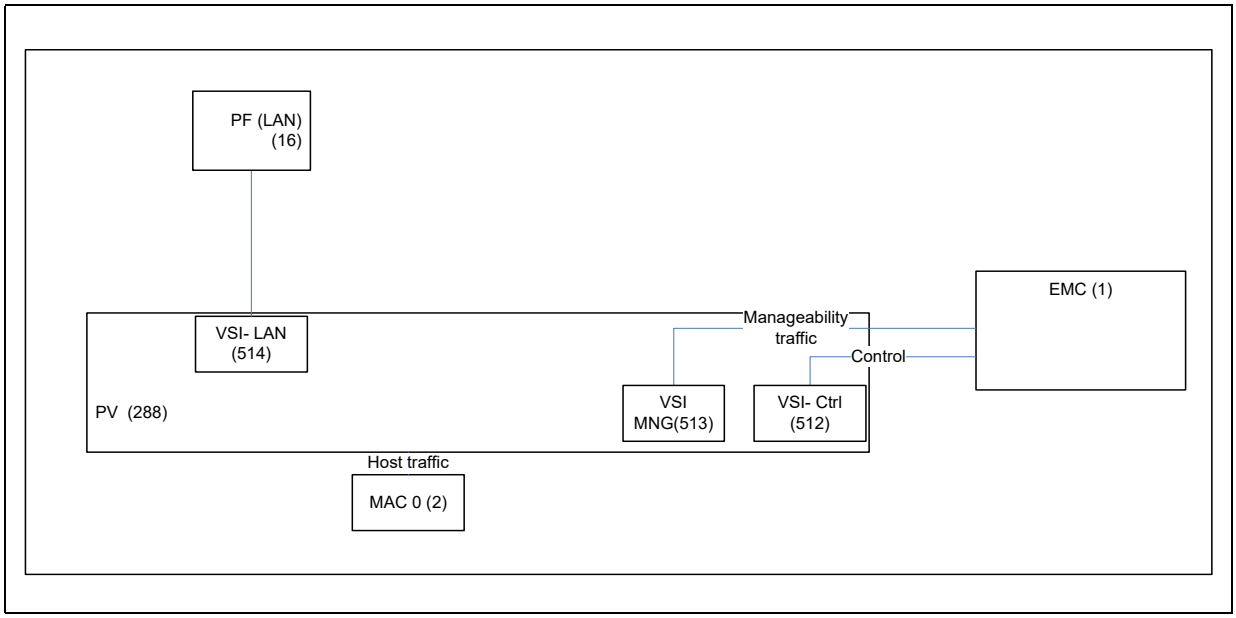
Figure 38-55.Port Virtualizer + VEB/VEPA



38.21.7.15.6 LAN S-comp

The initial configuration as exposed by the device is shown in the following figure.

Figure 38-56.LAN Initial Configuration



38.21.7.16 Switch Resource Allocation

38.21.7.16.1 Allocated Resources

Table 38-195 lists the resources that are part of the switch and should be allocated to the different switch elements.

Table 38-195 also lists the switching elements resources. These resources are available no matter what the configuration of the forwarding tables is. Table 38-196 lists the forwarding rules resources. These resources are different for different configuration of the forwarding tables.

Note: Each element is associated with a port, so two ports using the same value (such as the same MAC address) consumes two entries.

Table 38-195.Switching Elements Resources

Resource	Total Available	Notes
VEBs	16	There is no limitation on the number of VEBs assigned to a PF.
VSIs	384	Up to 12 of the 384 VSIs are reserved for EMP traffic (LLDP and pass-through traffic forwarding per port) and should not be allocated.
VSI List Entries	6144	Three 3 VSIs in each entry. These entries are used to create up to 2048 VSI lists such as multicast VSI lists or VLAN membership lists.
VLAN Statistic Pools	128	
TC Statistic Pools	128	Statically associated with the 16 VEBs (8 sets for 8 TCs for each of the VEBs).
Mirror Rules	64	
Queue Sets	1024	
Teredo UDP Ports	16	

Table 38-196 lists the resources allocated according to the filters configuration used. The image used is defined in the *Features enable.Switching mode* NVM field.

Table 38-196.Switching Rules Resources (Sheet 1 of 2)

Resource	Total Available	Notes
NVM Map	**Unified Image (B0)**	
First level filters		
Perfect Match MAC Addresses	1535	Each MAC address can be used for multiple VEBs within the same port. There can be up to 2048 VEB, MAC associations. One MAC address is statically allocated to the EMP.
VLANs	256	The entire range of 4096 VIDs can be covered; however, only 256 different VLANs can be stored in the forwarding tables. Each VLAN can be used in multiple VEB within a port. There can be 1024 VEB, VLAN associations.
EtherType Filters	64	These filters are tuples of switch ID, EtherTypes and direction.
S-tags	448	The entire range of 4096 S-VIDs can be covered, however, only 448 different S-tags can be stored in the forwarding tables.
Promiscuous Rules	1024	Unicast, broadcast or multicast promiscuous for different VEBs.
VLAN Promiscuous		The entire range of 4096 VIDs can be covered, however, only 1024 different VLANs/packet types (unicast, multicast, broadcast) /VEB can be stored in the forwarding tables.
Multicast Hash Entries	512	
Unicast Hash Entries	512	

Table 38-196.Switching Rules Resources (Sheet 2 of 2)

Resource	Total Available	Notes
NVM Map	**Unified Image (B0)**	
Inner VLAN	512	Inner VLAN used for UDP cloud filtering.
VN Key Low / GRE Key/Tenant ID		
Inner MAC	1024	Each such filter added occupies an entry in the table shared with the S-tag, VLAN mirroring filters.
S-tag, VLAN pairs (mirroring)	1024	Each such filter added occupies an entry in the table shared with the inner MAC filters.
Destination IPv4 / IPv6 Address (Application IP)	1024	Each such filter added occupies an entry in the table shared with the inner MAC, inner VLAN filters.
Source IPv4 / IPv6 Address (Application IP)	1024	Each such filter added occupies an entry in the table shared with the MAC-EtherType filters.
Combined filters		
MAC,VLAN Pairs	3072	For each entry in this table, an entry in the perfect match MAC addresses and a VLAN entry is used. The same is true for all the other types of filters.
Hash MAC,VLAN Pairs	1024	For each entry in this table, an entry in the hash MAC addresses and a VLAN entry is used. The same is true for all the other types of filters.
MAC-EtherType Filters	1024	Each entry is associated with an S-tag or switch ID. For each entry in this table, an entry in the Perfect match MAC addresses and a EtherType filters entry is used.
MAC L2 Filtering	512	This filter shares the same entries as the perfect match MAC address.
Inner MAC, Tenant ID	1024	
Outer MAC, Tenant ID, Inner MAC		
Inner MAC, Inner VLAN, Tenant ID	1024[1]	For each entry in this table, an entry in the inner MAC addresses, in the inner VLAN and a tenant ID filters entry is used.
Inner MAC, Inner VLAN Pair	1024[2]	For each entry in this table, an entry in the inner MAC addresses, in the inner VLAN filters entry is used.

Notes:

1. Shared between inner MAC, inner VLAN, tenant ID filters and inner MAC filters.
2. Shared between inner MAC, inner VLAN filters and inner IP filters.

38.21.7.16.2 PFs Allocation Method

The resources can be categorized as shared resources that are used by multiple PFs and resources that are allocated to a single PF. Resources used by multiple PFs are inherently shared and thus cannot be allocated to any PF. Each PF requesting a resource gets it given there are still empty entries in the shared pool. It is the responsibility of each software device driver to release resources it does not use.

For resources allocated to a single PF, there is a basic allocation in the NVM for each PF (that can be zero). The resources not reserved to a PF are allocated on a first come first served basis.

The *PF allocations* structure in the NVM contains the resource allocation initial values.

If, after the enumeration stage, a PF is not enabled, its resources are given back to the shared pool.

Resources are allocated first from the dedicated pools and only if the dedicated pool of a PF is exhausted, a resource from the shared pool is given.

The resources that can be dedicated to a PF are:

- VEBs (TCs statistics pools are also allocated as part of the VEB)
- VSIs
- Perfect match MAC addresses
- S-tags
- VLAN statistic pools
- Mirror rules
- Queue sets

All the other resources are considered as shared.

Note: The response of commands used to allocate dedicated resources includes information about the dedicated and shared resource left after the command is applied. The response of commands used to allocate shared resources includes information about the shared resource left after the command is applied.

The *Get Switch Resources Allocation* command returns the reserved allocation of each type of resource and the current level of usage of each resource.

Statistics Allocation

There are a few sets of statistics used by the switch. The allocation of statistics resources to the different switching elements is as follows:

- For each VSI, a set of per VSI statistics is allocated. The set to use is returned in the *Add VSI* response buffer in the *Statistic Counters* field.
- For each port and port virtualizer, a set of per-port/port virtualizer statistics accessed using per-port registers is allocated.
- For each VEB allocated to a function, a set of VEB uplink statistics and per VEB per TC statistics might be allocated if statistics gathering is not disabled in the Add VEB command. The set to use is returned in the response of the Add VEB command in the *Statistic Counter* field.
  - In addition, a VEB can be allocated a set of statistics per VLAN per VEB using the *Add Statistics* admin command. The set to use is returned in the *Statistic Counters* field in the response of this command. This request might fail if there are no free sets.

When statistics are not gathered, the following statistics are not valid for this VEB:

- GLPRT_RUPP[0]
- GLSW_GOTCH/L
- GLVEBVL_GOTC_[n]
- GLVEB_TCBCH/L[n]
- GLVEB_TCPCH/L[n]
- GLSW_UPTCH/L[n]
- GLSW_MPTCH/L[n]
- GLSW_BPTCH/L[n]
- GLSW_GORCH/L[n]

- GLVEB_VLBCH/L[n]
- GLVEB_RCBCH/L[n]
- GLVEB_RCPCH[n]
- GLSW_UPRCH/L[n]
- GLSW_MPRCH/L[n]
- GLSW_BPRCH/L[n]
- GLSW_RUPP_[n]
- GLVEB_VLUPCH/L[n]
- GLVEB_VLMPCH/L[n]
- GLVEB_VLBPCH/L[n]

38.21.7.16.3 VFs Allocation Method

The 10 GbE controller manages VF resources as part of the PF resources. The PF driver should manage the VF requests based on the PF allocation.

38.21.7.17 Switch Init Flow

The init flow of the switch is composed of the following steps:

1. Pre BIOS stages:
   a. Initializing the switch hardware (Section 38.21.7.17.1).
   b. Initializing a basic switch configuration according to the NVM (Section 38.21.7.17.2).
   c. Define the supported virtualization modes.
   d. Update from external configuration source like the EVB management protocol.
2. BIOS configuration (Section 38.21.7.17.3).
   a. Update of the switch configuration according to SMASH CLP.
   b. Configuring switch according to the updated configuration.
3. Software configuration (Section 38.21.7.17.4).
   a. Optionally, creating VEB or VEPA elements by a VMM.
   b. Allocating MAC addresses and VLANs to VSIs.

At the end of stage 1, basic connectivity to the pass-through management traffic and the EMP is enabled. In addition, WoL functionality is also available. If already known, the connectivity to PFs is also provided.

At the end of stage 2, connectivity to the host is added This is the configuration exposed to the pre-boot software device driver and to the operating system.

38.21.7.17.1 Initializing Switch Hardware

The switch is implemented using a programmable logic, enabling parsing of different packet types and implementing different forwarding rules. This programmable hardware is configured as part of the 10 GbE controller auto load of the CORE Registers Auto-load NVM section. The lookup tables and the switching rules logic are configured at this stage.

This stage is activated at each core reset.

38.21.7.17.2 Initial Switch Configuration

The internal EMP firmware defines the initial switch structure. This stage is done after all the hardware auto load is done and the number of currently enabled PCIe PF functions is known. Before any NVM configuration the following elements are defined by firmware:

Table 38-197. Initial Elements

Elements	SEID	Note
EMP	1	
Ports	2-5	Only for the ports enabled.
Physical functions	16-31	Only for the functions enabled.
Virtual functions	32-159	For all VFs. VFs not used will not be connected to VSIs.

At this stage, firmware loads the switch configuration from the NVM. The *EMP Settings* NVM module contains the entire configuration needed. It also contains the following information relevant to the switch operation:

- A list of capabilities supported by the switch (type of filtering, types of connections).
- PF allocations module (see PF Allocations).

This stage is part of the firmware initialization stage.

This stage is activated at each core reset.

Initial VSIs

At initialization time, only minimal forwarding capabilities are needed:

- Forwarding of pass-through traffic to manageability interface
- Forwarding of LLDP traffic to the EMP
- WoL detection

The initial switching configuration includes only the forwarding rules needed to support these flows.

At this stage, the host is not connected to the network and cannot get traffic. If an additional pre-boot configuration is not expected, then the connectivity to the host is also done at this stage.

The basic configuration for port connectivity is achieved by connecting a single VSI to each port to one PF. The connection is from the port to the function connected to it with the lowest function number. The mapping of functions to port is reflected in the *PFGEN_PORTNUM.PORT_NUM* field. These VSIs are created with a *Regular Data Port* connection type and a PF VSI type.

All the VSIs created at this stage have the default switch ID of the port. The following parameters should be set to a non-default value in these VSIs:

- Allow destination override should be set.
- For the PF VSIs, *Queue Mapping* should be set to contiguous and all the queues of the PF should be allocated to this VSI. The first queue is always queue 0 (of the PF).

As this flow is applied on each core reset, it is possible that a valid alternate RAM already exists. In this case, the alternate RAM based configuration, as described in Section 38.21.7.17.3, is applied.

PF Allocations

For each PF, the *PF allocation* sub-module in the *EMP Core Module* contains guaranteed allocations for the following resources:

- VEBs (TCs statistics pools are also allocated as part of the VEB)
- VSIs
- Perfect match MAC addresses
- S-tags
- VLAN statistic pools
- Mirror rules
- Queue sets
- Inner MAC
- IP addresses

After the basic topology is defined, an L2 filter should be added for each PF using the factory MAC address of this PF as stored in the PF allocations structure in the NVM or any alternate RAM override of those addresses. The *Load PF MAC Address* field in the *PF flags* word in the *PF allocation* sub-module is used to define which MAC addresses to add to the switching decision. If this bit is set, only packets that passes the MAC and if needed, the S-tag are forwarded to the PF.

In addition an L2 filter forwarding packets to the EMP should be added for each port using the port MAC addresses stored in the *PRTGL_SAL* and *PRTGL_SAH* registers.

By default, resources are allocated from the dedicated pool first and only when the dedicated pool is consumed or a *Use Shared* flag is set, the shared pool is used. A resource assigned from the dedicated pool can be used by a single PF only. An attempt to use it by another PF is treated as an error.

Firmware tracks and reports only the dedicated resources used by a function and not the shared ones.

Software can indicate a MAC address should be allocated from the shared pool, in which case it can be used by multiple PFs. This indication should be set for any resource expected to be used by multiple PFs (such as multicast addresses). MAC addresses assigned/removed as part of the Add/Remove Control Filter AQs are taken from the shared pool.

VSIs are always allocated from the dedicated pool first and from the shared pool after that. Release of VSIs is from shared first and then from dedicated.

In order to use a resource from the shared pool, the Use Shared <resource name> flag should be set in the Add MAC, VLAN pair and in Add Cloud Filter Admin commands. Such a resource might be used by multiple PFs. If not set, it is expected the resource is used by a single PF.

Note: Teamed ports using the same MAC address might allocate the MAC address from the dedicated pool, as MAC addresses are qualified with the port number.

38.21.7.17.3 BIOS Configuration

As part of the SMASH CLP stage, commands affecting the switch configuration might be received. These commands should be translated by the SMASH CLP code to the updates of the alternate RAM.

After all the SMASH CLP commands are received, the CLP code should update the firmware with the content of the alternate RAM.

For each of the enabled functions, a VSI is added and is connected to the MAC.

The VSIs created at this stage have the default switch ID of the port. The following parameters should be set to a non-default value in these VSIs:

- Allow destination override should be set.
- Queue mapping should be set to contiguous and all the queues of the PF should be allocated to this VSI. The first queue is always queue 0 (of the PF).

Once the alternate RAM is updated and executed by the firmware, it might, depending on the configuration, disable some of the admin commands software is allowed to execute.

38.21.7.17.4 Operating System Initialization

After the operating system boot, each PF is connected to a physical port or an S-channel. Before adding further switching elements, the PF driver should use the *Get Switch Configuration* admin command to get the SEID of the switch element to which this function is connected. See Get Switch Configuration (0x0200) for more detail.

After that, it might add VEB, VEPA or port virtualizer elements according to the instructions received from the switch control plane in the operating system using the admin commands described in Section 38.21.7.18.

Note: In operating systems where two separate VSIs are used for LAN and iWARP, the software device driver should create a VEB using the *Add VEB* command and connect a separate VSI for the LAN queues and the iWARP traffic. If a VEB is created for virtualization purposes, the same VEB can be used to connect the iWARP and LAN traffic VSIs.

38.21.7.18 Programming Interface

The programming of the different switching elements is done using admin commands. Commands to configure a switching element can be received only from the control port of the element.

38.21.7.18.1 Switch Configuration Admin Commands Summary

Table 38-198 lists the different commands used to configure switch elements.

Table 38-198. Switch Configuration Admin Commands (0x02xx) (Sheet 1 of 2)

Command	Opcode	Brief Description	Detailed Description
Generic Commands (0x020x)			
Get Switch Configuration	0x0200	Describe the networking structure of the port	Get Switch Configuration (0x0200)
Add Statistics	0x0201	Add a statistics block to a VLAN in a switch.	Add Statistics (0x0201)
Remove Statistics	0x0202	Remove a statistics block for a VLAN in a switch.	Remove Statistics (0x0202)

Table 38-198.Switch Configuration Admin Commands (0x02xx) (Sheet 2 of 2)

Command	Opcode	Brief Description	Detailed Description
Set Port Parameters	0x0203	Defines the default parameters of a LAN port	Set Port Parameters (0x0203)
Get Resources Allocation	0x0204	Reports the resources allocated to the PF.	Get Switch Resources Allocation (0x0204)
Set Switch Configuration	0x0205	Configure Switch global settings	"Set Switch Configuration Command (0x0205)"
VSI Commands (0x021x)			
Add VSI	0x0210	Add a VSI to a switching element	Add VSI (0x0210)
Update VSI Parameters	0x0211	Update parameters of a VSI	Update VSI (0x0211)
Get VSI Parameters	0x0212	Get the parameters of a VSI.	Get VSI Parameters (0x0212)
Port Virtualizer control (0x022x)			
Add Port Virtualizer	0x0220	Create an S-comp or a port extender	Add Port Virtualizer
Update Port Virtualizer Parameters	0x0221		Update Port Virtualizer Parameters
Get Port Virtualizer Parameters	0x0222		Get Port Virtualizer Parameters
VEB/Port aggregator Control (0x023x)			
Add VEB	0x0230	Create a VEB/Port aggregator	Add VEB
Get VEB Parameters	0x0232	Get the parameters of a VEB/VEPA	Get VEB Parameters (0x0232)
Switch Connectivity Configuration (0x024x)			
Delete Element	0x0243		Delete Element
Forwarding Table Configuration (0x025x)			
Add MAC,VLAN Pair	0x0250	Add a MAC/VLAN pair to the lookup table.	Add MAC, VLAN Pair (0x0250)
Remove MAC,VLAN Pair	0x0251	Remove a MAC/VLAN pair from the lookup table.	Remove MAC,VLAN Pair (0x0251)
Add VLAN	0x0252	Add a VLAN to the VEB/port aggregator (can be regular or local, primary or secondary).	Add VLAN (0x0252)
Remove VLAN	0x0253	Remove a VLAN or members of a VLAN from the VEB/port aggregator.	Remove VLAN (0x0253)
Set VSI Promiscuous Modes	0x0254		Set VSI Promiscuous Modes (0x0254)
Add S-tag	0x0255	Add an of S-tags to a VSI.	Add S-tag (0x0255)
Remove S-tag	0x0256	Remove an S-tag from a VSI.	Remove S-tag (0x0256)
Update S-tag	0x0259	Update the default S-tag of a VSI.	Update S-tag (0x0259)
Add Control Packet Filter	0x025A	Add EtherType (+MAC) filter to control port.	Add Control Packet Filter (0x025A)
Remove Control Packet Filter	0x025B	Remove EtherType (+MAC) filter from control port.	Remove Control Packet Filter (0x025B)
Add Cloud Filters	0x025C	Add a set of filters for cloud connections.	Add Cloud Filters (0x025C)
Remove Cloud Filters	0x025D	Remove a set of filters for cloud connections.	Remove Cloud Filters (0x025D)
Clear All WoL Switch Filters	0x025E	Remove all WoL related filters.	"Clear All WoL Switch Filters (0x0025E)"
Mirroring Configuration (0x026x)			
Add Mirror Rule	0x0260	Define a mirroring rule.	Add Mirror Rule (0x0260)
Delete Mirror Rule	0x0261	Remove a mirroring rule.	Delete Mirror Rule (0x0261)

38.21.7.18.2 Configuration Flow Examples

This section describes examples of software configuration flows for different types of switch topologies.

The following examples are provided:

- An SFP with a port virtualizer used to distribute the VM traffic. In this mode, each VF/VM is assigned a different S-channel.
- An SFP with a VEB used to distribute VM traffic. In this mode, each VF/VM is assigned a VSI within the VEB. The same flow can be used to add a VEPA element.

More complex network topologies are supported. The configuration of such topologies can be inferred from the examples that follow.

For each example, the admin command to use and the main parameters in each command are described.

Port Virtualizer (SFP)

In this mode, the internal firmware initiates a single VSI per port. This VSI is connected to the PF. Other VSIs might be connected to the EMP for MC or local traffic.

The software device driver should use the following command to create a port virtualizer and connect a set of VSIs to it:

- *Get Switch Configuration* — This command provides the default VSI connected to the port.
- *Get VSI* — This command provides the content of the current VSI context. The PF might then update this context using an *Update VSI* command.
- *Add* Port Virtualizer — This command adds a port virtualizer on top of the port. The *uplink SEID* should be the physical port. The *default VSI SEID* should be the default VSI received in the previous command.
- If needed, a separate control port can be added by using an *Add VSI* with the port virtualizer as the *uplink SEID* and a control port *connection type*.
- For each of the VMs/VFs that needs to be connected to the port virtualizer, an *Add VSI* should be sent with the port virtualizer as the uplink SEID, a regular data port *connection type* and a switch ID value equal to the S-tag assigned to this VM/VF. For VMs, the VSI type should be *VM*.

VEB (SFP)

In this mode, the internal firmware initiates a single VSI per port. This VSI is connected to the PF. Other VSIs might be connected to the EMP for MC or local traffic.

The software device driver should use the following command to create a VEB and connect a set of VSIs to it:

- *Get Switch Configuration* — This command provides the default VSI connected to the port.
- *Get VSI* — This command provides the content of the current VSI context. The PF might then update this context using an *Update VSI* command.
- *Add VEB* — This command adds a VEB on top of the port. The *uplink SEID* should be the physical port. The *default VSI SEID* should be the default VSI received in the previous command. The software device driver should register the switch ID assigned to this VEB.

- If needed, a separate control port might be added by using an *Add VSI* with the VEB as the *uplink SEID* and a control port *connection type*.
- For each of the VMs/VFs that needs to be connected to the VEB, an *Add VSI* should be sent with the port virtualizer as the uplink SEID, a regular data port *connection type* and a switch ID value equal to the switch ID of the VEB. For VMs, the VSI type should be *VM.*
- For each VM/VF, *Add MAC, VLAN pair* commands should be sent to allow adequate forwarding of the traffic.

38.21.7.18.3 Generic Commands (opcode 0x020x)

Get Switch Configuration (0x0200)

This command is used to discover the switch configuration of the port. The software device driver must use this command as part of the initialization flow before adding new switching elements or manipulating existing ones.

This function is available to any PF. This usually includes the switching structure from the port virtualizer and upwards.

Table 38-199.Get Switch Configuration Command (Opcode: 0x0200)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0200	Command opcode.
Datalen	4-5		Length of response buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
First SEID	16-17	0x0	If not zero, start the report from the requested SEID (used for continuation commands).
Reserved	18-23	0x0	Must be zero.
Data Address High	24-27		Address of response buffer.
Data Address Low	28-31		

Table 38-200 lists the response buffer for the Get Switch Configuration command.

Table 38-200.Get Switch Configuration Response (Opcode: 0x0200) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0200	Command opcode.
Datalen	4-5		Length of response buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Next SEID	16-17	0x0	If not zero, indicates that not all the configuration was returned and a new command should be sent with this value in the First SEID field.
Reserved	18-23	0x0	Must be zero.

Table 38-200.Get Switch Configuration Response (Opcode: 0x0200) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Data Address High	24-27		Address of response buffer.
Data Address Low	28-31		

Table 38-201 lists the structure of the response buffer for the Get Switch Configuration command.

Table 38-201.Get Switch Configuration Command Response Buffer

Offset (bytes)	Description
0-15	Header. See Table 38-202 for details
16-31	Switch element #1. See Table 38-203 for details.
....	
16*n+15-16*n	Switch element #n. See Table 38-203 for details.

Table 38-202.Get Switch Configuration Command Response Header

Offset (bytes)	Description
0-1	Number of switching elements in the structure. The size of the buffer is 16 * (number of elements + 1) If the buffer size is too small to return all the switching elements, only the elements fitting the buffer is returned and the software device driver might request the subsequent elements using a different first element. The maximum number of entries that can be returned in a single command is 255.
2-3	Total number of switching elements. The total number of switching elements. This might be larger than the number of elements in the structure.
4-15	Reserved.

Table 38-203.Get Switch Configuration (Switch Element) (Sheet 1 of 2)

Offset (bytes)	Description	Notes
0	Element Type. Listed in Table 38-191 and Table 38-192.	
1	Revision. Describes the revision of the element type. For the 10 GbE controller, the revision is always 1.	
2-3	SEID. Defines the switch element ID of this element.	
4-5	Uplink Connection. Indicates the SEID of the switching element connected directly below the current element. Below means towards the network in this case.	For sub-elements of a composed switch element, the uplink is the uplink of the entire switch element.
6-7	Downlink Connection. Indicates the SEID of the switching element connected directly above the current element. Above means towards the host in this case.	The Downlink of a composed element is the default port of this element.
8-10	Reserved.	Reserved.
11	Connection Type. 0x0 = Reserved. 0x1 = Regular Data Port. 0x2 = Default Port. 0x3 = Cascaded Port Virtualizer port. 0x4:0xFF = Reserved.	Defines the type of connection to the uplink element.
12-13	Reserved.	

Table 38-203.Get Switch Configuration (Switch Element) (Sheet 2 of 2)

Offset (bytes)	Description	Notes
14-15	Element Specific information. Provides additional information according to the element type as follows: **Element type** **Information.** MAC (1) The physical port number. PF (2) The physical function number. VF (3) The virtual function number. VSI (19) The VSI number. This field is reserved for all other element types.	

Add Statistics (0x0201)

The 10 GbE controller supports 128 smonVlanStats counters. This command is used to allocate a set of smonVlanStats counters to a specific VLAN in a specific switch.

Table 38-204.Add Statistics Command (Opcode: 0x0201)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0201	Command opcode.
Datalen	4-5	0x0	Length of buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Switch SEID	16-17		Defines the SEID of the switch for which the stats are requested.
VLAN ID	18-19		The VLAN ID for which the statistics are requested. 15:12 = Reserved. 11:0 = VLAN ID.
Statistic Counter	20-21	Statistic counters index	Zeroed by driver, firmware returns an index of the statistics counters block assigned to this VLAN.
Reserved	22-31	0x0	Reserved.

Table 38-205.Add Statistics Response (Opcode: 0x0201)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0201	Command opcode.
Datalen	4-5	0x0	Length of buffer.
Return Value/ VFID	6-7		Return value. The following error values can be returned: ENOENT - If the SEID does not point to a valid switch. ENOSPC - If there are not enough resources to assign a statistics block.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Switch SEID	16-17		Copied from command.
VLAN ID	18-19		Copied from command.
Statistic Counter	20-21	Statistic Counters Index	Zeroed by driver, firmware returns an index of the statistics counters block assigned to this VLAN.
Reserved	22-31	0x0	Reserved.

Remove Statistics (0x0202)

The 10 GbE controller supports 128 smonVlanStats counters. This command is used to deallocate a set of smonVlanStats counters from a specific VLAN in a specific switch.

Table 38-206.Remove Statistics Command (Opcode: 0x0202)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0202	Command opcode.
Datalen	4-5	0x0	Length of buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Switch SEID	16-17		Defines the SEID of the switch for which the stats are currently allocated.
VLAN ID	18-19		The VLAN ID for which the statistics are currently allocated. 15:12 = Reserved. 11:0 = VLAN ID.
Statistic Counter	20-21	Statistic Counters Index	The statistics counters block assigned to this VLAN.
Reserved	22-31	0x0	Reserved.

Table 38-207.Remove Statistics Response (Opcode: 0x0202)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0202	Command opcode.
Datalen	4-5	0x0	Length of buffer.
Return Value/ VFID	6-7		Return value. The following error values can be returned: ENOENT - If the SEID does not point to a valid switch. EINVAL - If the statistic block mentioned is not pointed by this VLAN, switch combination.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-31		Reserved.

Set Port Parameters (0x0203)

This command is used to define the default parameters of a physical port.

Table 38-208.Set Port Parameters Command (Opcode: 0x0203) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0203	Command opcode
Datalen	4-5	0x0	Length of buffer
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.

Table 38-208.Set Port Parameters Command (Opcode: 0x0203) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Command Flags	16-17	Bitfield	0 = Save Bad Packets. If set, packets with errors are forwarded to the bad frames VSI. 2 = Enable Double VLAN. If set, this port expects double VLAN packets. The remainder of the bits are reserved.
Bad Frames SEID	18-19		Defines the SEID of the VSI to which bad frames are forwarded. Bit 15 = Valid SEID. Ignored in this field. Bit 14:10 = Reserved. Bit 9:0 = SEID number of the VSI. Relevant only if Command Flags.Save Bad Packets is set.
Reserved	20-31	0x0	Reserved.

Table 38-209.Set Port Parameters Response (Opcode: 0x0203)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0203	Command opcode.
Datalen	4-5	0x0	Length of buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware. The following response might be returned by this command: EPERM - If the operation is not permitted.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command,
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Command Flags	16-17	Bitfield	
Bad Frames VSI	18-19		
Default Switch ID	20-21		returns the switch ID assigned to the port.
Reserved	22-31	0x0	Reserved.

Get Switch Resources Allocation (0x0204)

This command is used to query the resources allocated to a function.

Table 38-210.Get Switch Resources Allocation Command (Opcode: 0x0204)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0204	Command opcode.
Datalen	4-5		Length of response buffer. Should be big enough to contain the expected response buffer size.
Return value/VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-23	0x0	Reserved.
Data Address high	24-27		Return buffer address.
Data Address low	28-31		

Table 38-211.Get Switch Resources Response (Opcode: 0x0204)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0204	Command opcode.
Datalen	4-5		Length of buffer.
Return Value/ VFID	6-7		Return value. There are no specific errors to this command.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Number Of Entries	16	0xD/0x14[1]	Number of resource entries.
Reserved	17-23	0x0	Reserved.
Data Address High	24-27		Return buffer address.
Data Address Low	28-31		

Notes:

1. In cloud images (*Features Enable.Switching mode* is set to Cloud or UDP cloud, then the relevant filters are also returned.

The return buffer contains structures of 16 bytes for each resource type of the following format. The number of entries is returned in the number of entries field in the response.

Table 38-212.Return Buffer Structures for Resource Type Format (Sheet 1 of 2)

Name	Bytes.Bits	Description
Resource Type	0	0x0 = VEBs. 0x1 = VSIs. 0x2 = Perfect match MAC addresses. 0x3 = S-tags. 0x4 = Reserved. 0x5 = Multicast hash entries. 0x6 = Unicast hash entries. 0x7 = VLANs. 0x8= VSI List entries (3 VSI in each entry). 0x9 = Reserved. 0xA = VLAN statistic pools. 0xB = Mirror rules. 0xC = Queue sets. 0xD = Inner VLAN forward filters. 0xE = Reserved. 0xF = Inner MACs. 0x10 = IPs. 0x11 = GRE/VN1 keys. 0x12 = VN2 keys. 0x13 = Tunneling ports. 0x14:0xFF = Reserved.
Reserved	1	Reserved
Guaranteed Allocation	2-3	Number of resources from this type guaranteed to this function. For example, the value set in NVM for this PF and this resource type in the SR PF Allocations NVM section.
Total	4-5	Total number of resources from this type available to all functions (physical size of table).
Currently Used	6-7	Number of guaranteed resources from this type used by this function.
Total Un-allocated	8-9	Total number of resources from this type still un-allocated and not reserved by any function (Number of free shared resources currently available plus unallocated dedicated resources of PF).

Table 38-212.Return Buffer Structures for Resource Type Format (Sheet 2 of 2)

Name	Bytes.Bits	Description
Reserved	10-15	Reserved.

Set Switch Configuration Command (0x0205)

This command is used to set device wide switch configurations. The last configuration of any driver is used and overrides previous settings, apart from the Enable Automatic ATR Eviction, which is set if any function requests it.

Table 38-213.Set Switch Configuration command (Opcode: 0x0205)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0205	Command opcode.
Datalen	4-5	0x0	Length of buffer.
Return value/VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Flags	16-17	0x0	0: Promiscuous behavior: • 0: Packets are forwarded according to promiscuous filter even if matching an exact match filter. • 1: Packets are forwarded according to promiscuous filter only if not matching an exact match filter. 1: Enable L2 filtering. If set, the L2 filter table should pass in addition to the regular forwarding decision. 2: Enable Automatic ATR Eviction. 15:3: Reserved.
Valid Flags	18-19	0x0	0: Promiscuous behavior valid. 1: Enable L2 filtering valid. 2: Enable Automatic ATR Eviction is valid. 15:3: Reserved.
Reserved	20-31	0x0	Reserved.

Table 38-214.Set Switch Configuration response (Opcode: 0x0205)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0205	Command opcode.
Datalen	4-5	0x0	Length of buffer.
Return value/VFID	6-7		Return value.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-31		Reserved.

38.21.7.18.4 VSI Commands (Opcode 0x021x)

Add VSI (0x0210)

This command is used to add a new VSI. A VSI must connect to an existing switching element. A function can connect a VSI only to a switching element it controls.

When an Add VSI command is received, the internal firmware checks if all the requested resources are available and allocate them to the VSI. If part of the resources are not available, the command returns a list of unavailable resources

Table 38-215. Add VSI Command (Opcode: 0x0210)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0210	Command opcode.
Datalen	4-5	0x80	Length of buffer.
Return Value/ VFID	6-7	0x0	Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Uplink SEID	16-17	0x0	Defines the uplink SEID to which this VSI should be connected.
Connection Type	18	0x0	Defines to which port of the uplink element this VSI connects. 0x0 = Reserved. 0x1 = Regular data port. 0x2 = Default port. 0x3:0xFF = Reserved. ***Note:*** If set as default port, replaces the existing default port of the underlying switching element.
Reserved	19	0x0	Reserved
VF Function Number	20		Defines the VF function to which this VSI connects. Valid only if function type is VF. Should be ignored if VSI type is not VF. The VF number here is the absolute VF number (0-127) and not the number relative to the PF first VF.
Reserved	21		Reserved.
Command Flags	22-23		1:0: VSI type. 0 = VF. 1 = VMDq2 (a.k.a. VM). 2 = PF. 3 = EMP/MNG. 2: Cascaded port virtualizer. The S-tag manipulation commands can be used only if this bit is set. Bit 15:3 = Reserved.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

Table 38-216 lists the structure of the command buffer for the Add VSI command.

Table 38-216.Add VSI Command Buffer (Sheet 1 of 5)

Category	Byte/ Bit	Field	Description
Valid Sections	0-1	Defines which sections are valid in the command.	0 = Switching section is valid. Must be set for *Add VSI* commands when containing SEID is a port virtualizer. 1 = Security section is valid. 2 = VLAN handling section is valid. 3 = Cascaded port virtualizer section is valid. 4 = Ingress UP translation section is valid. 5 = Egress UP translation section is valid. 6 = Queue mapping section is valid. Must be set for *Add VSI* commands. If set for *Update VSI* command, modified queues must be disabled. 7 = Queuing option section is valid. 8 = Outer UP translation section is valid. Should not be set if inner UP to outer UP mapping is the identity mapping. 9 = Scheduler section is valid. This bit is ignored in *Update VSI* command. 10:15 = Reserved.
Switching	2-3.3	Switch ID	Defines the switch ID to which this VSI belongs. If it is not, S-tag is cleared, then this switch ID is an S-tag; otherwise, it is a virtual switch ID. If the VSI is connected to a port virtualizer, this value must be set. For a port connected to a VEB or directly to the MAC, this value is not an S-tag field and might be left at zero and the VEB switch ID or the default switch ID is used. For the Update VSI command it should be a valid tag (could be zero for VEB).
	3.4	Is not S-tag	See description of the switch ID parameter for the usage of this bit.
	3.5	Allow loopback	This bit should be set for VSIs that are connected to a VEB and should be cleared otherwise. A VSI connected to a VEB might have this bit cleared if used only for connection to the link (such as a management VSI). Cleared if section not valid.
	3.6	Allow local loopback.	This bit should be set for VSIs that are used as uplink of a software (cascaded) VEB, VEPA or port virtualizer. Cleared if section not valid.
	3.7-5.7	Reserved.	Reserved for future switching parameters. Must be zero.
Security	6.0	Allow destination override.	Allow the VSI to override the switching decision and fix the destination of a transmit packet. This bit should be set only for control ports. Cleared if section not valid.
	6.1	Enable VLAN anti-spoof	Cleared if section not valid. Enabling this mode might impact the transmit performance. It is recommended to use only the MAC anti-spoof mode.
	6.2	Enable MAC anti-spoof.	Cleared if section not valid.
	6.3 - 7.7	Reserved.	Reserved for future security parameters. Must be zero.

Table 38-216.Add VSI Command Buffer (Sheet 2 of 5)

Category	Byte/ Bit	Field	Description
VLAN Handling	8-9	PVID+ Default UP (16 bits).	VLAN ID to use in port-based VLAN insertion. This field is relevant only if the *Insert PVID* field is set. Cleared if section not valid.
	10-11	Reserved.	Reserved.
	12.0:1	VLAN driver insertion mode.	This field defines if the software device driver is allowed/should add a VLAN tag to the packets it sends. If *Insert PVID* field is set, this field should be set to 01b. 00b = Reserved. 10b = Admit.1Q tagged only. Allow only packets with VLAN. 01b = Admit untagged/Priority tagged only. Allow only packets without VLAN or with VLAN tag = zero. 11b = Allow all packets. If section not valid, the default value is 11b.
	12.2	Insert PVID.	Port-based VLAN insertion. This bit controls the port-based insertion of VLANs. Should be set for VFs/VMDq2 VSIs according to the VMM request. If this field is set, the *PVID* + *Default UP* field should be set to the port-based VLAN. Cleared if section not valid.
	12.3:4	VLAN and UP expose mode (Rx).	This field defines how received VLANs are handled. For non-VF VSIs, 00b or 11b should be used. For VF VSIs, the mode should be set according to the VLAN awareness of the VM and the offload requested. 00b = Show VLAN and UP in descriptor (legacy behavior). 01b = Hide VLAN show UP (VLAN ID = 0). 10b = Hide VLAN and UP. 11b = Do nothing (leave VLAN in packet). If section not valid, the default value is 11b.
	12.5-15.7	Reserved.	Reserved for future port VLAN parameters. Must be zero.
Ingress UP Translation	16-19	Ingress UP translation table.	Defines the UP translation table for received packets according to the following list: 16.0:16.2 UP set if received UP is 0. 16.3:16.5 UP set if received UP is 1. 16.6:17.0 UP set if received UP is 2. 17.1:17.3 UP set if received UP is 3. 17.4:17.6 UP set if received UP is 4. 17.7:18.1 UP set if received UP is 5. 18.2:18:4 UP set if received UP is 6. 18.5:18.7 UP set if received UP is 7. 19: Reserved. This map is used to translate the 802.1P user priority bits received in the packet to the user priority exposed to the host. Relevant only if the UP is exposed to the host (VLAN and UP expose mode not equal 10b). If section is not valid, the default value is identity mapping.

Table 38-216.Add VSI Command Buffer (Sheet 3 of 5)

Category	Byte/ Bit	Field	Description
Egress UP Translation	20-23	Egress UP translation table.	Defines the UP translation table for transmit packets according to the following list: 20.0:20.2 UP set if sent UP is 0. 20.3:20.5 UP set if sent UP is 1. 20.6:21.0 UP set if sent UP is 2. 21.1:21.3 UP set if sent UP is 3. 21.4:21.6 UP set if sent UP is 4. 21.7:22.1 UP set if sent UP is 5. 22.2:22.4 UP set if sent UP is 6. 22.5:22.7 UP set if sent UP is 7. 23: Reserved. This map is used to translate the 802.1P user priority bits sent by the host to the i user priority sent to the network. Note that the resulting user priority is further translated using the per TC translation table. If section is not valid, the default value is identity mapping.
Cascaded Port Virtualizer	24-25	S-tag.	The S-tag (S-VID) to be inserted in transmit packets. If *Switch ID* parameter is set and is an S-tag and not a cascaded port virtualizer, should be set to the same value as the *switch ID* parameter. This field is relevant only if *S-tag insert enable* field is set. This field is 16 bits and should include also the default PCP priority bits to insert in the S-tag. Cleared in SFP mode.
	26.0:1	S-tag extract mode.	This field defines how received S-tags are handled. 00b = Do nothing (cascaded port virtualizer without offload). 01b = Extract tag and do not insert in descriptor (regular VM). 10b = Extract tag from packet and expose in descriptor (cascaded port virtualizer with offload). 11b = Reserved. If section not valid, the default value is 00b in SFP mode.
	26.2:3	Reserved.	Reserved.
	26.4	S-tag insert enable.	This bit controls the port-based insertion of S-tags. Should be set, if VSI is part of an S-channel and is not a cascaded port virtualizer VSI. If the section is not valid, the default value is cleared in SFP mode. When this bit is set, the *Accept tag from host* bit should be cleared.
	26.5	Reserved.	Reserved.
	26.6	Accept tag from host.	Allow host to insert an S-tag in a descriptor or in a packet. Should be set only for cascaded port virtualizer or control ports. Set in SFP mode if section not valid. When this bit is set, the *S-tag insert enable* bit should be cleared.
	26.7-27.7	Reserved.	Reserved for future port S-tags parameters. Must be zero.

Table 38-216.Add VSI Command Buffer (Sheet 4 of 5)

Category	Byte/ Bit	Field	Description
Queue Mapping	28.0	Mapping method.	Selects between contiguous range of queues for this VSI vs. scattered range. 0b = The VSI is assigned a contiguous range of PF queues. 1b = The VSI is assigned a scattered range of PF queues.
	28.1 - 29	Reserved.	Reserved. Must be zero.
	30-61	Queue mapping.	If mapping method = 0 (contiguous): 30.0:31.2 = The first queue allocated for this VSI in the PF space. This VSI can access all the queues from this queue to the end of the PF queues allocation, but is bounded by the receive queues per TC configuration. 31.3:61 = Reserved. If mapping method = 1 (scattered). For each of the queues in the VSI, defines the actual queue in the PF according to the following encoding: For queue 'n' of the VSI offsets 30+2n - 31+2n are used the define the mapping to PF queues. For example for queue 0: 30.0:31.2 = The PF queue matching allocated to queue 'n' of the VSI. For non-allocated queue, a value of 0x7FF should be set. 31.3:31.7 = Reserved. The same mapping is used for the next queues. In this method, up to 16 queues can be assigned. For VSIs assigned to VFs, only the scattered method can be used. The first queue (in both modes) is the default queue of the VSI to which packets not queued by any filter is sent.
	62-77	Number and offset of queue pairs per TCs.	Fixes the number of queue pairs assigned to the VSI for each TC and the offset of these queues. 62.0 - 63.0 = Queue offset for TC0. 63.1 - 63.3 = Number of queues allocated to TC0. The actual number is 2^n. The allowed number of queues are: 1; 2; 4; 8; 16; 32; 64; 128. 63.4 - 63.7 = Reserved. If no queues needs to be associated to a TC, the queue offset should be set to 0 and the number of queues to 0 (1 queue). This way, traffic associated with this TC is sent to the default queue. The following addresses are used with the same format for the next TCs: 64:65 = TC1. 66:67 = TC2. 68:69 = TC3. 70:71 = TC4. 72:73 = TC5. 74:75 = TC6. 76:77 = TC7.
Queueing Options	78.0:1	Reserved.	Reserved.
	78.2	Multicast UDP packets enable.	Enable multicast UDP packets by the quad hash filter for the VSI. Cleared if section not valid.
	78.3	Unicast UDP packets enable.	Enable unicast UDP packets by the quad hash filter for the VSI. Cleared if section not valid.
	78.4	TCP packets enable.	Enable TCP packets by the quad hash filter for the VSI. Should be set for iWARP VSIs. Cleared if section not valid.

Table 38-216.Add VSI Command Buffer (Sheet 5 of 5)

Category	Byte/ Bit	Field	Description
	78.5.	Reserved.	Reserved.
	78.6	RSS LUT.	RSS LUT used. 0b = Use PF LUT 1b = Use VSI LUT. If section not valid, PF LUT is used for PF VSIs and VSI LUT for VFs VSI. A VF VSI should use the VSI LUT.
	78.7 - 81	Reserved.	Reserved for future queueing parameters. Must be zero.
Scheduler	82	Enabled TCs.	A bitmap indicating which TCs are enabled in this VSI. If section not valid, only TC0 is enabled.
	83	Reserved.	Reserved.
Outer UP Mapping	84-87	Egress inner UP to outer translation table.	Defines the UP translation table for transmit packets from inner-to-outer UP according to the following list: 84.0:84.2 Outer UP set if inner UP is 0. 84.3:84.5 Outer UP set if inner UP is 1. 84.6:85.0 Outer UP set if inner UP is 2. 85.1:85.3 Outer UP set if inner UP is 3. 85.4:85.6 Outer UP set if inner UP is 4. 85.7:86.1 Outer UP set if inner UP is 5. 86.2:86.4 Outer UP set if inner UP is 6. 86.5:86.7 Outer UP set if inner UP is 7. 87: Reserved. This map is used to translate the 802.1P user priority bits on the inner UP-to-outer UP. If section is not valid, the default value is identity mapping. In order to get a fixed outer UP, all values should be set to the requested outer UP.
Reserved		Reserved.	Reserved.
Response Space	96-127	Reserved.	Reserved for response space.

Table 38-217 and Table 38-218 list the add VSI response and the response buffer.

Table 38-217.Add VSI Response Buffer

Category	Byte/Bit	Field	Description
Command Space	0-95	Reserved	Reserved for command space. Should contain the values provided by the software device driver.
Queue Set Handles	96-97	QS_Handle 0.	The handle for queue set of TC0. Bits [9:0] of this handle are used by software to program the *RDYList* field in the transmit queues context for queues associated with TC0.
	98-99	QS_Handle 1.	The handle for queue set of TC1. Same format as QS_Handle 0.
	100-101	QS_Handle 2.	The handle for queue set of TC2. Same format as QS_Handle 0.
	102-103	QS_Handle 3.	The handle for queue set of TC3. Same format as QS_Handle 0.
	104-105	QS_Handle 4.	The handle for queue set of TC4. Same format as QS_Handle 0.
	106-107	QS_Handle 5.	The handle for queue set of TC5. Same format as QS_Handle 0.
	108-109	QS_Handle 6.	The handle for queue set of TC6. Same format as QS_Handle 0.
	110-111	QS_Handle 7.	The handle for queue set of TC7. Same format as QS_Handle 0.
Statistic Counter	112-113	Statistic counters index.	Returns an index of the statistics counters block assigned to this VSI.
Reserved	116-127	Reserved.	Reserved.

Table 38-218.Add VSI Response (Opcode: 0x0210) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0210	Command opcode.
Datalen	4-5	0x80	Length of buffer.
Return Value/ VFID	6-7		Return value. The following error values can be returned: ENOENT - If the uplink SEID or the function number do not point to valid elements. EINVAL - If the topology created is not valid. EEXIST - If a default port is requested and it already exists. ENOSPC - If there are not enough resources to assign a VSI.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
SEID	16-17		If the return value is zero, returns the SEID of the VSI. If return value is ENOSPC, contains a bitmap that indicates which resources are missing: 0 = No VSI left. 1 = Not enough scheduler nodes. 2 = Not enough statistics counters. 3 = Not enough switching entries. 4:15 = Reserved.
VSI Number	18-19		Returns the assigned VSI number.
VSIs Used	20-21		Number of VSIs used by this function.

Table 38-218.Add VSI Response (Opcode: 0x0210) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Total VSIs Un-allocated	22-23		Total number of VSIs still un-allocated and not reserved by any function.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

Add VSI Settings Recommendations

Table 38-219 lists the recommended settings for common types of VSIs.

Table 38-219.Add VSI Recommended Settings (Sheet 1 of 2)

Category	Field	VSI Type						
		PF	VF	VMDq2	VMDq1	Cascaded VEB	Cascaded S-comp	Floating VEB
Valid Sections	Switching section is valid.	0	1	1	1	1	1	1
	Security section is valid.	1						
	VLAN handling section is valid.	1						
	Cascaded port virtualizer section is valid.	1						0
	Ingress UP translation section is valid.	0						
	Egress UP translation section is valid.	0						
	Queue mapping section is valid.	1						
	Queuing option section is valid.	1						
	Scheduler section is valid.	0 if non-DCB						
Switching	Switch ID.	If the VSI is connected directly to a port virtualizer, this value must be set. For a port connected to a VEB or directly to the MAC, this value might be left at zero and the VEB switch ID or the default switch ID is used. When connecting to a port virtualizer, the switch ID should be set to the S-tag of the channel and is not a S-tag that should be cleared.					Set to the first tag covered by this S-comp.	VEB switch ID
	Is not S-tag.						0	1
	Allow loopback.	0	1 if VEB 0 if VEPA		1 if part of a VEB, 0 otherwise [1]	1 if VEB 0 if VEPA	0	1
	Allow local loopback.	0	0	0	1 if part of a VEB, 0 otherwise	1 if VEB 0 if VEPA	0	0
Security	Allow destination override.	1	0	1	1	1	1	0/1
	Enable VLAN anti-spoof.	0	1	0	0	0	0 [2]	0/1
	Enable MAC anti-spoof.	0	1	0	0	0	0	0/1

Table 38-219.Add VSI Recommended Settings (Sheet 2 of 2)

Category	Field	VSI Type						
		PF	VF	VMDq2	VMDq1	Cascaded VEB	Cascaded S-comp	Floating VEB
VLAN Handling	PVID+ default UP (16 bits).	N/A	Port-based VLAN	N/A	N/A	N/A	N/A	N/A
	VLAN driver insertion mode.	11b	Depends on VLAN awareness	11b	11b	11b	11b	11b
	Insert PVID.	0		0	0	0	0	0
	VLAN and UP expose mode (Rx).	00b (extract and show)		00b (extract and show)	00b (extract and show)	00b (extract and show)	00b (extract and show)	00b (extract and show)
Ingress UP Translation	Ingress UP translation table.	Identity mapping (default).						
Egress UP Translation	Egress UP translation table.	Identity mapping (default).						
Cascaded Port Virtualizer	S-tag.	Same as Switch ID if Port Virtualizer is part of the switch, N/A otherwise.					N/A	Switch ID of the floating VEB
	S-tag extract mode.	01b (Extract S-tag and do not insert in descriptor)					10b (extract S-tag from packet and expose in descriptor).	01b
	S-tag insert enable.	1 if port virtualizer is part of the switch topology, 0 otherwise.					0	1
	Prune based on internal S-tag enable.	1					0	1
	Accept tag from host.	0					1	0
Queue Mapping	Mapping method.	Depends on queue mapping method.						
	Queue mapping.							
	Number and offset of receive queues per TCs.	According to allocation to TCs.						
	Fragmented multicast UDP enable	0 unless used for UDA					0	0 unless used for UDA
	Fragmented unicast UDP enable	0 unless used for UDA					0	
	Multicast UDP packets enable	0 unless used for UDA					0	
	Unicast UDP packets enable	0 unless used for UDA					0	
Queueing Options	TCP packets enable.	1						
	RSS LUT.	PF	VSI	VSI	N/A	N/A	N/A	VSI
Scheduler	Enabled TCs.	According to allocated TCs (0x1 for non-DCB environment).						

Notes:

1. Different VSIs within a VEB may have different loopback modes.
2. VLAN anti-spoof does not work in cascaded S-comp mode.

Update VSI (0x0211)

This command is used to update the parameters of an existing VSI. A function can update only a VSI it controls. Not all the parameters of a VSI can be updated. Only parameters that do not impact the scheduler tree structure can be modified. In order to change the traffic classes allocated to a VSI, the *Configure VSI Bandwidth per Traffic Class* command should be used.

Table 38-220.Update VSI Command (Opcode: 0x0211)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0211	Command opcode.
Datalen	4-5	0x80	Length of buffer.
Return Value	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
SEID Number	16-17		VSI SEID number.
Reserved	18-21		Reserved.
Command Flags	22-23		15:0 = Reserved.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

The command buffer and the completion buffer used to update a VSI is the same as the command and completion buffers of the Add VSI command (Table 38-216 and Table 38-217). Specific limitations are listed in the tables.

All the filters set before the VSI type updates are kept, and the software device driver should guarantee they are compliant with the new VSI type.

Table 38-221.Update VSI Response (Opcode: 0x0211)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0211	Command opcode.
Datalen	4-5	0x80	Length of buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware. The following error values can be returned: ENOENT - If the SEID do not point to a valid VSI. EACCES - If the VSI is not owned by this PF.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
SEID Number	16-17	0x0	Copied from command.
VSI Number	18-19		Returns the assigned VSI number.
VSI Used	20-21		Number of VSIs used by this function.
Total VSIs Un-allocated	22-23		Total number of VSIs still un-allocated and not reserved by any function.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

Get VSI Parameters (0x0212)

This command is used to get the parameters of an existing VSI. A function can query only a VSI it controls.

Table 38-222.Get VSI Parameters Response (Opcode: 0x0212)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0212	Command opcode.
Datalen	4-5	0x80	Length of buffer.
Return Value/VFID	6-7		Return value. Zeroed by driver. Written by firmware. The following error values can be returned: ENOENT - If the SEID do not point to a valid VSI. EACCES - If the VSI is not owned by this PF.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
SEID Number	16-17		VSI SEID number (copied from command).
VSI Number	18-19		Returns the assigned VSI number.
VSIs Used	20-21		Number of VSIs used by this function.
Total VSIs Un-allocated	22-23		Total number of VSIs still un-allocated and not reserved by any function.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

Table 38-223.Get VSI Parameters Command (Opcode: 0x0212)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0212	Command opcode.
Datalen	4-5	0x80	Length of buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
SEID Number	16-17		VSI SEID number.
Reserved	18-23		Reserved.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

Table 38-224 lists the response buffer received when querying a VSI.

Table 38-224.Get VSI Parameters Response Buffer

Byte/Bit	Description
0-95	Same parameters as listed in Table 38-216.
96-127	Same parameters as listed in Table 38-217.

38.21.7.18.5 Port Virtualizer Commands (opcode 0x022x)

Add Port Virtualizer

This command is used to instantiate a port virtualizer on a port. A port virtualizer can be instantiated only when a single VSI is connected between the MAC and a PF. If additional switching elements are active on this port, they must be first disconnected before the port virtualizer can be added.

When an Add Port Virtualizer command is received, the internal firmware checks if all the requested resources are available and allocate them to the port virtualizer. If part of the resources are not available, the command returns a list of unavailable resources

Table 38-225.Add Port Virtualizer Command (Opcode: 0x0220)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0220	Command opcode.
Datalen	4-5	0x0	Length of buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Command Flags	16-17	0x0	0 = Port virtualizer type. Must be set to zero. 1 = Forward unknown S-tag. If set, packets with unknown S-tags are forwarded to the default VSI. *2* = Reserved. If bit 1 is set, the port type should be default; otherwise, these bits are ignored. 3 = Port type: 0 = Default. 1 = Control. 4:15 = Reserved.
Uplink SEID	18-19	0x0	Defines the SEID to which this port virtualizer should be connected. This can be only the MAC of the port on which this function resides.
Connected VSI SEID	20-21	0x	Defines the SEID of the first VSI connected to this port virtualizer. Port type is defined by the *Port Type* flag. Should point to a valid VSI, connected to the uplink SEID as uplink.
Reserved	22-31		Reserved.

Table 38-226.Add Port Virtualizer Response (Opcode: 0x0220)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0220	Command opcode.
Datalen	4-5	0x0	Length of buffer.
Return Value/ VFID	6-7		Return value. The following error values can be returned: ENOENT - If the port SEID do not point to a valid element. ENOSPC - If there are not enough resources to assign a port virtualizer. ESRCH - If the operation is not permitted (like virtualization is disabled). EACCES - If the port is not owned by this PF. EINVAL - If a software device driver tries to create a port virtualizer of a type contradicting the previously created port virtualizer type.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Port Virtualizer SEID/Error Reasons	16-17		If the return value is zero: Bytes 25:24 = Returns the SEID of the port virtualizer. If the return value is ENOSPC, it contains a bitmap that indicates which resources are missing: 0 = No port virtualizer left. 1 = Not enough scheduler nodes. 2 = Reserved. 3 = Not enough switching entries. 4:31 = Reserved.
Reserved	18-31		Reserved.

Update Port Virtualizer Parameters

This command is used to modify the parameters of a port virtualizer. In this command, the flags of the port virtualizer can be modified. Note that the default VSI cannot be modified using this command, only the routing of unknown packets.

In order to enable forwarding of unknown packets, a default VSI should be defined beforehand.

Table 38-227.Update Port Virtualizer Command (Opcode: 0x0221)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0221	Command opcode.
Datalen	4-5	0x0	Length of buffer.
Return Value	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Command Flags	16-19	0x0	0 = Reserved (the port virtualizer type cannot be modified on the fly). 1 = Forward unknown S-tag. If set, packets with unknown S-tags are forwarded to the default VSI. 2-31 = Reserved.
Reserved	20-31	0x0	Reserved.

Table 38-228.Update Port Virtualizer Response (Opcode: 0x0221)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0221	Command opcode.
Datalen	4-5	0x0	Length of buffer.
Return value	6-7		The following error values can be returned: ENOENT - If the SEID do not point to a port virtualizer. EACCES - If the port virtualizer is not owned by this PF.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-31		Reserved.

Get Port Virtualizer Parameters

This command is used to get the parameters of a port virtualizer.

Table 38-229.Get Port Virtualizer Command (Opcode: 0x0222)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0222	Command opcode.
Datalen	4-5	0x0	Length of buffer
Return Value	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
SEID	16-17		The SEID of the port virtualizer.
Reserved	18-31		Reserved.

Table 38-230.Get Port Virtualizer Response (Opcode: 0x0222)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0222	Command opcode.
Datalen	4-5	0x0	Length of buffer.
Return Value/ VFID	6-7		The following error values can be returned: ENOENT - If the SEID do not point to a port virtualizer. EACCES - If the port virtualizer is not owned by this PF.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-17		The SEID of the requested port virtualizer (reflects the command).
Reserved	18-19		Reserved.
Command Flags	20-21	0x0	0 = Port virtualizer type: 0 = S-comp. 1 = Port extender. 1 = Forward unknown S-tag. If set, packets with unknown S-tags are forwarded to the default VSI. 2:15 = Reserved.
Reserved	22-29	Reserved	Reserved.
Default Port SEID	30-31		Returns the SEID of the default port.

38.21.7.18.6 VEB/VEPA Commands (Opcode 0x023x)

Add VEB

This command is used to instantiate an VEB on a port/port virtualizer. A VEB can be instantiated either as a floating element or can be inserted between an existing VSI and its uplink.

When an Add VEB command is received, the internal firmware checks if all the requested resources are available and allocate them to the VEB. If part of the resources are not available, the command returns a list of unavailable resources.

Note: After a VEB is added, the *Allow Loopback* flag of the original VSI should be set using the Update VSI command.

Table 38-231.Add VEB Command (Opcode: 0x0230)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0230	Command opcode.
Datalen	4-5	0x0	Length of buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
uplink SEID	16-17	0x0	Defines the SEID to which this VEB should be connected. If the *Floating VEB* flag is set, the uplink element should be null (0).
Downlink SEID	18-19	0x0	Defines the SEID of the downlink VSI to which this VEB is connected. Port type is defined by flags. Should point to valid VSI, currently connected to uplink SEID. Should be null(0) for floating VEB.
Flags.Floating VEB	20.0	0x0	If set it can only send packets to attached VSIs; if cleared it can send and receive packets from the LAN.
Flags. Port Type	20.2:-20.1	0x0	Defines the type of the VSI defined by the Downlink SEID field. 00b = Reserved. 01b = Default. 10b = Data Port. 11b = Reserved. Should be default (01b) for floating VEB.
Flags. Reserved	20.3	0x0	Reserved.
Flags. Disable Statistics Gathering	20.4	0x0	Disable Statistics Gathering. 0b = Statistics are gathered for this VEB. 1b = Statistics are not gathered for this VEB.
Flags. Reserved	20.5:21	0x0	Reserved.
Enabled TCs	22	0x0	A bitmap of the TCs that should be enabled in this VEB. The TCs enabled should be already enabled in the uplink element (should already have a node in the scheduler in the uplink element).
Reserved	23-31	0x0	Reserved

Note: The enable L2 filtering flag should be set to the same values for all VEB instances in the device. Firmware uses the value set by the first *Add VEB* AQ and rejects subsequent *Add VEB* AQ with a conflicting setting with an EINVAL error.

Table 38-232.Add VEB Response (Opcode: 0x0230)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0230	Command opcode.
Datalen	4-5	0x0	Length of buffer.
Return Value/ VFID	6-7		Return value. The following error values can be returned: ENOENT - If the uplink SEID do not point to valid elements. ENOSPC - If there are not enough resources to assign a VEB. EACCES - If the uplink element is not owned by this PF. ERANGE - A TC not enabled in the uplink element was requested. EPERM - If the command is not permitted (such as virtualization is disabled).
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-21		Reserved.
Switch ID	22-23		Assigned switch ID. If connected to an S-channel, this is equal to the S-tag; otherwise, it is the switch ID assigned internally by firmware.
VEB SEID/Error Reasons.	24-25		If the return value is zero: Bytes 25:24 = Returns the SEID of the VEB. If return value is ENOSPC, contains a bitmap that indicates which resources are missing: 0 = No VEB left. 1 = Not enough Scheduler nodes. 2 = Not enough statistics counters. 3 = Not enough switching entries. 4:31 = Reserved.
Statistic Counter	26-27	Statistic Counters Index	Returns an index of the statistics counters block assigned to this VEB. This number is in the 0-15 range and points to the VEB statistics set. Relevant only if *Disable Statistics Gathering* flag is cleared in the command.
VEBs Used	28-29		Number of VEBs used by this function.
Total VEBs Un-allocated	30-31		Total number of VEBs still un-allocated and not reserved by any function.

Get VEB Parameters (0x0232)

This command returns the parameters of the VEB/VEPA

Table 38-233.Get VEB Parameters Command (Opcode: 0x0232)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0232	Command opcode.
Datalen	4-5	0x0	Length of buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
SEID	16-17	0x0	Defines the SEID of the VEB.
Reserved	18-31	0x0	Reserved.

Table 38-234.Get VEB Parameters Response (Opcode: 0x0232)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0232	Command opcode.
Datalen	4-5	0x0	Length of buffer.
Return Value/ VFID	6-7		Return value. The following error values can be returned: ENOENT - If the SEID do not point to a valid VEB/VEPA element. EACCES - If the VEB is not owned by this PF. EPERM - If the command is not permitted (such as virtualization is disabled).
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
SEID	16-17		The SEID requested in the command.
Switch ID	18-19		Assigned switch ID. If connected to an S-channel, this is equal to the S-tag; otherwise, it is the switch ID assigned internally by firmware.
Flags	20-21	0x0	0 = Floating VEB. If set, it can only send packets to attached VSIs If cleared it can send and receive packets from the LAN. 2:1 = Reserved. 3 = L2 filtering is enabled. If set, the L2 filter table should pass in addition to the regular forwarding decision. Relevant only for cloud and UDP cloud NVM images. Reserved otherwise. 15:4 = Reserved.
Statistic Counter	22-23	Statistic Counters Index	Returns an index of the statistics counters block assigned to this VEB.
VEBs Used	24-25		Number of VEBs used by this function.
Total VEBs Un-allocated	26-27		Total number of VEBs still un-allocated and not reserved by any function.
Reserved	28-31		Reserved.

38.21.7.18.7Switch Connectivity Commands (Opcode 0x024x)

Delete Element

This command is used to remove a switching element. A function can remove only an element it controls. The software device driver should make sure every queue in a VSI is disabled before a VSI is removed; however, queue groups tied to the VSI can be removed together with the VSI. It should also make sure the removed element is not tied to any other element before removing it. An exception to this rule is the case of a VEB, that when removed with a single VSI tied to it is replaced with this VSI.

Table 38-235.Delete Element Command (Opcode: 0x0243)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0243	Command opcode.
Datalen	4-5	0x0	Reserved.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
SEID	16-17	SEID	The SEID of the element to remove.
Reserved	18-31	0x0	Must be zero.

Table 38-236.Delete Element Response (Opcode: 0x0243)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0243	Command opcode.
Datalen	4-5	0x0	Reserved.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware. The following error values can be returned: ENOENT - If SEID do not point to a valid element. EACCES - If the element is not owned by this PF. EBUSY - If the element to remove has more than one uplink element tied to it.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-31		Reserved.

38.21.7.18.8 Forwarding Table Configuration Commands (Opcode 0x025x)

Note: All the MAC addresses in the forwarding table configuration commands should be given in big endian format.

Add MAC, VLAN Pair (0x0250)

This command is used to add a set of MAC or MAC, VLAN pairs to a set of VSIs. If one of the allocations fail due to lack of resources, the *Set VSI Promiscuous modes* command can be used to enable forwarding of all the packets of a given type to this VSI. See Set VSI Promiscuous Modes (0x0254) for more detail.

All the VSIs must have the same SwitchID.

Enables receiving untagged packets only, a MAC, VLAN = 0 filter should be added.

This command should not be used to forward broadcast packets using hash based filtering. The *Set VSI Promiscuous Modes* command should be used if broadcast forwarding without VLAN filtering is required or using this command with *Use Hash* flag cleared if broadcast filtering with VLAN filtering is needed. See Set VSI Promiscuous Modes (0x0254) for more detail.

Note: The ToQueue action might be ignored if the packet is forwarded due to multiple rules match (such as an exact match and promiscuous unicast).

Table 38-237.Add MAC, VLAN Pair Command (Opcode: 0x0250) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0250	Command opcode.
Datalen	4-5		Length of buffer. Should be equal to Num_addresses * 16 bytes.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Num_addresses	16-17		The number of MAC, VLAN pairs to add.
SEID 0	18-19	0x0	Bit 15 = Valid SEID. Bit 14:10 = Reserved. Bit 9:0 = SEID number of the VSI.

Table 38-237.Add MAC, VLAN Pair Command (Opcode: 0x0250) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
SEID 1	20-21	0x0	Bit 15 = Valid SEID. Bit 14:10 = Reserved. Bit 9:0 = SEID number of the VSI.
SEID 2	22-23	0x0	Bit 15 = Valid SEID. Bit 14:8 = Reserved. Bit 9:0 = SEID number of the VSI.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

The command buffer of this command contains the details of the MAC, VLAN pairs to add. It contains a set of *Num_addresses* 16-byte structures as listed in Table 38-238.

Table 38-238.MAC, VLAN Pairs to Add Detail

Field	Offset	Description
MAC Address	0-5	MAC address to add.
VLAN Tag	6-7	VLAN tag to add. Enables receiving untagged packets only, a VLAN ID of zero should be set and the *Ignore VLAN* flag should be cleared. 23:12 = Reserved. 11:0 = VLAN ID.
Flags	8-9	0: Use perfect match = If set, an perfect match MAC address can be used to create this MAC, VLAN pair. 1: Use Hash = If set, a hash MAC address can be used to create this MAC, VLAN pair. 2: Ignore VLAN = If set, the VLAN tag is ignored and the MAC address is used to forward packets from all VLANs. 3: ToQueue = Use MAC, VLAN to point to a queue. This flag should be set only if the filter points to a single VSI. If a filter points to multiple VSIs, an error is returned. 4: Use shared MAC: If set, the MAC address is taken from the shared pool. 15:5 = Reserved. At least one of the use perfect match or *Use Hash* flags should be set. If both are set, firmware attempts to use perfect match and falls back to using hash if the exact match table is full.
Queue Number		Bit 15:11 = Reserved. Bit 10:0 = Queue number. Valid only if the *Flags.ToQueue*e bit is set. The queue number is relative to the VSI.
Reserved		Reserved for response part.

Table 38-239.Add MAC, VLAN Pair Response (Opcode: 0x0250) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0250	Command opcode.
Datalen	4-5		Length of buffer. equals to Num_addresses * 16 bytes.
Return Value/ VFID	6-7		Return value. The following error values can be returned: ENOSPC - If there are not enough resources to assign all the MAC, VLAN pairs or an attempt to add a VSI to an existing filter with ToQueue set. The return buffer details which of the allocations failed. ENOENT - If the SEID does not point to a valid VSI. EACCES - If the VSI is not owned by this PF. EEXIST - If a queue is assigned to a multicast filter (more than one VSI). EINVAL - If a *ToQueue* flag is set for a multicast filter. ENOMEM - no space for WoL filters.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command

Table 38-239.Add MAC, VLAN Pair Response (Opcode: 0x0250) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Perfect Match MAC Used	16-17		Number of perfect match MAC used by this function.
Perfect Match MAC Un-allocated	18-19		Total number of perfect match MAC still un-allocated and not reserved by any function.
Unicast Hash MAC Un-allocated	20-21		Total number of unicast hash MAC still un-allocated.
Multicast Hash MAC Un-allocated	22-23		Total number of multicast hash MAC still un-allocated.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

The response buffer of this command contains the results of the MAC, VLAN pairs allocation. It contains a set of *Num_addresses* 16-byte structures as listed in Table 38-240.

Table 38-240.MAC, VLAN Pairs Allocation Detail

Field	Offset	Description
Reserved		Reserved for command part.
Matching Method		0x0 = Perfect match was used. 0x1 = Hash match was used. 0x2:0xFE = Reserved. 0xFF = Request failed due to lack of resources.
Reserved		Reserved.

Remove MAC,VLAN Pair (0x0251)

This command is used to remove a set of MAC or a MAC, VLAN pairs from up to three VSIs.

All the VSIs must have the same SwitchID.

If a hash MAC was used, the address should be removed only if this hash value is no longer needed for this VSI(s).

Table 38-241.Remove MAC, VLAN Pair Command (Opcode: 0x0251) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0251	Command opcode.
Datalen	4-5		Length of buffer. Should be equal to Num_addresses * 16 bytes.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Num Addresses	16-17		The number of addresses in the command buffer.
SEID 0	18-19	0x0	Bit 15 = Valid SEID. Bit 14:10 = Reserved. Bit 9:0 = SEID Number of the VSI.

Table 38-241.Remove MAC, VLAN Pair Command (Opcode: 0x0251) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
SEID 1	20-21	0x0	Bit 15 = Valid SEID. Bit 14:10 = Reserved. Bit 9:0 = SEID Number of the VSI.
SEID 2	22-23	0x0	Bit 15 = Valid SEID. Bit 14:8 = Reserved. Bit 9:0 = SEID Number of the VSI.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

The command buffer of this command contains the details of the MAC, VLAN pairs to remove. It contains a set of *Num_addresses* 16-byte structures as listed in Table 38-242.

Table 38-242.Remove MAC, VLAN Pair Response (Opcode: 0x0251)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0251	Command opcode
Datalen	4-5		Length of buffer. Indicates the number of entries filled by firmware (16*Num Addresses) even if the *Datalen* in the command was larger than that.
Return Value/ VFID	6-7		Return value. The following error values can be returned: EINVAL - If all the VSIs do not point to the same switchID. ENOENT- If the MAC, VLAN pair does not exist or if one of the SEID does not point to a valid VSI. The details of which remove request failed is found in the response buffer. If a VSI is not connected to one of the MAC, VLAN to remove, it is silently ignored. EACCES - If one of the VSI is not owned by this PF.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Perfect Match MAC Used	16-17		Number of perfect match MAC used by this function.
Perfect Match MAC Un-allocated	18-19		Total number of perfect match MAC still un-allocated and not reserved by any function.
Unicast Hash MAC Un-allocated	20-21		Total number of unicast hash MAC still un-allocated.
Multicast Hash MAC Uun-allocated	22-23		Total number of multicast hash MAC still un-allocated.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

Table 38-243.MAC, VLAN Pairs to Remove Detail

Field	Offset	Description
MAC Address	0-5	MAC address to remove VSI from.
VLAN Tag		VLAN tag to remove VSI from.
Flags		0: perfect match: If set, an perfect match MAC address was used to create this MAC, VLAN pair. 1: Hash: If set, a hash MAC address was used to create this MAC, VLAN pair. 2: Reserved. 3: Ignore VLAN: If set, the forwarding is currently based only on MAC. 4: Remove from all VSIs. This will remove the filter from all the VSIs owned by this PF. 7:5: Reserved.
Reserved		Reserved.

The response buffer of this command contains the results of the MAC, VLAN pairs removal. It contains a set of *Num_addresses* 16-byte structures as listed in Table 38-244.

Table 38-244.Response Buffer Results (MAC, VLAN Pairs Removal)

Field	Offset	Description
Reserved	0-11	Reserved for command part.
Error Code	12	0x0 = Successful removal. 0x1:0xFE = Reserved. 0xFF = Request failed (The MAC, VLAN pair does not exist).
Reserved	13-15	Reserved.

Add VLAN (0x0252)

This command is used to add a set of VLANs to the VLAN table or to add a set of VLAN filters to up to three VSIs. All the VSIs must have the same SwitchID.

Note: In order to support egress (Tx) VLAN filtering, the *Enable VLAN anti spoof* flag should be set in the relevant Add VSI command. In order to support ingress (Rx) VLAN filtering, the *Promiscuous VLAN* flag in the Set VSI Promiscuous Modes command should be cleared.

For Private VLAN functionality both ingress and egress VLAN filtering are needed.

Note: By default VLAN filtering is disabled. The first time this command is applied, VLAN filtering is enabled.

Table 38-245.Add VLAN Command (Opcode: 0x0252) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0252	Command opcode.
Datalen	4-5		Length of buffer. Should be equal to Num_VLAN * 8 bytes.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Num_VLAN	16-17		Number of VLANs to add.

Table 38-245.Add VLAN Command (Opcode: 0x0252) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
SEID 0	18-19	0x0	Bit 15 = Valid SEID. Bit 14:10 = Reserved. Bit 9:0 = SEID number of the VSI.
SEID 1	20-21	0x0	Bit 15 = Valid SEID. Bit 14:10 = Reserved. Bit 9:0 = SEID number of the VSI.
SEID 2	22-23	0x0	Bit 15 = Valid SEID. Bit 14:8 = Reserved. Bit 9:0 = SEID number of the VSI.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

The command buffer of this command contains the details of the VLAN IDs to add. It contains a set of *Num_VLAN* 8-byte structures as listed in Table 38-246.

Table 38-246.Add VLAN Response (Opcode: 0x0252)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0252	Command opcode.
Datalen	4-5		Length of buffer - equals to Num_VLAN * 8 bytes.
Return Value/ VFID	6-7		Return value. The following error values can be returned: EINVAL - If all the VSIs do not point to the same switchID. ENOSPC - If there are not enough resources to assign the VLANs. The response buffer details which of the allocations failed. ENOENT - If one of the VSIs is not valid. EACCES - If one of the VSIs is not owned by this PF.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-19		Reserved.
VLAN Used	20-21		Total Number of VLAN used (VLAN is a shared resource).
VLAN Un-allocated	22-23		Total number of VLAN still un-allocated.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

Table 38-247.Command Buffer - VLAN IDs to Add Detail

Field	Offset	Description
VLAN Tag	0-1	VLAN tag to add.
Flags	2	0 = Local VLAN 2:1 = Private VLAN type: 00b = Regular VLAN (not private) 01b = Primary VLAN 10b = Secondary VLAN 11b = Reserved 4:3 = Private VLAN Port type (valid only if Private VLAN type is Primary or Secondary VLAN): 00b = Regular VLAN 01b = Promiscuous VSIs 10b = Community VSIs 11b = Isolated VSIs ***Note:*** If primary or secondary VLAN is set, the port type can not be zero. If both are not set, it must be zero. ***Note:*** The definition of a VLAN ID should be consistent within a VEB. Thus, subsequent commands should set the same private VLAN type an local VLAN flags when adding VSIs to the same VLAN ID. The private VLAN port type can be different for various VSIs within the same VLAN.
Reserved	3-7	Reserved for response part.

The response buffer of this command contains the results of the VLANs allocation. It contains a set of *Num_VLAN* 8-byte structures as listed in Table 38-248.

Table 38-248.Response Buffer - VLANs Allocation Detail

Field	Offset	Description
Reserved	0-3	Reserved for command part.
Result	4	0x0 = Allocation success. 0x1:0xFD = Reserved. 0xFE = Request failed due to inconsistent VLAN definition. 0xFF = Request failed due to lack of resources.
Reserved	5-7	Reserved.

Remove VLAN (0x0253)

This command is used to remove a set of VLANs from the VLAN table or to remove VLAN filters from up to three VSIs.

All the VSIs must have the same SwitchID.

Table 38-249.Remove VLAN Command (Opcode: 0x0253) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0253	Command opcode.
Datalen	4-5	0x0	Length of buffer. Should be equal to Num_VLAN * 8 bytes.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Num_VLAN	16-17		Number of VLANs to add.

Table 38-249.Remove VLAN Command (Opcode: 0x0253) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
SEID 0	18-19	0x0	Bit 15 = Valid SEID. Bit 14:10 = Reserved. Bit 9:0 = SEID number of the VSI.
SEID 1	20-21	0x0	Bit 15 = Valid SEID. Bit 14:10 = Reserved. Bit 9:0 = SEID number of the VSI.
SEID 2	22-23	0x0	Bit 15 = Valid SEID. Bit 14:8 = Reserved. Bit 9:0 = SEID number of the VSI.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

The command buffer of this command contains the details of the VLAN pairs to remove. It contains a set of *Num_VLAN* 8-byte structures as listed in Table 38-250.

Table 38-250.Command Buffer - VLAN Pairs to Remove Detail

Field	Offset	Description
VLAN tag	0-1	VLAN tag to remove VSI(s) from.
Flags	2	0 = Remove entire VLAN. 7:1 = Reserved.
Reserved	3	Reserved.
Reserved	4-7	Reserved for response part.

Table 38-251.Remove VLAN Response (Opcode: 0x0253)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0253	Command opcode.
Datalen	4-5		Length of buffer. Equal to Num_VLAN * 8 bytes.
Return Value/ VFID	6-7		Return value. The following error values can be returned: EINVAL - If all the VSIs do not point to the same switchID. ENOENT - If the VLAN does not exist or if one of the SEID does not point to a valid VSI. The response buffer details which of the removal failed. If a VSI is not connected to one of the MACs, VLAN to remove, it is silently ignored. EACCES - If one of the VSIs is not owned by this PF.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-19		Reserved.
VLAN Used	20-21		Total Number of VLAN used (VLAN is a shared resource).
VLAN Un-allocated	22-23		Total number of VLAN still un-allocated.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

The response buffer of this command contains the results of the MAC, VLAN pairs removal. It contains a set of *Num_VLAN* 8-byte structures as listed in Table 38-252.

Table 38-252.Response Buffer - MAC, VLAN Pairs Removal Detail

Field	Offset	Description
Reserved	0-3	Reserved for command part.
Error code	4	0x0 = Successful removal. 0x1:0xFE = Reserved. 0xFF = Request failed (the VLAN do not exist or if one of the VSIs is not valid).
Reserved	5-7	Reserved.

Set VSI Promiscuous Modes (0x0254)

This command is used to enable a VSI to set various promiscuous mode and other generic filtering options.

Note: If the *Default VSI* flag is set for a VSI within a VEB, this command changes the default VSI of the VEB to the VSI pointed by this command.

Table 38-253.Set VSI Promiscuous Modes Command (Opcode: 0x0254) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0254	Command opcode.
Datalen	4-5	0x0	Length of buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Flags	16-17		0 = Promiscuous unicast. 1 = Promiscuous multicast. 2 = Promiscuous broadcast. 3 = Default VSI. Accept packets within the switch ID not matching any specific address to this VSI. 4 = Promiscuous VLAN. 14:5 = Reserved. 15: Apply promiscuous mode to Rx traffic only. This bit is relevant only if one of options 0,1,2 is used. This bit is relevant only if switch is configured to work in limited promiscuous mode, and packet is not hitting VLAN mirroring rule. Multiple flags might be set. Default VSI is supposed to be used only for VMDq1 scenarios where there is no VEB. If this command is used in presence of a VEB, it changes the default VSI of the VEB. Default VSI should not be set if the VSI is connected directly to the port (not via a VEB or a PV).
Valid Flags	18-19		0 = Promiscuous unicast flag is valid. 1 = Promiscuous multicast is valid. 2 = Promiscuous broadcast is valid. 3 = Default VSI is valid. 4 = Promiscuous VLAN is valid. 15:6 = Reserved. Multiple valid bits might be set.
SEID Number	20-21		Bit 15 = Valid SEID. Ignored in this field. Bit 14:10 = Reserved. Bit 9:0 = SEID number of the VSI.

Table 38-253.Set VSI Promiscuous Modes Command (Opcode: 0x0254) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
VLAN ID	22-23		15 = VLAN is valid. 14:12 = Reserved. 11:0 = VLAN ID. If bit 15 is set, the *Promiscuous Unicast*, *Multicast*, and *Broadcast* flags applies only to this VLAN; otherwise, these modes apply to all VLANs. If VSI is in promiscuous VLAN mode, the VLAN ID should not be used.
Reserved	24-31	0x0	Reserved.

Table 38-254.Set VSI Promiscuous Modes Response (Opcode: 0x0254)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0254	Command opcode.
Datalen	4-5	0x0	Length of buffer.
Return Value/ VFID	6-7		Return value. The following error values can be returned: ENOENT - If the SEID does not point to a valid VSI. ENOSPC - If there are not enough resources to apply the promiscuous rules. EACCES - If the VSI is not owned by this PF. ENOMEM - No space for WoL filters.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-31		Reserved. Should contain the parameters from the command.

Add S-tag (0x0255)

This command is used to associate an S-tag with a VSI. This command can be given only by the port controlling the port virtualizer. This command should be used for cascaded port virtualizer VSIs where more than one S-tag points to the same VSI. For regular channels, the *Add VSI* command already adds the single S-tag of the VSI.

An S-tag can be directed to a specific queue in the VSI. This is done by setting the *ToQueue* flag and providing a valid queue in the *QueueNumber* field. In order to assign a queue to the default VSI, this command should be used with the default S-tag as tag value.

Note: The first S-tag added to the cascaded port virtualizer VSI via the *Add VSI* command is considered as the switch ID for this VSI and should not be manipulated by the Add S-tag and Remove S-tag commands.

Note: A VEB or L2 filters cannot be applied to a cascaded VSI on any S-tag. The VSI should be in default mode for its initial S-tag. The VSI should be defined with a *Connection Type* of *Default Port command*.

Table 38-255.Add S-tag Command (Opcode: 0x0255) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0255	Command opcode.
Datalen	4-5	0x0	Length of buffer.
Return Value/ VFID	6-7	0x0	Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.

Table 38-255.Add S-tag Command (Opcode: 0x0255) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Flags	16-17		0 = ToQueue. 15:1 = Reserved.
SEID	18-19	0x0	Bit 15 = Valid SEID. Ignored in this field. Bit 14:10 = Reserved. Bit 9:0 = SEID number of the VSI. Defines the VSI that uses this S-tag. This VSI should belong to the port virtualizer through which this command is received.
Tag	20-21	0x0	The value of the S-tag.
Queue Number	22-23	0x0	Bit 15:11 = Reserved. Bit 10:0 = Queue number. Valid only if the *Flags.ToQueue* bit is set. The queue number is relative to the VSI.
Reserved	24-31		Reserved.

Table 38-256.Add S-tag Response (Opcode: 0x0255)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0255	Command opcode.
Datalen	4-5	0x0	Length of buffer.
Return Value/ VFID	6-7		Return value. The following error values can be returned: ENOENT - If the SEID do not point to a valid element. ENOSPC - If there are not enough resources to assign a tag. EACCES - If the VSI is not owned by this PF. EEXIST - If this tag already points to another VSI. EPERM - Attempt to add tags to a VSI which is not a cascaded type.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-27	0x0	
S-tags Used	28-29		Number of tags used by this PF.
S-tags Un-allocated	30-31		Total number of tags still un-allocated.

Remove S-tag (0x0256)

This command is used to remove an S-tag from the forwarding table of a VSI.

Note: This command should be used only to remove tags in a cascaded port virtualizer VSIs.

Table 38-257.Remove S-tag Response (Opcode: 0x0256)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0256	Command opcode.
Datalen	4-5	0x0	Length of buffer.
Return Value/ VFID	6-7		Return value. The following error values can be returned: ENOENT - If the SEID do not point to a valid element. EACCES - If the VSI is not owned by this PF. ENXIO - This S-tag does not point to this VSI. EPERM - Attempt to remove tags from a VSI which is not a cascaded type.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-27		Reserved.
S-tag Used	28-29		Number of S-tags used by this function.
S-tag Un-allocated	30-31		Total number of S-tags still un-allocated.

Table 38-258.Remove S-tag Command (Opcode: 0x0256)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0256	Command opcode.
Datalen	4-5	0x0	Length of buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
VSI SEID	16-17	0x0	Bit 15 = Valid SEID. Ignored in this field. Bit 14:10 = Reserved. Bit 9:0 = SEID number of the VSI. Defines the VSI SEID that uses this tag. This VSI should belong to the port virtualizer through which this command is received.
S-tag	18-19	0x0	The value of the tag.
Reserved	20-31	0x0	Reserved.

Update S-tag (0x0259)

This command is used to update the S-tags associated with a VSI. This command can be given only by the port controlling the port virtualizer. This command can be used only for cascaded port virtualizer VSIs.

If a queue was associated with the previous tag, the association is kept for the new tag.

Table 38-259.Update S-tag Command (Opcode: 0x0259)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0259	Command opcode.
Datalen	4-5	0x0	Length of buffer.
Return Value/ VFID	6-7	0x0	Return value. Zeroed by driver. Written by Firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
SEID Number	16-17	0x0	Bit 15 = SEID Valid. Ignored in this field. Bit 14:10 = Reserved. Bit 9:0 = Defines the SEID of the VSI that uses this tag. This VSI should belong to the PF through which this command is received.
Old Tag	18-19	0x0	The original value of the S-tag. If the value is zero, it is assumed the VSI does not have a tag.
New Tag	20-21	0x0	The new value of the tag.
Reserved	22-31		Reserved.

Table 38-260.Update S-tag Response (Opcode: 0x0259)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0259	Command opcode.
Datalen	4-5	0x0	Length of buffer.
Return Value/ VFID	6-7		Return value. The following error values can be returned: ENOENT - If the SEID do not point to a valid element. ENOSPC - If there are not enough resources to assign a new tag. EACCES - If the VSI is not owned by this PF. EEXIST - If the new tag already points to another VSI. EPERM - An attempt to modify a tag in a VSI which is not a cascaded port extender type.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-27	0x0	
S-tags Used	28-29		Number of S-tags used by this PF.
S-tags Un-allocated	30-31		Total number of S-tags still un-allocated.

Add Control Packet Filter (0x025A)

This command is used to add a control filter to forward packets to a control VSI. This function should be used to forward packets to control VSIs; however, there is no enforcement of this rule and the software device driver might use it to forward control packets to other VSIs. Only packets reaching the switching element is forwarded by this rule. This means that:

- If the VSI is connected as a control port of the MAC, a port virtualizer or a VEB directly connected to the MAC, this filter applies to untagged packets (no S-tag).
- If the VSI is connected to an S-channel, either directly or as a VEB port, only packets tagged with the S-channel tag is forwarded.

Note: These filters are exclusive. If a request to set a filter on an existing flow type is received, it is rejected with an EEXIST reason code. If a filter is set to forward an EtherType ignoring the MAC address (Ignore MAC = 1), a filter with the same EtherType and a MAC address should not be applied within the same switch.

Note: The EtherType programmed by this command should not be one of the L2 tags EtherType (VLAN, S-tag, etc.) and should not be IP or IPv6.

Table 38-261.Add Control Packet Filter Command (Opcode: 0x025A)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x025A	Command opcode.
Datalen	4-5		Length of buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
MAC Address	16-21		The MAC address to use in the filter.
EtherType	22-23		The EtherType to use: The 16-bit value of the EtherType.
Command Flags	24-25		0 = Ignore MAC. If set, forwarding is based only on EtherType. 1 = Drop filter. If set, packets received or sent with this destination MAC address and EtherType are dropped. 2 = ToQueue. If set, the packets matching this filter is sent to a specific queue. If cleared, the regular queuing mechanism is used. A queue defined by a lower priority switch filter (such as MAC filter) is ignored. 3 = Direction: • 0 = Apply to Rx traffic. • 1 = Apply to Tx traffic. 15:4 = Reserved.
SEID Number	26-27		Bit 15 = SEID Valid Ignored in this field. Bit 14:10 = Reserved. Bit 9:0 = Defines the SEID of the VSI that should get the packet. ***Note:*** This field is ignored if the *Drop Filter* flag is set.
Queue Number	28-29		Queue to send the packet to if the *ToQueue* flag is set. The queue number is relative to the VSI.
Reserved	30-31		Reserved.

Table 38-262.Add Control Packet Filter Response (Opcode: 0x025A)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x025A	Command opcode.
Datalen	4-5		Length of buffer.
Return Value/ VFID	6-7		Return value. The following error values can be returned: ENOSPC - If there are not enough resources to assign the requested filter. ENOENT - If the VSI is not valid. EACCES - If the VSI is not owned by this PF. EEXIST - If such a filter already exists and is allocated to another VSI. ENOMEM - No space for WoL filters.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
MAC, EtherType Used	16-17		Number of perfect match MAC, EtherType used by the device.
EtherType Used	18-19		Number of perfect EtherType used by the device.
MAC, EtherType Not Allocated	20-21		Number of perfect match MAC, EtherType still un-allocated.
EtherType Not Allocated	22-23		Number of perfect EtherType still un-allocated.
Reserved	24-31		Reserved.

Remove Control Packet Filter (0x025B)

This command is used to remove a control filter. The filter to remove is defined by the switching element to which the VSI is connected. This means that:

- If the VSI is connected as a control port of the MAC, a port virtualizer or a VEB directly connected to the MAC, the removed filter relates to untagged packets (no S-tag).
- If the VSI is connected to an S-channel, either directly or as a VEB port, the removed filter relates to packet tagged with the S-channel tag.

Table 38-263.Remove Control Packet Filter Command (Opcode: 0x025B) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x025B	Command opcode.
Datalen	4-5		Length of buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
MAC Address	16-21		The MAC address in the filter.
EtherType	22-23		The EtherType used: The 16-bit value of the EtherType.
Command Flags	24-25		0 = Ignore MAC. If set, forwarding is based only on EtherType. 2:1 = Reserved 3 =: Direction: • 0 = Apply to Rx traffic. • 1 = Apply to Tx traffic. 15:4 = Reserved.

Table 38-263.Remove Control Packet Filter Command (Opcode: 0x025B) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
SEID Number	26-27		Bit 15 = SEID Valid. I - ignored in this field. Bit 14:10 = Reserved. Bit 9:0 = Defines the SEID of the VSI to which the filter is associated.
Reserved	28-31		Reserved.

Table 38-264.Remove Control Packet Filter Response (Opcode: 0x025B)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x025B	Command opcode.
Datalen	4-5		Length of buffer.
Return Value/ VFID	6-7		Return value. The following error values can be returned: ENOENT- If the VSI is not connected to the MAC, EtherType or EtherType filter to remove. EACCES - If the VSI is not owned by this PF.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
MAC, EtherType Used	16-17		Number of perfect match MAC, EtherType used by the device.
EtherType Used	18-19		Number of perfect EtherType used by the device.
MAC, EtherType Not Allocated	20-21		Number of perfect match MAC, EtherType still un-allocated.
EtherType Not Allocated	22-23		Number of perfect EtherType still un-allocated.
Reserved	24-31		Reserved.

Add Cloud Filters (0x025C)

This command is used to add a set of cloud filters to a VSI. The filters set by this commands are the filters specific to cloud formats. To add MAC or MAC, VLAN filters, the *Add MAC, VLAN pairs* (0x0250) command should be used.

Note: The response cannot contain all the cloud resources available, the software device driver needs to use the *Get Switch Resources Allocation* command (0x0204) to get the resources left after allocation of a filter. See Get Switch Resources Allocation (0x0204) for more detail.

Table 38-265.Add Cloud Filters Command (Opcode: 0x025C) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x025C	Command opcode.
Datalen	4-5		Length of buffer. Should be equal to 64 * Num_filters bytes.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Num_filters	16		The number of filters to add.
Reserved	17	0x0	Reserved.

Table 38-265.Add Cloud Filters Command (Opcode: 0x025C) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
SEID	18-19	0x0	Bit 15 = SEID Valid. Ignored in this field (SEID is always valid). Bit 14:10 = Reserved. Bit 9:0 = Defines the SEID of the VSI.
Reserved	20-23	0x0	Reserved.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

The command buffer of this command contains the details of the filters to add. It contains a set of *Num_filters* 64-byte structures as listed in Table 38-266.

Table 38-266.Add Cloud Filters Response (Opcode: 0x025C)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x025C	Command opcode.
Datalen	4-5		Length of buffer Equals to 64 * Num_filters bytes.
Return Value/ VFID	6-7		Return value. The following error values can be returned: ENOSPC - If there are not enough resources to assign all the filters. The return buffer details which of the allocations failed. ENOENT - If the VSI is not valid. EACCES - If the VSI is not owned by the offload PF. EEXIST - This filter is already allocated to another VSI. ENOMEM - No space for WoL filters.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Num_filters	16		The number of filters in the response buffer.
Reserved	17-23		
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

Table 38-267.Add Cloud Filters - Command Buffer (Sheet 1 of 2)

Field	Offset	Description
Outer MAC Address	0-5	Outer MAC Address to add.
Inner MAC Address	6-11	Inner MAC Address add.
Inner VLAN Tag	12-13	Inner VLAN tag to add (only 12 LSBits are relevant to filter).
IP	14-29	IP address to add. For IPv4 addresses, bytes 14-25 are reserved.

Table 38-267.Add Cloud Filters - Command Buffer (Sheet 2 of 2)

Field	Offset	Description
Command Flags	30-31	6:0 = Filter type: • 0x0 = Reserved. • 0x1 = Reserved. • 0x2 = Reserved. • 0x3 = Inner MAC, inner VLAN (for NVGRE, VXLAN or Geneve packets). • 0x4 = Inner MAC, inner VLAN, Tenant ID (for NVGRE, VXLAN or Geneve packets). • 0x5 = Reserved. • 0x6 = (Inner MAC, tenant ID) (NVGRE packet or VXLAN/Geneve packets). • 0x7 = Reserved. • 0x8 = Reserved. • 0x9 = Outer MAC L2 filter. • 0xA = Inner MAC filter. • 0xB = Outer MAC, tenant ID, inner MAC. • 0xC = Application destination IP. • 0xD:0x3F = Reserved. ***Note:*** Filter 0xB should not be used in modes with port extenders. An outer MAC L2 filter (0x9) should be added only once per relevant MAC address. This filter applies to all the VSI of type cloud. 7 = ToQueue: If set, a packet matching this filter is sent to a specific queue. Not relevant if filter type = 0x9. 8 = IP address type: • 0 = IPv4. • 1 = IPv6. Relevant only for filter type 0xC. 12:9 = Tunnel Type: • 0x0 = VXLAN. • 0x1 = NVGRE or other MAC in GRE. • 0x2 = Geneve. • 0x3 = IP in GRE. • 0x4 = Reserved. • 0x5:0xF = Reserved. Relevant only for filter types 0x4, 0x6, and 0xB. Relevant only for filter types 0x4, 0x6, and 0xB. 13: Use shared outer MAC: If set, the outer MAC is taken from the shared pool. Relevant only for filters involving outer MAC. 14: Use shared inner MAC: If set, the inner MAC address is taken from the shared pool. Relevant only for filters involving inner MAC. 15: Use shared Outer IP: If set, the outer IP address is taken from the shared pool. Relevant only for filters involving outer IP.
Tenant ID	32-35	Key to add: • NVGRE tunnel type: TNI. • VXLAN/Geneve tunnel type: VNI. • Other GRE tunnel type: GRE Key. ***Note:*** The tenant ID is 3 bytes (32..34) and byte 35 is reserved.
Reserved	36-39	Reserved.
Queue Number	40-41	Bit 15:11: Reserved. Bit 10:0: Queue number. Valid only if the *Command Flags.ToQueue* bit is set. The queue number is relative to the VSI.
Reserved	42-55	Reserved.
Reserved	56-63	Reserved for response part.

The response buffer of this command contains the results of the filters allocation. It contains a set of *Num_filters* 64-byte structures as listed in Table 38-268.

Table 38-268.Add Cloud Filters - Response Buffer

Field	Offset	Description
Reserved	0-55	Reserved for command part.
Allocation result	56	0x0 = Filter assigned. 0x1:0xFE = Reserved. 0xFF = Request failed due to lack of resources.
Reserved	57-63	Reserved.

Remove Cloud Filters (0x025D)

This command is used to remove a set of cloud filters from a VSI. The filters removed by this command is the filters specific to cloud formats. To remove MAC or MAC, VLAN filters, the *Remove MAC, VLAN pairs* (0x0251) command should be used.

Table 38-269.Remove Cloud Filters Command (Opcode: 0x025D)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x025D	Command opcode.
Datalen	4-5		Length of buffer. Should be equal to 64 * Num_filters bytes.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Num_filters	16		The number of filters to remove.
Reserved	17	0x0	Reserved.
SEID	18-19	0x0	Bit 15 = SEID Valid. Ignored in this field. Bit 14:10 = Reserved. Bit 9:0 = Defines the SEID of the VSI.
Reserved	20-23	0x0	Reserved.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

The command buffer of this command contains the details of the filters to remove. It contains a set of *Num_filters* 16-byte structures as defined in the *Add Cloud filters* command buffer (see Table 38-267).

Note: The *ToQueue* flag and *Queue Number* field are not used by the *Remove Cloud Filter* command.

Table 38-270.Remove Cloud Filters Response (Opcode: 0x025D) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x025D	Command opcode.
Datalen	4-5		Length of buffer. Equals to 64 * Num_filters bytes.

Table 38-270.Remove Cloud Filters Response (Opcode: 0x025D) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Return Value/ VFID	6-7		Return value. The following error values can be returned: ENOENT - If the VSI is not connected to one of the cloud filters or the cloud filter entry does not exist or if one of the VSIs is not valid. The details of which remove request failed is found in the response buffer. EACCES - If the VSI is not owned by the offload PF (VSI is on another port).
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Num_filters	16		The number of filters in the response buffer.
Reserved	17-23	0x0	Reserved.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

The response buffer of this command contains the results of the filters allocation. It contains a set of *Num_filters* 16 bytes structures as defined below.

Table 38-271.Remove Cloud Filters - Response Buffer

Field	Offset	Description
Reserved	0-13	Reserved for command part.
Allocation Result	14	0x0 = Successful removal. 0x1:0xFE = Reserved. 0xFF = Request failed (one of the VSIs is not connected to this filter or the filter entry does not exist).
Reserved	15	Reserved.

Clear All WoL Switch Filters (0x0025E)

This command clears all the switch filters directed to the WoL VSI of the PF that sent the command.

Table 38-272.Clear All WoL Switch Filters Command (Opcode: 0x0025E)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x025E	Command opcode.
Datalen	4-5	0	
Return value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-31		Reserved.

Table 38-273.Clear All WoL Switch Filters Response (Opcode: 0x025E) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x025E	Command opcode.
Datalen	4-5	0	

Table 38-273.Clear All WoL Switch Filters Response (Opcode: 0x025E) (Sheet 2 of 2)

Return value/VFID	6-7		Return value. No specific errors.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-31		

38.21.7.18.9 Mirroring Commands (Opcode 0x026x)

Add Mirror Rule (0x0260)

This command is used to add a mirror rule to a specific switch. Mirror rules are supported for VEBs or VEPA elements only.

Table 38-274.Add Mirror Rule Command (Opcode: 0x0260)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0260	Command opcode.
Datalen	4-5		Length of buffer. Should be equal to 2 * Number of mirrored entries.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
SEID	16-17		Defines the SEID of the switch to which the rule refers.
Rule Type	18-19		2:0 = Rule Type: • 000b = Reserved. • 001b = Virtual port ingress mirroring. • 010b = Virtual port egress mirroring. • 011b = VLAN mirroring. • 100b = All ingress traffic (to this switch). Includes traffic sent from all VSIs connected to this switch. Does not include traffic received from LAN. • 101b = All egress traffic (from this switch). Includes traffic sent to all VSI in this switch. Does not include traffic sent to the LAN. • 11xb = Reserved. 15:3 = Reserved. If the rule type is 100b (all ingress) or 101b (all egress), then there is no associated buffer. The flags in bytes 0-1 should be set accordingly. If all egress or all ingress is needed, then no other egress/ingress rules respectively within the VEB/VEPA should be set, as a packet cannot be replicated by two rules. VSIs added to a switch after the all ingress/all egress rule was added are not included in the rule and the rule should be removed and re-applied to include them.
Number Of Mirrored Entries	20-21		Defines the number of VSI/VLANs that should be mirrored. The values are in the command buffer.
Destination VSI	22-23		Defines the VSI SEID to which the packets matching the mirror rule are mirrored.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

The command buffer is built as a number of mirrored entries of two bytes each containing a single VSI/VLAN, depending on the rule requested as listed in Table 38-275.

Table 38-275.Add Mirror Rule Command Buffer

Byte	Description
0-1	Mirrored VSI (SEID) or VLAN ID. For all rule types but ingress VLAN mirroring, the values are SEIDs of VSI. For ingress VLAN mirroring rule type, the values are VLAN IDs. The VSIs in the list should be part of the switch defined by the SEID.

Table 38-276 lists the add mirror rule response (with no buffer)

Table 38-276.Add Mirror Rule Response (Opcode: 0x0260)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0260	Command opcode.
Datalen	4-5		Length of buffer - equals to 2 * Number of mirrored entries.
Return Value/ VFID	6-7		Return value. The following error values can be returned: ENOENT - If the mirror SEID do not point to a valid switch element or the SEID of the mirrored VSI does not point to a valid VSI. ENOSPC - If there are not enough resources to assign an mirror rule. EACCES - If the VEB is not owned by this PF. EINVAL - If the same VLAN mirroring rule is added twice. EEXIST - rule already assigned. The rule should be removed before being assigned again.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-17		Reserved.
Rule ID	18-19		Defines the rule ID that is returned in the receive descriptor. This number is assigned by firmware and should be used as a handle when requesting deletion of an existing rule. The rule ID is not relevant for the ingress VLAN mirroring rule type (011b).
Mirror Rules Used	20-21		Number of mirror dedicated rules used by this function.
Mirror rules Un-allocated	22-23		Total number of mirror rules still un-allocated.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

Delete Mirror Rule (0x0261)

This command is used to delete an existing mirror rule. If the rule to remove is of ingress VLAN mirroring type, a buffer should be provided containing the currently mirrored VLAN to remove.

Table 38-277.Delete Mirror Rule Command (Opcode: 0x0261) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0261	Command opcode.
Datalen	4-5	0x0	Length of buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.

Table 38-277.Delete Mirror Rule Command (Opcode: 0x0261) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command
SEID	16-17		Defines the SEID of the switch to which the rule refers.
Rule Type	18-19		2:0 = Rule Type: • 000b = Reserved. • 001b = Virtual port ingress mirroring. • 010b = Virtual port egress mirroring. • 011b = Ingress VLAN mirroring. • 100b = All ingress traffic (to this switch). • 101b = All egress traffic (from this switch). • 11xb = Reserved. 15:3 = Reserved.
Number Of Mirrored Entries	20-21		Defines the number of VSI/VLANs that should be mirrored. The values are in the command buffer.
Rule ID	22-23		Defines the rule ID that is returned in the receive descriptor. This ID identifies the rule to delete. This ID is the number returned from the Add Mirror Rule response. Relevant only if rule type is not *Ingress VLAN mirroring* (011b).
Data Address High	24-27		Address of buffer. Relevant only if rule type is *Ingress VLAN mirroring* (011b).
Data Address Low	28-31		

The structure of the buffer used to indicate the VLAN to remove is listed in Table 38-278.

Table 38-278.Delete Mirror Rule - Command Buffer

Byte	Description
0-1	Mirrored VLAN ID to remove. 15:12 = Reserved. 11:0 = VLAN ID.

Table 38-279 lists the delete mirror rule response (with no buffer).

Table 38-279.Delete Mirror Rule Response (Opcode: 0x0261) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0261	Command opcode.
Datalen	4-5		Length of buffer.
Return Value/ VFID	6-7		Return value. The following error values can be returned: ENOENT - If the SEID do not point to a valid switch element. EINVAL - If the rule ID does not exist. EACCES - If the VEB is not owned by this PF.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-19		
Mirror Rules Used	20-21		Number of mirror rules used by this function.
Mirror rules Un-Allocated	22-23		Total number of mirror rules still un-allocated.

Table 38-279.Delete Mirror Rule Response (Opcode: 0x0261) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

38.22 Interrupts

38.22.1 Interrupt Signaling

The 10 GbE controller supports the following interrupt signaling according to per PF setting options:

- Legacy INTA/INTB/INTC/INTD interrupt message on the PCIe is supported for the PFs. The 10 GbE controller exposes legacy interrupt support in the *Interrupt Pin* field in the PCI configuration space of the PF. The interrupt pin parameter is loaded from the *INTPIN* field in the NVM (to the PFPCI_CNF register) defining the interrupt pin (A,B,C,D or none) per PF. It is the NVM programmer's responsibility to follow the PCI rules for allocating orderly interrupt pins for the PCI functions. The legacy interrupt is triggered by the interrupt vector zero of the PF.
- MSI is exposed in the MSI capability structure in the PCI configuration space of each PF. The MSI interrupt is triggered by the interrupt vector zero of the PF. The MSI capability is enabled per PF by the MSI_En bit in the PFPCI_CNF (loaded from the NVM).
- MSI-X enables multiple interrupts for the PFs as well as the VFs. MSI-X is exposed in the MSI-X capability structure in the PCI configuration space of all PFs and VFs. The number of supported MSI-X vectors is defined by the table size parameters in the MSI-X capability structure in the PCI configuration space of the PFs and the VFs. The table size parameters is loaded from a global MSI_X_PF_N and MSI_X_VF_N parameters in the GLPCI_CNF2 register (loaded from the NVM), shared for all PFs and all VFs respectively. The MSI_X_PF_N and MSI_X_PF_N parameters must be lower or equal than the maximum number of MSI-X vectors per functions as described in the text that follows. The 10 GbE controller supports up to 1168 MSI-X vectors that are allocated to PFs and VFs uniformly as a function of the number of enabled PFs and the number of enabled VFs. The maximum number of MSI-X vectors per function is listed in the Table 38-280.

Table 38-280.MSI-X Vector Allocation per Function

Index of Max Enabled PF	Max Number Of MSI-X Vectors Per PF	Number of Registers Per PF Indicated As INTPF
1 ... 4	1 + 128	128
5 ... 8	1 + 64	64
9 ... 16	1 + 32	32

Index Of Max Enabled VF	Max Number Of MSI-X Vectors Per VF	Number of Registers Per VF Indicated As INTVF Or INTVP
1 ... 32	1 + 16	16
33 ... 64	1 + 8	8
65 ... 128	1 + 4	4

Note: The Index of max enabled PF and the index of max enabled VF are defined by the values of the *PCIPFCNT* and *PCIVFCNT* fields, respectively, in the GLGEN_PCIFCNCNT register (loaded from the NVM)

Note: All registers with attribute INTPF, INTVF and INTVP are reflected in the function's space as 512 registers, which is the total number of these registers in the device. Still each function might only access its own registers (starting at register index zero) according to the number of registers listed in Table 38-280.

The INTVP registers are mapped in the PF space in the following manner:

The register index in the PF space for register ‘n’ of VF ‘m’ = Number of Registers per VF * ‘m’ + ‘n’, while ‘m’ is the relative VF index within the PF and the Number of Registers per VF can be 4, 8 or 16 as listed in Table 38-280.

The VP registers are mapped in the PF space in the following manner:

The register index in the PF space for VF 'm' = 'm', while 'm' is defined the same as previously described.

A register address in the PF space = base address of the specific registers + register index * 4.

38.22.1.1 Interrupt Enable Procedure

Interrupts are enabled at three levels:

- Enablement on the PCIe interface by PCI configuration registers programmed by the operating system.
  - Legacy INTA/INTB/INTC/INTD interrupt message is controlled by the *Interrupt Disable* flag in the Command register in the PCI config space per PF.
  - MSI is enabled by the *MSI Enable* flag in the MSI capability structure in the PCI config space per PF.
  - MSI-X is enabled by the *MSI-X Enable* flag in the MSI-X capability structure in the PCI config space per PF and per VF and further enablement by the *Mask* bit per MSI-X vector in the MSI-X table structure.
- Enablement of the interrupts by the software device driver by the *INTENA* flag in the xxINT_DYN_CTL0 and xxINT_DYN_CTLN registers (where xx=PF or VF).
  - The software device driver sets the *INTENA* flag to enable the relevant interrupt signal. Upon interrupt assertion, the *INTENA* flag and the interrupt level are auto-cleared. Auto-clearing the INTENA can be disabled globally by the GLINT_CTL.
- Enablement of the interrupts per cause by the *CAUSE_ENA* flag in the cause control registers (see Section 38.22.2).
- Interrupt moderation:
  - Interrupt Throttling (ITR) is described in Section 38.22.4.1.
  - Interrupt rate limiting (INTRL) is described in Section 38.22.4.2.

38.22.1.2 Pending Interrupt Array - PBA

Including the interrupt signaling on the PCIe bus, the 10 GbE controller also supports the standard PBA structure in the MSI-X BAR. The PBA is relevant only when MSI-X is enabled. It is described as part of the MSI-X capability structure for the PF for the VFs. A bit in the PBA is set to 1b when an interrupt is triggered internally and cleared when the MSI-X vector is sent on the PCIe bus.

38.22.1.3 Interrupt Sequence

This section describes the interrupt sequence of events starting by an internal events that triggers an interrupt until it is sent to the PCIe bus and the expected software response. Note that the description of the interrupt sequence refers to flags that are explained in the following sections.

MSI and MSI-X interrupts while interrupts are enabled

1. Any of the interrupt causes has an event that sets an internal *INTEVENT* flag for the matched interrupt signal.
2. If the interrupt is enabled by *INTENA and also enabled by the interrupt moderation policy (ITR and rate limiting)*, hardware executes the following steps in the following order:
   a. Scan all queues associated with this interrupt by a linked list. Write back the status of all completed descriptors that were not reported so far and clear the internal EVENT flags of these queues.
   b. Set the matched bit in the pending interrupt block array (PBA) (represented by the ARB block). Note that the PBA is reflected externally only with MSI-X (and not with legacy interrupts nor MSI).
   c. The *INTEVENT* and *INTENA* are auto-cleared.
3. If the interrupt is enabled by the operating system (by PCIe setting), the interrupt message is sent to the PCIe.
   a. The interrupt indication in the PBA is auto-cleared.
4. During the interrupt handler, software processes each individual interrupt causes. Specific to interrupt zero of the function (the other causes interrupt), software can optionally read the ICR0 register identifying the interrupt causes to be processed. Reading the ICR0 register clears it making it ready to reflect the events of the next interrupt.
5. At the end of the interrupt handler, software re-enables the interrupts by setting the *INTENA*.
   a. On the same register, software also sets the *CLEARPBA* flag that clears the matched bit in the PBA.
   b. On the same register, software can update one of three ITRs of this interrupt by setting the *ITR_INDX* and *INTERVAL* fields. Setting the *ITR_INDX* to 11b does not impact the ITRs. See ITR explanation in Section 38.22.4.1.

MSI-X interrupts while interrupts are disabled by the operating system

Steps 1 and 2 are the same as previously described.

6. The operating system polls the PBA and schedules the interrupt handler.

Steps 4 and 5 are the same as previously described.

Legacy interrupts while interrupts are enabled by the operating system (mapped to interrupt zero of the PF)

Steps 1 and 2 are the same as previously described.

1. If the interrupt is enabled by the operating system (by PCIe setting), the interrupt message is sent to the PCIe.

2. At the very beginning of the interrupt service routing, the software device driver clears the internal PBA by setting the *CLEARPBA* flag in the PFINT_DYN_CTL0 register.
   a. As a result, an interrupt de-assertion message is sent on the PCIe bus.

Note: When several PFs are assigned to the same legacy interrupt, the interrupt de-assertion message must be only sent when all PFs associated with that interrupt have cleared the internal PBA previously mentioned.

   b. The software device driver also reads PFINT_ICR0 the schedules the matched processes for the active flags in the register.
3. During the interrupt handler, software processes each individual interrupt causes. Software can optionally read the ICR0 register identifying the interrupt causes to be processed. Reading the ICR0 register clears it making it ready to reflect the events of the next interrupt.
4. At the end of the interrupt handler, software re-enables the interrupts by setting *INTENA*.
   a. On the same register, software can update one of three ITRs as previously explained.

Figure 38-57.Conceptual Interrupt Logic

38.22.2 Interrupt Causes

This section lists all interrupts causes (sources) while mapping these causes to the interrupt vectors as described in Section 38.22.3. Note that only the PF has access to any of the cause registers listed in this sub-section. Therefore, VF is required to request its registers programming from the PF by admin command or any other sideband channel.

38.22.2.1 LAN Transmit Queues

The 10 GbE controller supports 1536 LAN transmit queues for the entire device while each queue is a potential interrupt cause. A status reporting of a completed transmit descriptor with the *EOP* bit set is considered as a transmit event that can trigger an interrupt (if the interrupt is enabled).

LAN transmit queue ‘n’ is enabled for interrupts by the CAUSE_ENA flag in its QINT_TQCTL[n] register. The queues are mapped to any interrupt vector within the function space by the MSIX_INDX field in the QINT_TQCTL[n] register and mapped to any of it’s ITRs (or immediate interrupt) by the ITR_INDX in this register. See ITR description in Section 38.22.4.1. Using interrupt vector zero, the transmit queues are mapped by the MSIX0_INDX field in the QINT_TQCTL[n] register to one of the QUEUE_x flags in the xxINT_ICR0 registers. The MSIX0_INDX of PF queues can be set to 0...7 while VF queues can be set only to 0...3.

38.22.2.2 LAN Receive Queues

The 10 GbE controller supports 1536 LAN receive queues for the entire device while each queue is a potential interrupt cause. A DMA completion of a descriptor with an *EOP* flag and its buffer is considered as a receive event that triggers an interrupt (if the interrupt is enabled). Furthermore, if the number of free descriptors on the receive queue drops below the threshold, it is considered as an immediate event (described in the following sub-sections). The low threshold is defined by the LRXQTRESH parameter in the receive queue context.

Similar to the LAN transmit queues, the LAN receive queues are mapped to any ITR of any interrupt vector by the matched QINT_RQCTL registers.

38.22.2.3 PE Queues

A completion of an AEQ entry or a CEQ entry is considered as an event that triggers an interrupt (if the interrupt is enabled). PE CEQs and AEQs can be mapped to any interrupt with optional interrupt moderation described in the following sub-sections.

The 10 GbE controller supports a single AEQ interrupt control register (per function). It is enabled for interrupts by the *CAUSE_ENA* flag and mapped to any ITR of any interrupt vector by the xxINT_AEQCTL register ('xx' equals to 'PF' or 'VP').

CEQs (per function) are enabled for interrupts by the *CAUSE_ENA* flag and mapped to any ITR of any interrupt vector by the xxINT_CEQCTL registers ('xx' equals to 'PF' or 'VP'). These registers are allocated evenly to PFs and VFs depending on the number of the functions (as listed in Table 38-280).

Using interrupt vector zero, the queues are mapped to one of the *QUEUE_x* flags in the xxINT_ICR0 registers by the *MSIX0_INDX* field in the previously identified registers. The *MSIX0_INDX* of PF queues can be set to 0...7 while VF queues can be set only to 0...3.

Note that all PFs and up to 32 VFs can be enabled for the PE. Each enabled function has a single AEQ and a total of 256 global CEQs. Yet, all functions have the xxINT_AEQCTL and VPINT_CEQCTL registers (total of 144 AEQ interrupt control registers and 1024 CEQ interrupt control registers). Mapping the physical 256 CEQs to the xxINT_CEQCTL registers is done by the PE firmware. It is expected that software uses only those registers that represent real AEQs and CEQs behind them.

38.22.2.4 Other Interrupt Causes

The 10 GbE controller supports asynchronous events that can generate interrupts for the PFs and the VFs. The other interrupt events supported by the PFs are indicated in the PFINT_ICR0 register and enabled by the PFINT_ICR0_ENA register (per PF) as listed in Table 38-281. The other interrupt events supported by the VFs are indicated in the *VFINT_ICR0* register and enabled by the *VFINT_ICR0_ENA* register (per VF) as listed in the Table 38-282.

The other interrupt cause is mapped to MSI-X vector zero of the functions. It is mapped to any of its ITRs (or immediate interrupt) as programmed by the *OTHER_ITR_INDX* field in the *PFINT_STAT_CTL0* register for the PF and the VFINT_STAT_CTL0 register for the VF (see Section 38.22.4.1).

During normal operation it is expected that *xxINT_ICR0* is used only as a read/clear register by software. Setting the flags in the *xxINT_ICR0* register (other than the queue flags and the *SWINT* flag), emulates an interrupt event of the specific cause (if enabled by the *xxINT_ICR0_ENA* register).

Table 38-281.Other Interrupt Causes of the PFs

PF Other Cause	Description
ECC_ERR	Unrecoverable ECC Error. This bit is set when an unrecoverable error is detected in one of the device memories.
MAL_DETECT	Malicious programming detected.
GRST	Global Resets Requested (CORER, GLOBR or EMPR).
GPIO	GPIO event indicates an event on any of the GPIO pins enabled for interrupt by the PFINT_GPIO_ENA register. The GPIO state can be fetched on the GLGEN_GPIO_STAT register. The level transition that generates an interrupt is set for GPIO 'n' by the *INT_MODE* field in the matched GLGEN_GPIO_CTL[n] register.
TIMESYNC	Any of the TimeSync interrupt causes.
HMC_ERR	HMC error as indicated in the PFHMC_ERRORINFO and PFHMC_ERRORDATA registers.
PE_CRITERR	Protocol Engine Critical Error. Indicates that the PE has encountered a critical error. See the GLPE_CRITERR register to determine the specific error that caused the interrupt.
VFLR	VFLR was initiated by one of the VFs of the PF. The PF should read the GLGEN_VFLRSTAT getting an indication for the VF that generated the VFLR.
ADMINQ	Send / receive admin queues interrupt.
SWINT	Software interrupt (see Section 38.22.2.5).

Table 38-282.Other interrupt causes of the VFs

PF Other Cause	Description
ADMINQ	Send / receive admin queues interrupt.
SWINT	Software interrupt (see Section 38.22.2.5).

38.22.2.5 Software Initiated Interrupt

In some cases, software might not be able to process all events in a single interrupt handler. In such cases, software might schedule another interrupt to complete processing all interrupt events. Software can trigger an interrupt on any vector and any of its ITRs or NoITR. The software interrupt is mapped to one of three ITRs or immediate interrupt by the *SW_ITR_INDX* field in the *xxINT_DYN_CTLx* registers ('xx' stands for PF or VF and 'x' stands for '0' or 'N'). When programming the *SW_ITR_INDX* parameter, the *SW_ITR_INDX_ENA* flag in this register should be set as well.

Software initiates the interrupt by setting the *SWINT_TRIG* flag in the *xxINT_DYN_CTLx* registers. When setting the *SWINT_TRIG* flag, software should set also the *INTENA* flag in the same register.

Setting the software interrupt in vector 0 of the PFs, its status indication is reflected in the *PFINT_ICR0* register (including interrupt triggering).

38.22.2.6 Interrupt Status Registers

During normal operation, software avoids any possible read accesses. Interrupt zero is used for all the other causes, which enforces read cycles. The 10 GbE controller provides a status indication of all causes of interrupt zero of the PFs and the VFs. The xxINT_ICR0 registers ('xx' stands for PF or VF) provide indication for the following events:

- Any of the other interrupt causes.
- Up to eight status flags (QUEUE_0 ... QUEUE_7) in the PFs and up to four status flags (QUEUE_0 ... QUEUE_3) in the VFs for any LAN transmit or receive queues or PE_CEQ associated with interrupt zero.

- *SWINT* indication, which is a result of software initiated interrupt.

The status indication in the xxINT_ICR0 registers is independent on the xxINT_ICR0_ENA registers setting. The xxINT_ICR0_ENA registers enable interrupt assertion by each of the ICR0 causes. Interrupt on the PCIe is further enabled by the interrupt vector 0 logic (legacy interrupt or MSI or MSI-X 0).

38.22.3 Interrupt Linked List

The queues are linked to a specific interrupt vector by the MSIX_INDX field in the xxxQCTL registers. Specifically, the LAN queues and the PE completion event queues are associated to an interrupt vector by the following registers: QINT_RQCTL, QINT_TQCTL, PFINT_CEQCTL, VPINT_CEQCTL. These queues are chained together in a linked list that is configured by these xxxQCTL registers. Interrupt causes mapping to the interrupt vectors and the interrupt linked list are illustrated in Figure 38-58.

When an interrupt sequence is triggered, the 10 GbE controller processes all queues in the linked list of this interrupt. While processing the queues in the linked list, hardware writes back the status for those completed descriptors that were not reported already as listed as follows:

- LAN receive queues — Hardware triggers a status write back of all completed descriptors.
- LAN transmit queues — Hardware write back the completion indication of the last completed transmit descriptor that has an EOP flag (regardless of the RS bit) and was not reported already.
- PE CEQ entries — Hardware write back the completion indication of all completed CEQ entries.

Note: The xxxQCTL registers are accessible only to the PF. A VF can assign causes to interrupts with the help of the PF using an admin command or any other sideband message (out of scope of this document).

Note: The linked list could contain LAN transmit and receive queues as well as PE completion event queues. If multiple types of interrupt causes are mapped to the same interrupt, Intel recommends (for optimized performance) to interleave them in the linked list as possible.

The beginning of the linked list is indicated per interrupt signal by the following parameters in the xxINT_LNKLSTx registers (while ‘xx’ stands for PF or VP and ‘x’ stands for ‘0’ or ‘N’).

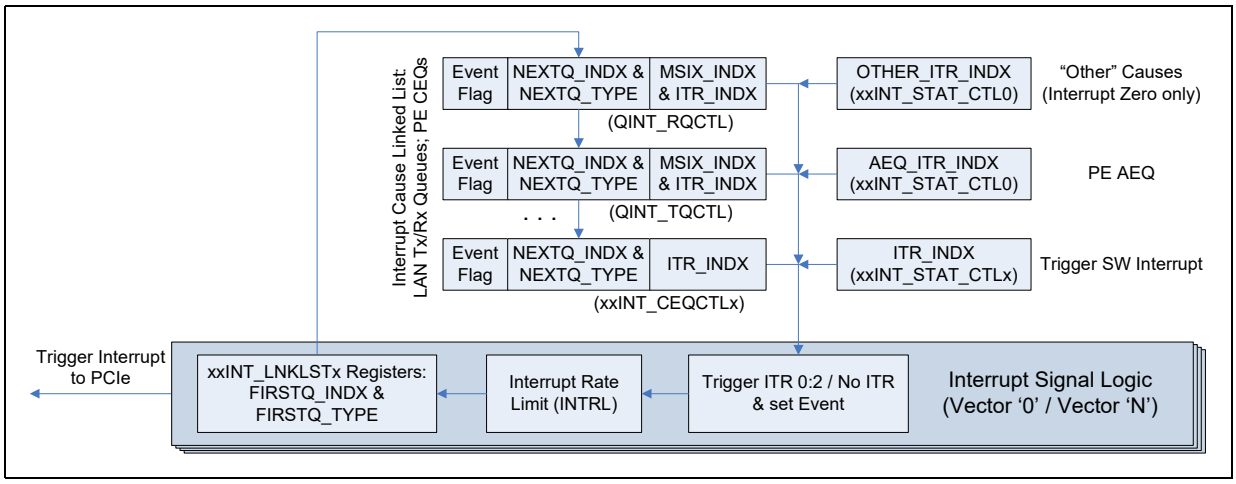
- FIRSTQ_TYPE — The type of the first cause in the list assigned to the interrupt signal. The type can be one of the following: receive queues, transmit queues and PE CEQ.
- FIRSTQ_INDX — The queue index of the first cause in the linked list assigned to this interrupt signal. For receive and transmit queues it is the index within the PF space (both PF and its VF queues are defined relative to the PF queue space). For CEQ, it is an index within the function space (VF queues are relative to the VF space and the PF queues are relative to the PF space). The FIRSTQ_INDX could be NULL pointer for no linked list.

An interrupt vector assigned only to the other causes and possibly to AEQ as well, does not have a linked list. In this case, software should set the FIRSTQ_INDX to a NULL value (0x7FF).

The interrupt causes are linked to each other by similar parameters as previously described in the following cause control registers: QINT_RQCTL; QINT_TQCTL and xxINT_CEQCTLx registers ('xx' stands for PF or VP and 'x' stands for '0' or 'N').

- NEXTQ_TYPE — The type of the next cause in the list assigned to the same interrupt signal. It is the same definition as the FIRSTQ_INDX previously described.
- NEXTQ_INDX — The queue index of the next cause in the linked list assigned to this interrupt signal. For receive and transmit queues it is the index within the PF space (both PF and its VF queues are defined relative to the PF queue space). For CEQ, it is an index within the function space (VF queues are relative to the VF space and the PF queues are relative to the PF space). The NEXTQ_INDX is a NULL pointer for the last cause in the linked list (equals to 0x7FF).

Figure 38-58.Mapping Interrupt Causes to Interrupt Signaling



38.22.3.1 Interrupt Linked List Management

This section describes the required software flow for adding and removing an interrupt cause from an interrupt linked list.

38.22.3.1.1 Initial Setting of a Linked List

Associating interrupt causes to an interrupt signal can be programmed by software by any arbitrary order. As long as interrupts are not generated, the linked list is considered as static so any programming order is acceptable.

38.22.3.1.2 Adding an Interrupt Cause to an Active Interrupt

Once an interrupt is active, its linked list is considered as dynamic resource. Special programming ordering is required when adding an interrupt cause as follows:

- Program the cause register as required while the NEXTQ_INDX and NEXTQ_TYPE parameters point to the next cause in the linked list.
- Update the NEXTQ_INDX and NEXTQ_TYPE parameters in the previous cause pointing to the added one.
- Note that there is no need to disable the interrupt before these two steps.

38.22.3.1.3 Removing an Interrupt Cause from an Active Interrupt

Removing an interrupt cause from an interrupt linked list can be done in one of the following two cases:

Case 1 — The interrupt is disabled and it is guaranteed that this interrupt does not have any possible pending interrupts. In this case, the linked list is considered static and there are no special programing ordering. Software should update the NEXTQ_INDX and NEXTQ_TYPE parameters in the previous cause to the next cause, skipping the cause that is removed from the linked list. At this point, software can modify the cause register of the removed interrupt cause if required.

Case 2 — The interrupt is active. In this case, the linked list is considered dynamic resource and the software should follow a strict flow.

- Update the NEXTQ_INDX and NEXTQ_TYPE parameters in the previous cause to the next cause, skipping the cause that is removed from the linked list. In case it is the first cause in the linked list then update the FIRSTQ_INDX and FIRSTQ_TYPE parameters in the matched xxINT_LNKLSTx register, skipping the cause that is removed from the linked list.
- Clear the CAUSE_ENA flag in the matched xINT_xQCTL register.
- Software should wait a sufficient amount of time until it is guaranteed that hardware fetched the updated value of the previous cause. Waiting a sufficient amount of time could be done by scheduling a software interrupt and waiting for that interrupt that follows.
- At this point, software can modify the cause register of the removed interrupt cause if required.
- Note that there is no need to disable the interrupt before starting the previous sequence.

38.22.4 Interrupt Moderation

The 10 GbE controller is able to throttle interrupts in two layered methods: Interrupt Throttling (ITR) and Interrupt Rate limiting (INTRL). These methods are detailed in the following sub-sections.

38.22.4.1 Interrupt Throttling (ITR)

ITR is a mechanism that guarantees a minimum gap between two consecutive interrupts (other than possible jitter caused by handling the interrupts). The 10 GbE controller counts the time since the last interrupt is scheduled and compares it against the ITR setting. If an event associated with this ITR happens before the ITR expires, the interrupt assertion is delayed until the ITR expires. If the ITR expires before any event associated with this interrupt, the interrupt logic is armed and the interrupt can be asserted the moment the event happens. The ITR intervals-per-vector are programmed by the xxINT_ITRx registers (xx stands for PF or VF and x stands for 0 or N). The ITR is measure in units of 2 usec. Note that the ITR expiration sequence is triggered only when all the following conditions are met:

- The ITR timer is expired.
- The INTENA or the WB_ON_ITR flags in the matched GLINT_DYN_CTL register is set.
- The interrupt has credit(s) by the interrupt rate limiting logic described in the section that follows.

The 10 GbE controller supports three ITRs per MSI-X vector as well as a NoITR option. The interrupt causes are mapped to one of the ITRs by the ITR_INDX field (per cause). The ITR intervals can be programmed directly to the xxINT_ITRx registers or via the xxINT_DYN_CTLx registers (xx stands for PF or VF and x stands for 0 or N). It might be useful to set the initial values using the xxINT_ITRx registers and dynamic update by the xxINT_DYN_CTLx registers as explained in step #4 of the interrupt sequence explained in Section 38.22.1.3. When any ITR interval of an interrupt with pending event is expired and the INTRL (Section 38.22.4.2) credit is positive, hardware follows these steps:

- Clear the other ITRs of the same interrupt.
- Process all causes of the same interrupt (associated to all ITRs) as defined by the linked list (see Section 38.22.3).
- (*) See next section for a description of the interrupt rate limiting (INTRL)

38.22.4.2 Interrupt Rate Limiting (INTRL)

INTRL is a credit-based mechanism that limits the maximum average number of interrupts per second. The PF controls its interrupt rate limit by the PFINT_RATE0 and PFINT_RATEN registers. The PF also controls its VF's interrupt rate limit by the VPINT_RATE0 and VPINT_RATEN registers. The control parameters of these registers are detailed as follows:

- INTRL_ENA — Enable / disable option for the INTRL scheme. When disabled, interrupts can be generated without rate limiting control.
- INTERVAL — The INTRL is a 6-bit interval defined in 4 µs units that controls the time gap on which new interrupt credit is gained.

38.22.5 Write Back on Interrupts

Following packet reception, the status of completed descriptors are posted (write back) to host memory once every several packets or at ITR expiration. Similarly, following transmit packet DMA completion, its completed descriptors are reported to host memory as well. It is reported if the descriptors were programmed with active *RS* flag or at ITR expiration (either descriptor write back or header write back). In some cases queues should not generate interrupts. Associating these queues to an interrupt vector with active *WB_ON_ITR* and inactive INTENA generates no interrupts and maintains the previous rules for reported status on packet reception and transmission. This option is called write back on ITR.

Note: *WB_ON_ITR* and *INTENA* are mutually exclusive. Only one can be enabled for a given interrupt vector.

Note: Queues that should generate no interrupts and should not be reported on ITR completion should not be associated to any interrupt linked list.

38.23 Virtualization

38.23.1 Overview

I/O virtualization is a mechanism to share I/O resources among several consumers. For example, in a virtual system, multiple operating systems are loaded and each executes as though the entire system's resources were at its disposal. However, for the limited number of I/O devices, this presents a problem because each operating system might be in a separate memory domain and all the data movement and device management

has to be done by a Virtual Machine Monitor (VMM). VMM access adds latency and delay to I/O accesses and degrades I/O performance. Virtualized devices are designed to reduce the burden of VMM by making certain functions of an I/O device shared. Thus, they can be accessed directly from each guest operating system or Virtual Machine (VM).

Two modes to support operation in a virtualized environment were implemented in previous products:

1. Direct assignment of part of the port resources to different guest OSes using the PCI sig SR-IOV standard (also known as native mode or pass-through mode). This mode is called IOV mode in this chapter.
2. Central management of the networking resources by an IOVM or by the VMM (also known as software switch acceleration mode). This mode is called Next Generation VMDq mode.

The 10 GbE controller supports fully Next Generation VMDq mode and SR-IOV.

The 10 GbE controller supports two modes of offloads as part of Next Generation VMDq: VMDq1 and VMDq2.

In VMDq1, all the VMs are part of the same VSI and each VM is allocated a single queue. There is no replication of packets to different VMs and there is no forwarding of traffic from one VMDq1 VM to another.

In VMDq2, each VM is assigned a switch port (VSI). Thus there can be full switching between ports, including VM to VM switching and replication of multicast packets.

In a virtualized environment, the 10 GbE controller serves up to 256 virtual machines (VMs) per device; 128 of these can be directly assigned and any of them can be accessed in Next Generation VMDq mode. See Section 38.23.2.4 for details of the VFs and VMs resource allocation.

Most configurations and resources of the device are shared across VMs. The PF driver must resolve any conflicts in configuration between the VMs. For example, the PF driver should manage all the link configuration requests of the VFs.

Most of the virtualization offload capabilities provided by the 10 GbE controller, apart from the replication of functions defined in the PCI-sig IOV specification, are also part of Next Generation VMDq.

A hybrid model, where some of the virtual machines are assigned a dedicated share of the port and the others are serviced by an IOVM is also supported. This model can be used when some of the VMs run operating systems for which VF drivers are available. Such configurations can benefit from IOV. Others may run older operating systems for which VF drivers are not available and are serviced by an intermediary or models where VFs are assigned to VMs requiring a higher networking bandwidth. In this last case, the IOVM or VMM is assigned some VSIs and receives all the packets with MAC addresses of the VMs behind it.

The following section describes the support the 10 GbE controller provides for virtualization. This chapter assumes a single-root implementation of IOV and no support for multi-root.

38.23.1.1 Direct Assignment Model

The direct assignment support in the 10 GbE controller is built according to the software model defined by the SR-IOV specification.

The PF driver is responsible for the initialization and the handling of the common resources of the port. Other drivers (VF drivers) might read part of the status of the common parts but can not change it. The PF driver might run either in the VMM or in some service operating system. It might be part of an IOVM or part of a dedicated service operating system.

In addition, part of the non time-critical tasks are also handled by the PF driver. For example, access to CSR through the I/O space or access to the configuration space are available only through the master interface. Time critical CSR space, like control of the Tx and Rx queue or interrupt handling, is replicated per VF. It is directly accessible by the VF driver.

Note: In some systems with a thick hypervisor, the service operating system might be an integral part of the VMM. For these systems, each reference to the service operating system in this document refers to the VMM.

A channel is provided between the VF driver and the PF driver through the use of admin commands.

38.23.1.1.1 Rationale

Direct assignment's purpose is to enable each of the VMs to receive and transmit packets with minimum overhead. The non time-critical operations (such as init and error handling) can be done via the PF driver. In addition, it is important that the VMs can operate independently with minimal disturbance. It is also preferable that the VM interface to the hardware should be as close as possible to the native interface in non-virtualized systems in order to minimize the software development effort.

The main time critical operations that require direct handling by the VM are:

1. Maintenance of the data buffers and descriptor rings in host memory. In order to support this, the DMA accesses of the queues associated to a VM should be identified as such on the PCIe using a different requester ID.
2. Handling of the hardware ring (tail bump and head updates).
3. Interrupt handling.

The capabilities needed to provide independence between VMs are:

1. Per VM reset and enable capabilities.
2. TX QoS control.
3. Allocation of separate CSR space per VM.

The queue context creation, rate control and VF enable capabilities are controlled by the PF.

38.23.1.2 Virtualized System Overview

The following diagrams show the various elements involved in the I/O process in a virtualized system. Figure 38-59 shows the flow in software Next Generation VMDq operation mode and Figure 38-60 shows the flow in IOV mode.

This document assumes that in IOV mode, the software device driver on the guest operating system is aware that it works in a virtual system (para-virtualized) and there is a channel between each of the VM drivers and the PF driver allowing message passing (such as configuration request or interrupt messages). This channel might use the mailbox implemented in the 10 GbE controller or any other means provided by the VMM vendor.

Figure 38-59.VMDq System

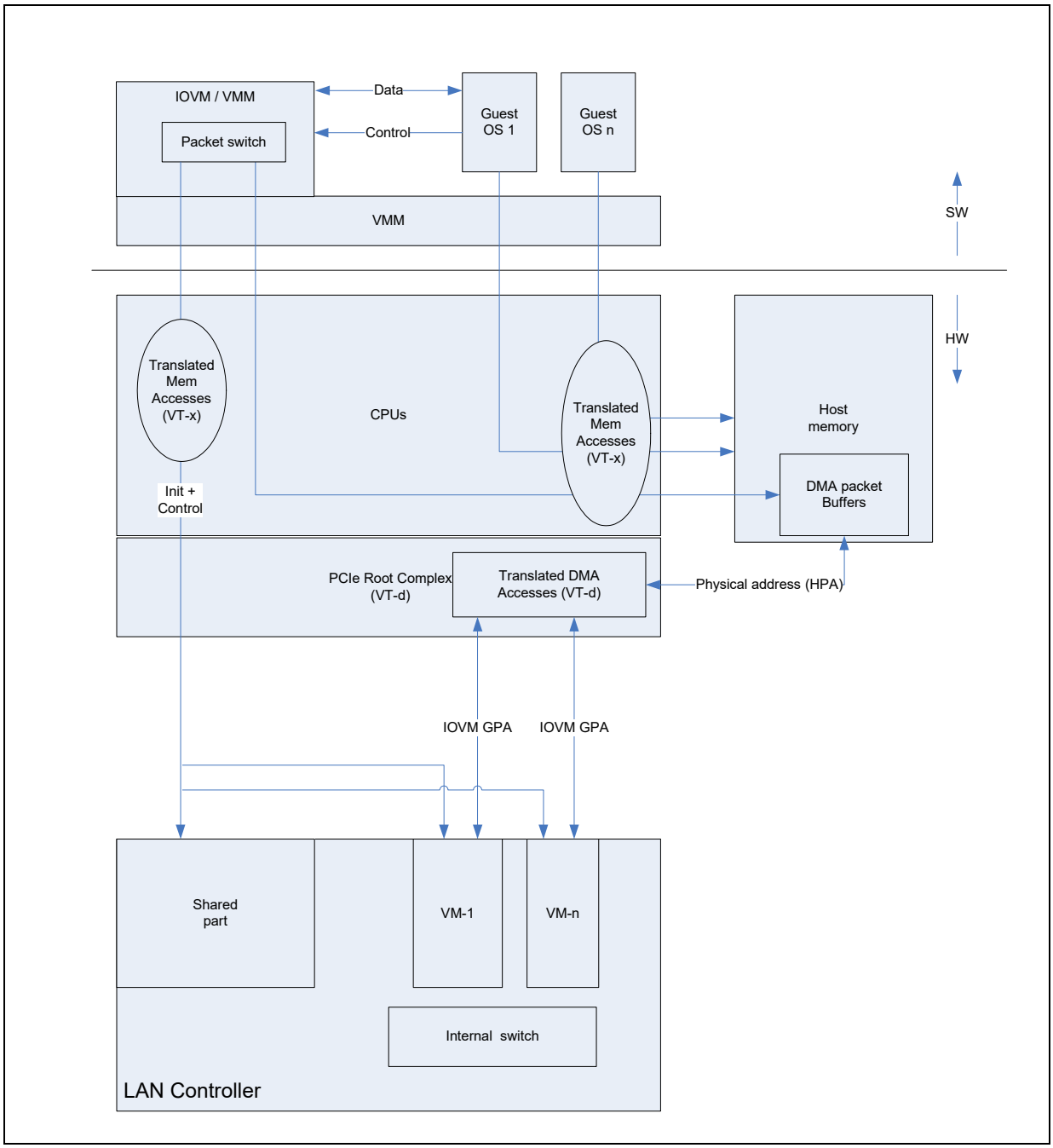
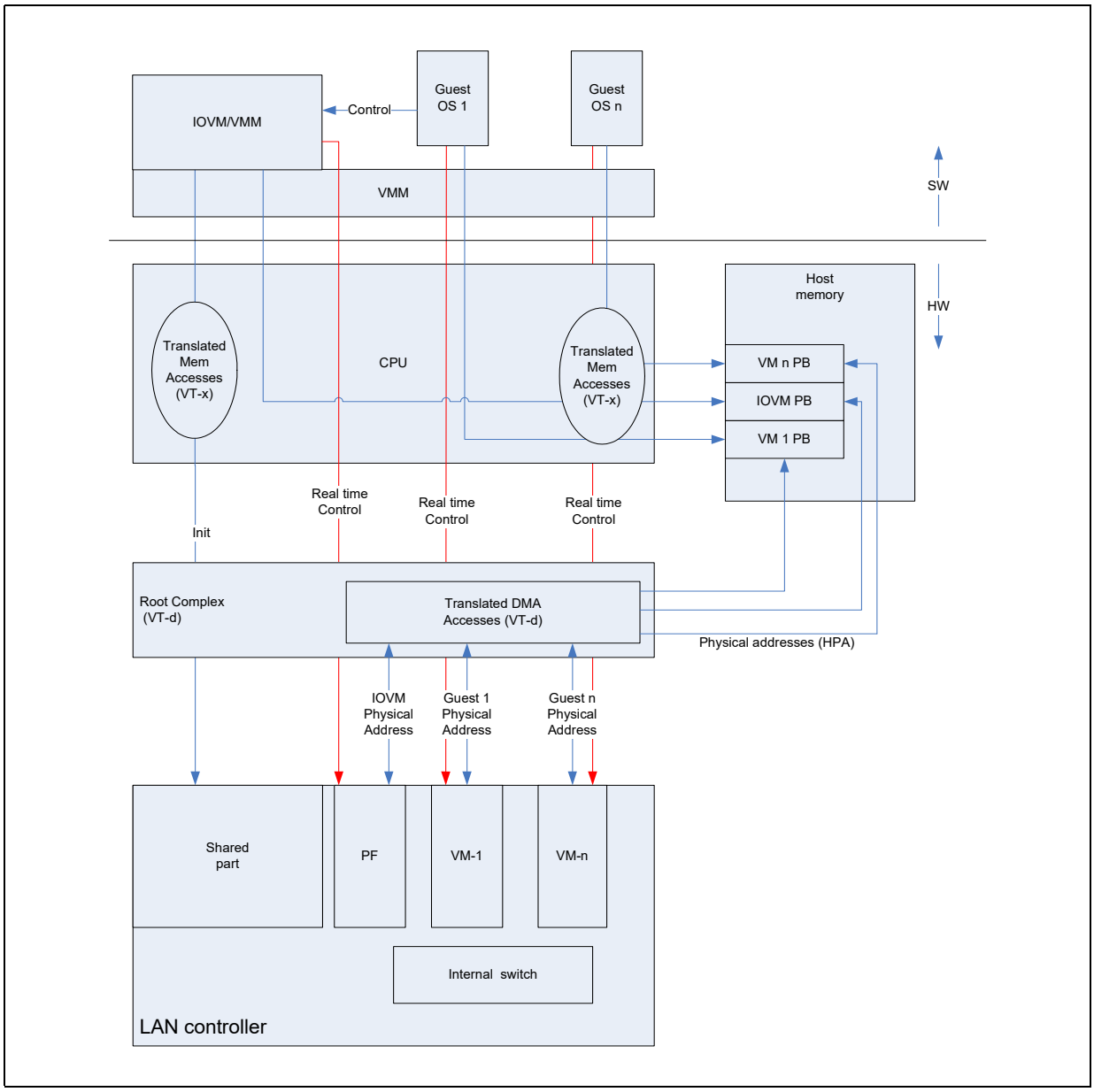


Figure 38-60.SR-IOV-Based System



38.23.1.3 Virtualization Supported Features

The 10 GbE controller supports a super set of the virtualization features supported in previous products. Table 38-283 lists the virtualization features of the 10 GbE controller with these of the 82599.

Table 38-283.Product Name Versus the 82599 Virtualization Support

Feature	10 GbE Controller Support	82599 Support
Direct attach (SR-IOV) features		
SR-IOV Support	Yes	Yes
ATS Support	No	No
VF-to-PF Mailbox	Yes	Yes
Max Number of Virtual Functions	128 per device (globally)	64 per port (single queue)
VMDq features		
Max number of Queues Allocatable to VMs	1536	128
Max Number of Queues per VF	16	8
Max Number of Queues per VMDq2 VSI	16	8
Max Number of Queues per VMDq1 VM	1 (A VMDq1 VSI may support multiple VMs).	8
Max Number of VMDq2 Ports	256 per device (globally)	64 per port (single queue)
MAC Addresses	1024 per device (globally)	128 per port
VLAN Tags	512 per device (global)	64 per port
Queuing to Pool Method	SA, VLAN pairs or SA or VLAN	SA or VLAN or (SA and VLAN)
RSS per VF	Yes	No (Single RSS used for all VFs).
DCB in VF	Yes	Yes
Switching Modes	VEB, VEPA, EVB	VEB
VM-to-VM Switching	Yes	Yes
Broadcast and Multicast Replication	Yes	Yes
MAC and VLAN Anti-spoof Protection	Yes	Yes
VLAN Filtering	Global and per pool	Global and per pool
Drop If No Pool	Yes	Yes
per Pool Statistics	Yes	Yes
Per Pool Offloads	Yes	Yes
Mirroring	Yes	Yes
Long Packet Filtering	Global and per pool	Global and per pool
Promiscuous Modes per VM	VLAN, Multicast, Unicast	Multicast

38.23.1.3.1 Enablement of Virtualization Features

Virtualization feature enablement can be controlled using soft SKU. Table 38-284 lists the way to enable each of virtualization feature.

Table 38-284.Virtualization Features Enablement

Technology	Included features	Enablement
SR-IOV		Set the *GLPCI_CAPSUP.IOV_EN* bit.
VEB	VEB, VEPA, VMDq2, mirroring.	Set the *GLGEN_STAT.VTEN* bit.
802.1BR/802.1Qbg	Port extender creation. CDCP handling.	Set the *GLGEN_STAT.EVBEN* bit.

38.23.2 SR-IOV Implementation

38.23.2.1 IOV Concepts

The SR-IOV specification defines the following entities in relation to I/O virtualization:

1. VVM — A VM to which I/O resources are assigned.
2. A PCIe device — The physical device that might contain a few physical functions. In this case, the 10 GbE controller.
3. PF — A function representing a physical instance. I in this case, a PCIe function that represents a physical port or a logical port. The PF driver is responsible for the configuration and management of the shared resources in the function.
4. VF — A part of a PF assigned to a VM.

38.23.2.2 IOV Control

In order to control the IOV operation, the physical driver is provided with a set of registers and capabilities. These include:

1. The *PF_VIRT_STATUS* register — Indicates whether SR-IOV is enabled and the number of VFs enabled.
2. Driver-to-driver communication provided by the virtualization admin commands.
3. Switch and filtering control admin commands.
4. Reset indications and traffic enables registers per VF using the *GLGEN_VFLRSTAT.VFLRE* bit indicating that a VFLR reset occurred in one of the VFs. When the *GLGEN_VFLRSTAT.VFLRE* bit is set for a given VF, this VF cannot send or receive packets. The PF should clear this bit to enable a VF.
5. Malicious driver detection (described later in this section).

Refer to the flow used to configure a function when SR-IOV is enabled.

38.23.2.2.1 Interrupt on Misbehavior of VM (Malicious Driver Detection)

The 10 GbE controller can protect itself from faulty or malicious behavior on the part of a VM driver. This is done by checking for specific illegal events.

The bits that enable these checks are grouped into three categories: RX checks, TX descriptor checks and TX data checks. Individual checks are controlled by global registers GL_MDCK_RX, GL_MDCK_TCMD and GL_MDCK_TDAT respectively. If a check is not enabled (the bit is cleared) and such an event happens, the monitoring hardware will not react to the condition and the device may malfunction.

Table 38-285, Table 38-286 and Table 38-287 list the checks in each group, the register bits used to enable them and the value read from the MDET registers when an event is detected.

The default values for these registers are loaded from NVM.

The checks apply both to VF and PF activity. If such behavior is detected, the queue is stopped and an interrupt is sent to the PF that owns the function. This means that a PF will be interrupted both on events from it's queues and on events from it's VFs queues. Tracking which functions have caused an event is done by reading each function's VP_MDET_RX and VP_MDET_TX registers for VFs or PF_MDET_RX and PF_MDET_TX for PFs. The registers are cleared by writing 1s to them. After such an event the function needs to reset the queue. Since the malicious driver event indication in the *VP/PF_MDET_TX* registers is per function and the details of the event in the *GL_MDET_TX* register is common to all functions, it may not be simple to determine which queue caused the event, therefore the driver may elect to reset the whole function.

TX Data protection is different in that it does not stop the queue; it drops the offending packet.

Two global debug registers (GL_MDET_TX and GL_MDET_RX) record the function number event ID and queue number for the first event observed on TX and RX, respectively. These registers are cleared by writing 1s.

38.23.2.2.2 TX Data Checks (GL_MDCK_TDAT register)

The table below describes the checks that are done by the 10 GbE controller on data fetches for TX data.

Table 38-285.Data Checks on Fetches for Tx Data

Check type	GL_MDET_TDAT bit	Description
Bad Header Length	2 MAL_LENGTH_DIS	Malicious detection of bad header length in the transmit descriptor.
Illegal Commands	3 MAL_CMD_DIS	Malicious detection of bad combination of commands in the transmit descriptor.

38.23.2.2.3 TX Descriptor Validity Checks (GL_MDCK_TCMD register)

The table below describes the checks that are done by the 10 GbE controller on TX descriptor.

Table 38-286.Tx Descriptor Validity Checks

Check type	GL_MDCK_TCMD bit	Event ID on MDET_TX register	Description
TX descriptor address	0 DESC_ADDR	10	descriptor fetch failed
Too many buffers	2 MAX_BUFF	14	Single send packet or large send segment is spread on more than 8 data descriptors
Too many header buffers	3 MAX_HEAD	15	LSO with header spanning more than 3 buffers.
No header buffers	4 NO_HEAD	24	Zero header length when large send offload is enabled
Wrong Size	5 WRONG_SIZE	16	A single send packet or large send segment that is not between min / max sizes defined in the GLTLAN_MIN_MAX_PKT register

Table 38-286.Tx Descriptor Validity Checks

Check type	GL_MDCK_TCMD bit	Event ID on MDET_TX register	Description
Tail update bigger than ring size	7 ENDLESS_TX	21	endless transmit ring When this flag is cleared, endless transmit ring option is enabled. When this flag is set, endless transmit ring is considered malicious event.
Bad LSO packet length	8 BAD_LSO_LEN	16	TSO Total Length not equal sum of buffers or equals to zero
Bad LSO MSS	9 BAD_LSO_MSS	18	LSO with MSS is smaller than 64 or larger than 9668.
More than seven context descriptors	12 M_CONTEXTS	21	7 or more consecutive non-data descriptors are fetched in a transmit queue.
Bad descriptor sequence	13 BAD_DESC_SEQUENSE	18	Bad descriptors sequence. Include: MSS min/max; Max number of non-data descriptors; DIF/DIX descriptors (enabled by bit 16); Flex descriptors (enabled by bit 17).
Descriptor type	14 BAD_DESC_TYPE	21	Illegal descriptor type used
No Packet	15 NO_PACKET	21	Tail update that does not contain at least one full packet.
Disable DIF / DIX descriptors	16 DIS_DIF_DIX	11	DIF/DIX descriptors are considered malicious descriptors.
Disable Flexible Descriptors	17 DIS_FLEX	20	Flexible descriptors are considered malicious descriptors.
Zero Bsize	18 ZERO_BSIZE	23	A descriptor with a zero BSIZE is found

38.23.2.2.4 RX Checks (GL_MDCK_RX register)

The table below describes the checks that are done by the 10 GbE controller on data fetches for RX.

Table 38-287.Rx Data Fetch Checks

Check type	GL_MDET_RX bit	Event ID on MDET_RX register	Description
RX descriptor address	0 DESC_ADDR	1	descriptor fetch failed

38.23.2.3 Hardware Resources not Assigned to VFs

Certain device capabilities are not exposed to VFs, neither directly nor via a PF. The following features are available only to PFs or controlled solely by the PF:

- Power management and WoL are PF resources and are not supported per VF.
- Link Control - The link is a shared resource and as such is controllable only by the PF. This includes PHY settings, speed and duplex settings, flow control settings, etc. Flow control packets are sent using the station MAC address stored in the EEPROM. The watermarks of the flow control process and the time-out value are also controllable by the PF only. In a DCB environment, the parameters of per-TC-flow control and the ETS settings are also PF responsibilities.
- Configuration of Control Domains and Profiles is done solely by the PFs and is not exposed to VFs.
- DCB policy and configuration of the device (either via DCBx or otherwise) is not open to VFs' control. A VF may still associate its Queues with specific TCs, may request (via its PF) for Tx Scheduler nodes to be added under specific VFs, and to set the appropriate QoS fields in its transmitted packets.

- Special Filtering Options - Save Bad Packets is a debug feature. As such, Save Bad Packets is available only to the PF. Bad packets are forwarded to a control VSI and thus should not be seen by a VF in regular operation.
- Reception of long packets is controlled separately per queue. As this impacts flow control thresholds, the PF should be made aware of the decisions of all VMs. Because of this, the setup of large send packets is centralized by the PF and each VF might request this setting.

38.23.2.4 Hardware Resources Assignment to VFs

Table 38-288 lists how the 10 GbE controller shared resources are distributed between different VFs. Details for each type of resource can be found in the relevant section of this document.

Table 38-288.VF Resource Allocation (Sheet 1 of 2)

Resource	Allocation method
General Resources	
Tx and Rx Queues	Dynamic allocation. Each VF can have a different number of queues (up to 256). Each PF allocates the queues to its VFs and their VSIs. The allocation is done using the *Add VSI* or *Update VSI* Admin commands. Queue initialization is done by the PF. Following initialization, the VF manages its queues for Tx/Rx operation.
Admin Queues	An admin queue is allocated to each VF to communicate with its PF. The queue accesses a shared mailbox in the device.
Interrupt Causes and Vectors	Allocated according to total number of exposed VFs.The interrupt causes and vectors are allocated to VFs only and not to VMs that are controlled by the PF. These VMs use the PF interrupts. 512 vectors allocated evenly among VFs. Assignment of interrupt causes: • Causes for Tx queues, Rx queues and PE queues to interrupt vectors are done by each VF. • Other causes are mapped to vector 0. • VFs operate with MSI-X interrupts only.
Transmit Scheduling	A VF is represented in the scheduling tree through its VSI(s). Nodes associated with such a VSI might be configured with all supported attributes, such as guaranteed bandwidth, rate limiting, and arbitration scheme. Configuration is done by the PF and not directly by the VF.
Stateless Offloads	Tx — All the regular transmit offloads like checksum and TSO are available to VFs. Enabling these offloads is done by the PF as part of the queue initialization process. Rx — All the regular receive offloads like checksum and header split are available to VFs. Enabling these offloads is done by the PF as part of the queue initialization process.
Statistics	The VF is not directly exposed to statistics counters. If a VF needs to get statistics for its traffic, it is done via its PF.
IEEE 1588	IEEE 1588 is a per link function and thus is controlled by the PF driver. VMs have access to the real time clock register via CSRs.
RDMA Resources	
RDMA capabilities	32 VFs might have RDMA capabilities. The *GLPE_VFPFMAP* table is used to map RDMA enabled VFs to the RDMA register sets.
QPs/CQs/CQEs/SDs	According to the profile of the device, the QP/CQ/CQE/SD resources are distributed evenly between the PFs supporting iWARP. The resources of a PF might be evenly distributed between the VFs supporting RDMA.
CSRs	All VFs have a set of registers to control RDMA functionality. However, only 32 of these sets are active. The RDMA register set contains registers in regular VF CSR space. These registers are accessible to the VF driver and provide a common doorbell page accessible to all the user space processes in the VF.
Switching Resources	

Table 38-288.VF Resource Allocation (Sheet 2 of 2)

Resource	Allocation method
Switch Resources	Switch configuration and resource allocation are not done by a VF directly, but through its PF.
VSIs	Up to 256 VSIs can be directly supported by the 10 GbE controller. These VSI resources are distributed between the different PFs dynamically. Each VF is guaranteed to receive at least one VSI. The PF might allocate multiple VSIs to a VF. The allocation is done using the using the *Add VSI* Admin command. There is no limit on the number of VSIs that can be assigned to a VF up to the number of Tx/Rx queue pairs the VF is allocated.
Unicast MAC Addresses, Multicast Addresses, VLAN tags	The PF driver is responsible for the resource allocation to its VFs. The 10 GbE controller does not manage the per-VF resources of the switch. The PF assigns filters to VFs VSI using Admin commands.
Filtering Resources	
RSS	Directly controlled by each VF. RSS per VSI with either small or medium RSS table.
Flow Director	Flexible population of FD entries with VF rules. Programming is done through the PF.
Quad Hash Filtering	Each PF allocates a private memory region for itself and its VFs. Each PF populates its private memory region with its VFs' entries. The on-die quad-hash filter caches entries for both PFs and VFs.

38.24 Data Center Bridging (DCB)

This chapter assumes the reader is familiar with the following specifications:

- IEEE P802.1Qbb-2011 — Priority-based Flow Control (PFC) specification
- IEEE P802.1Qaz-2011 — Enhanced Transmission Selection for Bandwidth Sharing Between Traffic Classes (ETS) specification
  - — DCB Center Bridging Exchange Protocol (DCBX) specification refers to Clause 38 in the ETS specification
- IEEE Std 802.1AB-2009 — Link Layer Discovery Protocol (LLDP) specification

38.24.1 Receive Path DCB

38.24.1.1 Receive Path Enhanced Transmission Selection (ETS)

38.24.1.1.1 Identifying Low Latency (LL) Traffic in Receive

Only two types of traffic are identified along the receive data path, LL and Bulk (B). This approach requires a priority differentiation between low latency and bulk traffic. TC indexes are used for that purpose according to a setting made in PRTDCB_RETSC.LLTC bitmap.

The entity that runs DCBX must configure all non-ETS TCs as LL traffic and only these TCs.

38.24.1.1.2 UP-to-TC in Receive

Mapping of UP-to-TC in Receive

Register PRTDCB_RUP2TC controls the mapping of incoming packets to TCs according to the UP field they carry. The same mapping is used for packets received from the wires and for those looped back internally. It defines on the account of which TC a packet is stored in the Rx packet buffer, and which UPs bits are set in the PFC XOFF/XON frames issued to the link partner when the filling state of a TC requires it. See Section 38.24.1.1.6.

Packets received with no 802.1p tag are also mapped to a TC according the setting made in NOVLANUP field of PRTDCB_RUP register that is applied as an input to the UP to TC mapping table of PRTDCB_RUP2TC register. To ensure that LLDP and other MAC Control packets are not dropped internally by the device before they reach EMP or the host, it is recommended that the PRTDCB_RUP.NOVLANUP field be set to the no-drop UP with the lowest index.

The *GL_SWT_L2TAGCTRL.HAS_UP* bit indicates per tag type if its UP is candidate for DCB usage. The UP for DCB is taken from the first tag encountered in the packet that has this bit set. The same method is used both for receive and loopback traffic. This bit should be set for S-tags and VLAN (internal and external) EtherTypes.

Remapping of the UP Field in Receive

The remapping scheme applies also to the traffic destined to EMP (or MC), which is represented in the tables by its own four VSI numbers.

38.24.1.1.3 Receive Flow for ETS

Receive Linked Lists

The total available PCIe bandwidth allocated for traffic reception is *initially* allocated among the LAN ports, and *secondly* allocated among TCs (and among the different UPs attached to it) for each LAN port separately. Bandwidth allocated to a TC) over a LAN port might differ to the traffic allocated to the same TC index over another LAN port. Since the number of TCs to be supported over a port might vary from 1 to 8 as per DCBX exchange, the Rx packet buffer is organized in a fixed manner into 32 receive linked lists, one for each UP over each LAN port.

Receive Arbitration Levels

The two levels of bandwidth allocation previously described, one across LAN ports and one across the TCs of a LAN port, induce two levels of arbitration. One Port Round Robin (PRR) arbiter that is used to select the Rx LAN port to be handled and one ETS arbiter that is used to select the TC to be handled within the selected LAN port. Since both arbitration schemes do not deal with absolute throughput allocation, no PCIe bandwidth should be wasted in case for some time some LAN ports or TCs offer less workload than allocated to them. Unused bandwidth by a TC is proposed first to other TCs over the LAN port, and secondly to other LAN ports.

38.24.1.1.4 Receive ETS Arbiter

The 10 GbE controller's receive ETS arbiters determine the order in which the per UP linked lists of a port are serviced at the Rx packet buffer’s exit. Note that within each linked list, packets are drained in the order they arrived, indifferently whether they arrived from the Rx port or from the Tx loopback path of the port.

The arbitration algorithm between the linked lists is either strict priority or packet-based round robin.

Two operating modes are supported according to setting made in using the *NON_ETS_MODE* bit in the PRTDCB_RETSC register:

1. **Strict Priority (SP) Mode** — TCs are served in a strict priority order between them starting from the TC with the higher index until it has no workload, and so forth down to the TC with the lowest index.
2. **Round Robin (RR) Mode** — TCs are served in a packet-based round-robin manner between them until no workload is left to any of them.

When several UPs are mapped (by DCBX) to the same TC index in PRTDCB_RUP2TC register, two operating modes are supported:

1. **Strict Priority (SP) Mode** — This mode is selected by setting *UPINTC_MODE* field to 0b for the corresponding TC index in PRTDCB_RETSTCC register array. UPs are served in a strict priority order between them within the total bandwidth allocated to the TC. Each time the TC is selected by the ETS arbiter, the highest UP is served first until it offers no traffic, and so forth down to the lowest UP of the TC. This is the default operating mode.
2. **Round Robin (RR) Mode** — This mode is selected by setting *UPINTC_MODE* field to 1b for the corresponding TC index in PRTDCB_RETSTCC register array. UPs are served in a packet-based round-robin manner between them within the total bandwidth allocated to the TC. Each time the TC is selected by the ETS arbiter, a different UP offering workload is served among the UPs mapped to the TC, and so forth cyclically.

According to the total number of enabled ports and to the number of TCs of the port, these fields are configured by the DCBX agent as follows:

- Quad port configuration
  - Ports with 1 TC:
    - PRT_SWR_PM_THR.THRESHOLD = 0xA
    - PRTDCB_RPPMC.RX_FIFO_SIZE = 0x1E
  - Ports with 2 to 4 TCs:
    - PRT_SWR_PM_THR.THRESHOLD = 0xA
    - PRTDCB_RPPMC.RX_FIFO_SIZE = 0xA
  - Ports with 5 to 8 TCs:
    - PRT_SWR_PM_THR.THRESHOLD = 0xA
    - PRTDCB_RPPMC.RX_FIFO_SIZE = 0x5
- Dual port configuration
  - Ports with 1 TC:
    - PRT_SWR_PM_THR.THRESHOLD = 0x14
    - PRTDCB_RPPMC.RX_FIFO_SIZE = 0x3C
  - Ports with 2 to 4 TCs:
    - PRT_SWR_PM_THR.THRESHOLD = 0x14
    - PRTDCB_RPPMC.RX_FIFO_SIZE = 0x14
  - Ports with 5 to 8 TCs:
    - PRT_SWR_PM_THR.THRESHOLD = 0x14
    - PRTDCB_RPPMC.RX_FIFO_SIZE = 0xA

- Single port configuration
  - 1 TC:
    - PRT_SWR_PM_THR.THRESHOLD = 0x28
    - PRTDCB_RPPMC.RX_FIFO_SIZE = 0x78
  - 2 to 4 TCs:
    - PRT_SWR_PM_THR.THRESHOLD = 0x28
    - PRTDCB_RPPMC.RX_FIFO_SIZE = 0x28
  - 5 to 8 TCs:
    - PRT_SWR_PM_THR.THRESHOLD = 0x28
    - PRTDCB_RPPMC.RX_FIFO_SIZE = 0x14

Depending on the locality of Rx traffic, while in quad port configuration, ports with 5 to 8 TCs do not achieve the Rx performance goals.

38.24.1.1.5 Receive Low Latency Prioritization (Rx-LLP) Arbiter

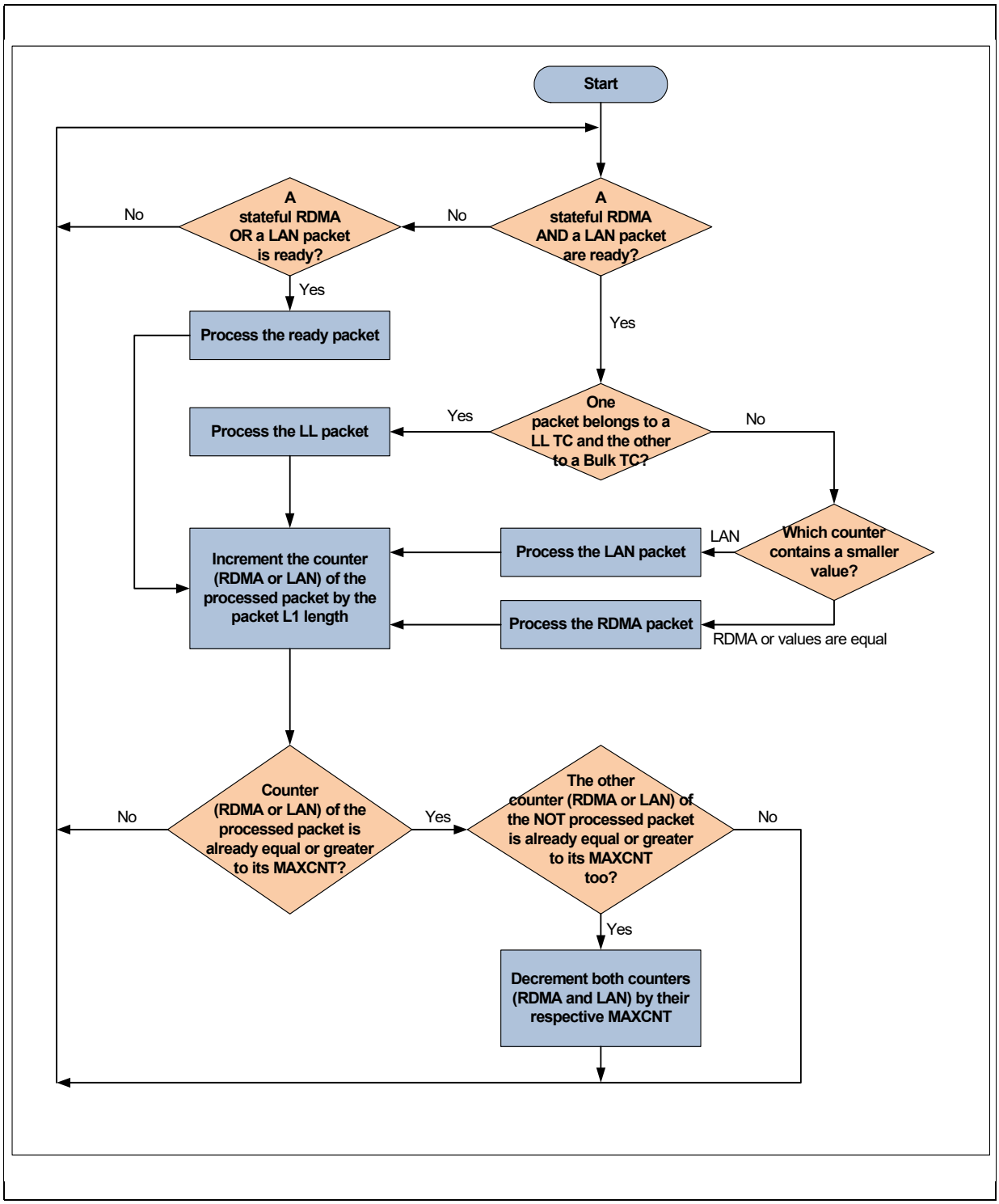
Without this last level DCB arbitration, the simplest low latency benchmark test would fail: a single LAN low latency packet is received while a number of RDMA packets that were under processing get ready to be sent to host. In such a situation the low latency packet suffers from longer delays than usual, even if it get prioritized by the PRR/WSP arbiters.

The following algorithm is run to decide which packet should be sent to host between the two: a packet that is ready at the head of the RDMA pipe or that at the head of the LAN pipe. It requires handling a 1-byte counter per each pipe RDMA and LAN (read accessible via GLDCB_RLLPSRDMA / GLDCB_RLLPSLAN register, respectively) to ensure the following:

1. Packets from either pipe (RDMA or LAN) that belong to a LL TC are always strictly prioritized each time they compete with bulk packets on the other pipe.
2. Each time the competing RDMA/LAN head of pipe packets belong to the same type, either both LL or both bulk, they are serviced in a 50%/50% round-robin ratio relatively to the amount of bytes already serviced per RDMA/LAN type.

Note: Byte counts can overflow the configured limit (MAXCNT) up to one MFS (minus one byte) to get aligned with packets boundary. From then, they saturate until the other counter reaches its upper limit as well.

Figure 38-61.Operation of the Rx Low Latency Prioritization Arbiter



b. Delay of one MFS Ethernet frame due to the last minute decision made at Rx-LLP arbiter.

Note that different latency delays are measured for LL LAN traffic depending if RDMA pipe is enabled or not. This is due to the extension of LAN pipe monitor

required when operating with two separate pipe monitors. Also, in case low latency and bulk TCs are mixed over RDMA traffic, low latency RDMA traffic suffers from the deep RDMA Rx processing path that is shared between them. Ideally, RDMA Rx processing path should have made possible for low latency flows to bypass other flows, but this is not supported by the 10 GbE controller.

3. Low latency prioritization — When one pipe (RDMA/LAN) is fed only with LL traffic, and the other only with bulk traffic, then LL traffic is served (first). When both pipes (RDMA and LAN) are fed with traffic from the same type, either both LL or both bulk, then both pipes must be equally served in bytes amount, in a round robin manner.
4. Impact of packet size — if PCIe bandwidth is much lower than LAN bandwidth (TC buffers are full), and TCs have similar bandwidth allocation, but asymmetric.

38.24.1.1.6 Rx ETS Configuration Rules

PRTDCB_RPRRC are dynamic registers that are auto-programmed by the device. The device retrieves different default values from NVM, one for each the link speed. Each time a port changes its link speed, the corresponding default value is loaded by hardware.

PRTDCB_RUP2TC, PRTDCB_GENC,PRTDCB_RPPMC.RX_FIFO_SIZE, and PRT_SWR_PM_THR are also dynamic registers. Refer to the flow described in Section 38.24.3.3.1 for the way they are modified.

All other Rx-ETS settings described in Section 38.24.1.1 are static. They must not be modified at run time. They are loaded from the NVM only at initialization/reset time.

38.24.1.2 Receive Path Priority Flow Control (PFC) and Rx Packet Buffer (RPB)

This section describes the guaranteeing of no packet loss inside the 10 GbE controller for the PFC-enabled traffic, whether it is received from the LAN wires or from the internal switch. Since this feature is tightly related to the allocation of receive packet buffer to the TCs, the partitioning scheme of the Rx packet buffer is described first.

RPB has a maximum byte capacity of 968 KB and a maximum packet capacity of 7744 packets (968 KB / 128B).

TCs are classified in two types relatively to PFC: no-drop (loss less) TCs, and drop TCs (best effort delivery). PRTDCB_TC2PFC bitmap controls the use of PFC by a TC over a link. The setting applies for both Tx and Rx directions.

Refer to Ethernet flow control basic operation section, which is relevant whether or not the device can operate in DCB mode.

38.24.1.2.1 Requirements on Rx Packet Buffer

DCB requires that separate queuing resources be allocated to each TC along the entire path from the source to the destination node. This guarantees the good functioning of the ETS scheme, and mainly when it comes to prioritize LL TCs over others.

- Handle one linked list of packets per each UP, per port. The list maintains arriving order of packets inside a flow. UPs are attached to TCs in Rx according to settings made in PRTDCB_RUP2TC register.
- No crosstalk be sensed between the LAN ports, at least on regular basis and if the average incoming packet size is greater or equal to 128 bytes.

- Be capable to allocate fully independent buffers for up to five basic TCs per port, as follows:
  - 2 no-drop TCs, one for any other traffic type that requires PFC-enabled and 9.5 KB Jumbo frames (such as iSCSI traffic).
  - 3 best effort delivery TCs (drop TCs) for which PFC is disabled and which carry 9.5 KB jumbo frames.
- Be capable to allocate fully independent buffers for up to 8 best effort TCs per port.
- Three extra TCs beyond the five basic TCs previously described or any other TC settings up to eight TCs per port are supported. It can however be handled via some level of buffer sharing between them.
- Partitioning of Rx packet buffers across the enabled LAN ports is set equally at init time, according to NVM settings.
- Partitioning of the amount of Rx packet buffers allocated to a port is done dynamically according to the following parameters:
  - Number of TCs supported by the port, as per DCBX
  - Number of PFC-enabled TCs over the port, as per DCBX
- Support LPI, and especially the worst case Tx LPI exit time of 17.38 µs while Rx is not in LPI state.

38.24.1.2.2 Normal Partitioning of Rx Packet Buffer

- Eight dedicated pools per port, one per TC. The size of each dedicated buffer is set by PRTRPB_DPS array of registers. Setting a null size to a dedicated packet buffer is allowed (such as for non-used TCs or for the extra TCs), it is equivalent to state that no dedicated buffer is allocated to the TC.
  - Normal partitioning assumes at least the five basic TCs per port get dedicated buffers.
  - Eight dedicated filling counters per port, one per TC.
  - Eight high watermarks per port, one per TC. It is set by PRTRPB_DHW array of registers.
  - Eight low watermark per port, one per TC. It is set by PRTRPB_DLW array of registers. At any time, it must be set to a lower value than the corresponding high watermark. Setting a null low watermark is allowed.
- One shared pool per port. The size of the shared buffer is set by PRTRPB_SPS register. Setting a null size to the shared buffer of a port is allowed, it is equivalent to state that no shared buffer is allocated to the port. This might be the case when only the five basic TCs are supported on a port.
  - One shared pool filling counter per port.
  - One high watermark per shared buffer. It is set by PRTRPB_SHW register.
  - One low watermark per shared buffer. It is set by PRTRPB_SLW register. At any time, it must be set to a lower value than the corresponding high watermark. Setting a null low watermark is allowed.
  - Eight occupancy counters per port, one per TC, to track the amount of bytes of the TC stored on the account of the shared buffer of the port.
  - Eight high thresholds per port, one per TC. It is set by PRTRPB_SHT register. The threshold is relative to the corresponding occupancy counter.
  - Eight low thresholds per port, one per TC. It is set by PRTRPB_SLT register. At any time, it must be set to a lower value than the corresponding high threshold. Setting a null low threshold is allowed. The threshold is relative to the corresponding occupancy counter.

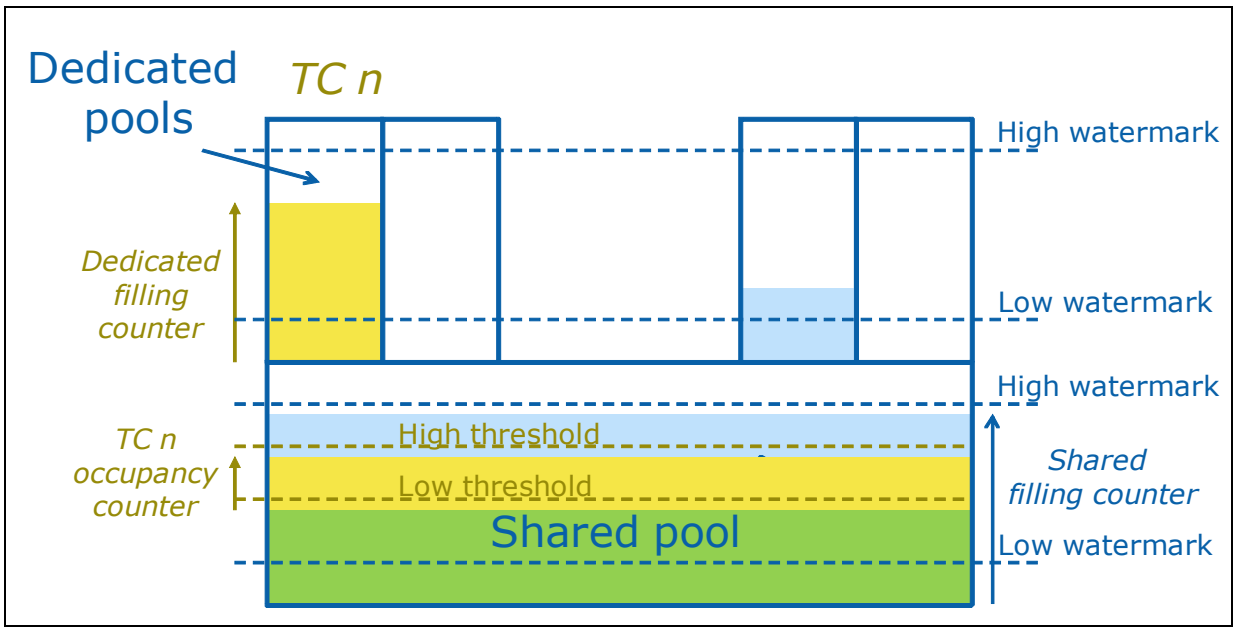
- One packet buffer is allocated per port. Setting a null size to the port packet buffer is allowed (like for disabled ports), it is equivalent to state that no packet buffer is allocated to the port.
  - Normal partitioning assumes the sum of the four per-port buffer sizes equals to the total size of the Rx packet buffer in the 10 GbE controller. For example, 968 KB.
  - The size of the per -ort packet buffer is computed internally as the shared pool size plus the sum over all the dedicated pools size of the port.
  - One per port filling counter.
- One global filling counter for the entire Product Name's receive packet buffer.
  - One global high watermark for the entire device, relative to the global filling counter. It is set by GLRPB_GHW register.
  - One global low watermark for the entire device, relative to the global filling counter. It is set by GLRPB_GLW register. Normal partitioning assumes it is set to zero.

Note: The receive packet buffer accounting granularity is 16 bytes, a physical memory line. To all buffer sizes, threshold, and watermarks registers defined in bytes, the device discards the four LS bits written by software.

The maximum capacity in bytes of the receive packet buffer is 968 KB, provided that the average size of the packets stored is equal or greater than 128 bytes.

- One global packet counter for the entire Product Name's receive packet buffer.
  - One packet high watermark for the entire device, relative to the global packet counter. It is set by GLRPB_PHW register. It is used to reflect the implementation limit, and must normally be set equal or lower than full Rx packet buffer size (968 KB) divided by 128 bytes.
  - One packet low watermark for the entire device, relative to the global packet counter. It is set by GLRPB_PLW register.

Figure 38-62.Normal Partitioning of the Per-port Rx Packet Buffer



38.24.1.2.3 Receive Drop Policy

When receiving a packet attached to a drop TC, either from the LAN wire or from the internal loopback path, the packet is dropped if ANY of the following conditions occur when storing it into the Rx packet buffer:

a. The dedicated pool (PRTRPB_DFC) is filled above or equal to its high watermark (PRTRPB_DHW), and the TC occupancy counter has reached its high threshold.

b. The dedicated pool (if any, PRTRPB_DFC) is filled above or equal to its high watermark (PRTRPB_DHW), and the shared pool filling counter has reached its high watermark and has not returned to the low watermark yet.

c. The dedicated pool (if any, PRTRPB_DFC) is filled above or equal to its high watermark, and the global filling counter has passed above its global high watermark (GLRPB_GHW) and has not returned to the global low watermark yet.

d. The global packet counter has passed above its packet high watermark.

Note: The hysteresis between high and low watermarks is required to avoid starving the no-drop TCs, which can use only the shared pool account with drop TC traffic.

When receiving a packet, even for a no-drop TC, either from the LAN wire or from the internal loopback path, the packet is dropped if there would be no place for storing it entirely in the global Rx packet buffer.

38.24.1.2.4 Policy for Issuing XOFF

When receiving a packet attached to a no-drop TC, either from the LAN wire or from the internal loopback path, a PFC XOFF notification is issued onto the wire and internally to the loopback path for all the UPs mapped to the TC if ANY if the following condition occurs:

a. The dedicated pool (if any, PRTRPB_DFC) is filled above or equal to its high watermark (PRTRPB_DHW).

b. No dedicated pool for the TC (PRTRPB_DPS=0b), and the TC occupancy counter (which is relative the shared pool only) has reached its high threshold (PRTRPB_SHT).

c. No dedicated pool for the TC (PRTRPB_DPS=0b), and shared pool filling counter is above its high watermark (PRTRPB_SHW). In this case XOFF is only concerned with all the no-drop TCs of the port that do not have a dedicated pool.

d. Global filling counter has reached its global high watermark (GLRPB_GHW). In this case XOFF is only concerned with all the no-drop TCs of all ports.

e. The global packet counter has passed above its packet high watermark (GLRPB_PHW). In this case XOFF is only concerned with all the no-drop TCs of all ports.

Next time such a condition occurs, no XOFF is issued until the congested condition disappeared (such as XON sent) or until the XOFF timer expired.

38.24.1.2.5 Accounting Packet Reception

When a packet is stored into the receive packet buffer (the packet has not been dropped), the global filling counter is increased by the number of bytes required to store the packet. Also increment the FIRST counter for which the condition is satisfied when checking it in the following order:

1. Shared pool filling counter — Unless one of the following conditions is met:
   a. No shared pool for the port (null shared pool size, PRTRPB_SPS=0b).

b. Shared pool filling counter has reached its high watermark (PRTRPB_SHW) and there is a dedicated pool for the TC (PRTRPB_DPS>0).

c. TC occupancy counter (relative to the shared pool only) has reached its high threshold (PRTRPB_SHT) and there is a dedicated pool for the TC (PRTRPB_DPS>0).

2. Dedicated pool filling counter — Unless there is no dedicated pool for the TC (PRTRPB_DPS>0).

38.24.1.2.6 Accounting the Servicing of a Receive Packet

When a packet is fetched from the receive packet buffer (packet sent to the host), the global filling counter is decremented by the number of bytes used to store the packet in the receive packet buffer. Also decrement the counter(s) for which the condition is satisfied when checking them in the following order until the amount of served bytes has been reached:

1. Dedicated pool filling counter — If both conditions are fulfilled:

   a. There is a (non-null) dedicated buffer attached to the TC (PRTRPB_DPS>0).

   b. The dedicated pool filling counter has still not reached zero (non-empty). Note that the shared pool is no longer full. However, there is no hurry to send XON for a TC that has still not been served internally by Rx-ETS since it gets into XOFF state. This approach might encounter some tolerable fairness issues across the TCs.

2. Shared pool filling counter — When no or empty dedicated pool and as long as shared pool filling has not reached zero.

Note: When a port is disabled or disconnected, its Rx packet buffer is drained in such a way that all its associated filling counters get down to zero within a short and bounded time.

Note: When a packet is served, it may that 2 x 48 bytes = 96 bytes are not decremented from the dedicated pool filling counter and from the shared pool occupancy counter. This happens if the packet is not aligned with 64-byte memory blocks and if packets in the TC are not served as per their insertion order.

38.24.1.2.7 Policy for Issuing XON

When servicing a packet attached to a no-drop TC (packet sent to the host), a PFC XON notification is issued onto the wire and internally to the loopback path if ALL the following conditions are satisfied:

1. XOFF was issued for this UP.
2. TC occupancy counter is below or equal to its low threshold.
3. The global packet counter is below or equal its packet low watermark (GLRPB_PLW) AND the relevant condition is met:

   a. If there is a dedicated pool for the TC and it has a non-null low watermark (PRTRPB_DLW>0): when the dedicated pool filling is below or equal its low watermark.

   b. If there is a dedicated pool for the TC (PRTRPB_DPS>0) and it has a null low watermark (PRTRPB_DLW=0b) when the shared pool filling is below or equal its low watermark.

   c. If there is no dedicated pool for the TC when the shared pool filling is below or equal its low watermark.

38.24.1.2.8 Prevention of PFC crosstalk Between PCIe Functions

Though Rx queues are allocated per TC in RSS mode (in the 10 GbE controller), no TC index is stored in the Rx queue context. Intel recommends to avoid mixing drop and no-drop traffic over the same Rx queue, as the no-drop traffic might cause head of line blocking of the drop traffic.

There is a need to prevent a malfunctioning function, a paused function, or a non-trusted VM from blocking an entire PFC-enabled TC if it does not provide Rx descriptors. For this purpose, a per UP timer is started each time a packet located at the head of the per UP linked list is waiting for software to free Rx descriptors in the Rx queue to which the packet is destined. A timer starts counting on if the UP is associated with a TC of type No Drop (if the TC is of type Drop, head-of-line packets are dropped when descriptors are not available).

The timer is reset when Rx descriptors are freed.

The ports/UPs for which the timer timed out can be read via GLDCB_RUPTI bitmap register. Writing 1b to a bit in the bitmap restarts the corresponding timer. The Rx queue that blocked the UP is reported to the PRTDCB_RUPTQ array of registers. The queue index reported in this register is the absolute queue index in the device space, which is different than the queue index used for the Tx and Rx queue registers.

When a timer expires, the EMP is interrupted and takes the following steps:

1. EMP identifies the port and UP of the offending queue by reading the GLDCB_RUPTI register.
2. EMP reads the PRTDCB_RUPTQ register to identify the offending queue.
3. EMP clears the timer by writing a 1b to the corresponding bit in GLDCB_RUPTI.
4. EMP reads the QTX_CTL[Q] register, where Q is the index of the offending RX queue. QTX_CTL identifies the following:

Note: This flow is based on the assumption that the software programs the QTX_CTL register for any matched transmit queues of all enabled receive queues. If this rule is not addressed by the software, a reported LAN Queue Overflow Event admin command is sent to PF0.

   a. Whether the queue belongs to a PF or VF (the PFVF_Q field).
      - A VM queue is considered to be a PF queue for this section.
   b. The PF that owns the queue (or the parent PF in case the queue belongs to a VF) (the PF_INDX field).
   c. The VF that owns the queue (for the case that the queue belongs to a VF) (the VFVM_INDX field).

5. If the queue belongs to a PF:
   a. If the NVM bit *PF reset on queue overflow* is set.
      - The EMP issues a PFR to the function. Note that the PFR also resets the PF's VFs.
   b. If the NVM bit *PF reset on queue overflow* is cleared.
      - The EMP sends a LAN Queue Overflow AQ event to the PF, notifying the event.
6. The EMP indicates to the MC that the PF driver is not present, if the queue belongs to a VF.
   a. The EMP sends a LAN Queue Overflow AQ event to the respective PF that owns the VF, notifying the event.

b. The PF issues a VFR to the function.

Note: It is advisable to assign manageability traffic to UPs associated with Drop TCs, therefore avoiding the risk of stalling due to host queue that overflows.

38.24.1.2.9 Rx Packet Buffer Configuration Flow

All RPB registers are dynamic registers. They can be loaded from the NVM at initialization/reset time, and/or they can be modified at run time further to a change in DCBX resolution. Re-configuring RPB while working might lead to the issuing of XOFF/XON notifications to several TCs at once.

When a change has been made to one of the input parameters described in Section 38.24.1.2.10, the RPB settings have to be updated.

The modified Rx-PB setting must be loaded to the device according to the following order:

1. Low thresholds and low watermarks that require to be decreased — registers PRTRPB_SLT, PRTRPB_SLW, PRTRPB_DLW.
2. High thresholds and high watermarks that require to be decreased — registers PRTRPB_SHT, PRTRPB_SHW, PRTRPB_DHW.
3. Dedicated pools size that require to be decreased.
4. Shared pool size — register PRTRPB_SPS.
5. Dedicated pools size that require to be increased.
6. High thresholds and high watermarks that require to be increased — registers PRTRPB_SHT, PRTRPB_SHW, PRTRPB_DHW.
7. Low thresholds and low watermarks that require to be increased — registers PRTRPB_SLT, PRTRPB_SLW, PRTRPB_DLW.

38.24.1.2.10 Configuration Rules for RPB Partitioning

The computing flow described in this section is run by the entity responsible to run DCBX (see Section 38.24.3). The flow is run for each port independently over the amount of memory allocated to the port, as part of the Rx-PB configuration flow described in Section 38.24.1.2.9.

It makes use of the following parameters as input:

- Number of enabled ports (static parameter after initialization time), used to determine the amount of memory allocated to a port. It is set to 968 KB divided by the number of enabled ports.
- Tx LPI enabled/disabled over the port (as per PRTPM_EEER.TX_LPI_EN register bit). When Tx LPI is enabled over the port, the amount of memory usable for PFC by the port is reduced by [Tx LPI Exit Time - Max MFS over all the TCs of the port]. This reduced effective packet buffer size of the port will be referred as RPB_ESize(Port). For simplicity, worst case Tx LPI Exit Time value is taken regardless of the port type or speed, which corresponds to Tw_phy of 10GBASE-KR Case-2: 14.25 us @ 10G, equivalent to 17.4 KB.
- Number of supported TCs in receive (PRTDCB_GENC.NUMTC).
- Number of PFC-enabled TCs (TC2PFC field in PRTDCB_TC2PFC and for 40 GbE links PRTMAC_HSEC_CTL_RX_PAUSE_ENABLE and PRTMAC_HSEC_CTL_TX_PAUSE_ENABLE).

- MFS per TC table (see Section 38.24.3.3.3, it is maintained by the DCBX handling entity, and can be also loaded from the NVM). MFS over a TC is referred as MFS(TC), while maximum MFS over all TCs of a port is referred as MFS(max).
- PCIe round trip time (see PCIRTT).
- PFCLinkDelayAllowance — Delay Value (DV) over the port. Since MFS is a per-TC value, there is a different DV value per TC, referred as DV(TC). See PFCLinkDelayAllowance.

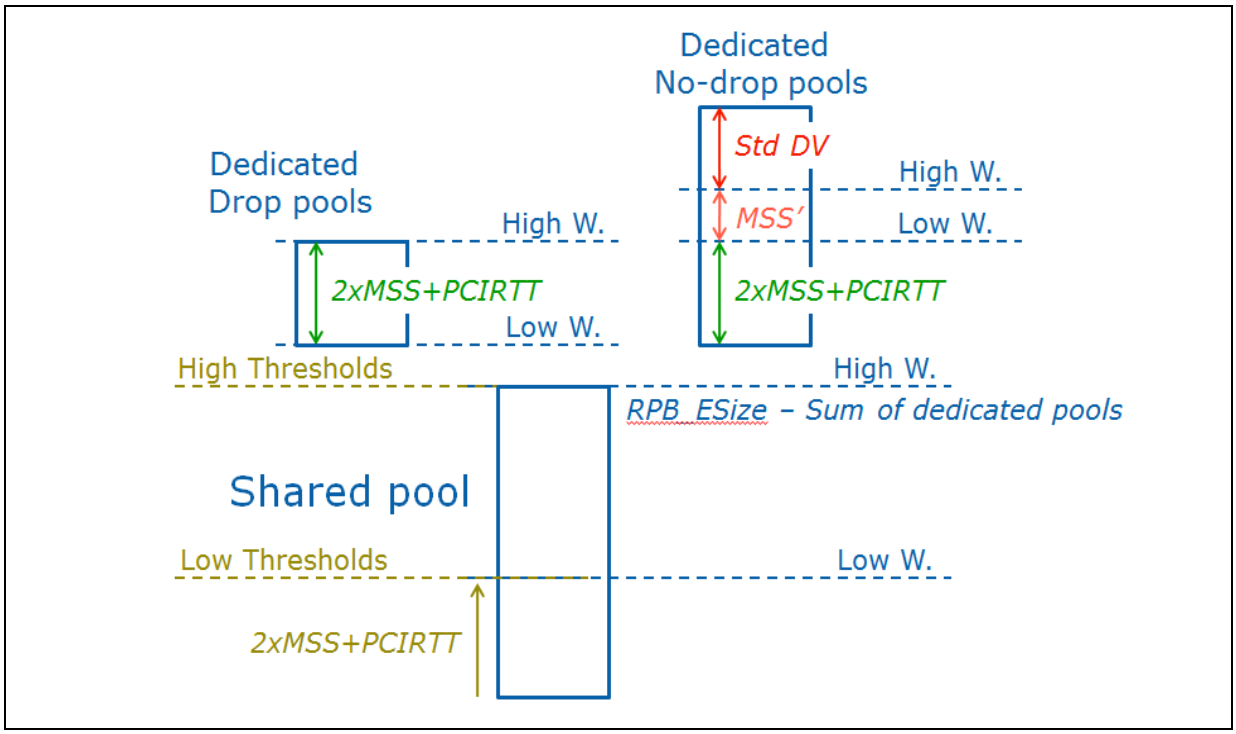
The flow makes use of DV(TC) extended by one MFS(TC) because of a last minute packet from the same TC that can be looped back into the RPB. It is noted as **Std DV(TC) = DV(TC) + MFS(TC)**.

1. **Allocate fully independent buffers to all TCs** and allocate the remaining buffer space (if any) to the shared pool of the port.
   a. Step 1 dedicated no-drop TC:
      - Dedicated no-drop low watermark = 2 x MFS(TC) + PCIRTT
      - Dedicated no-drop high watermark = Dedicated no-drop low watermark + MAX{MFS(max), 4.5KB}; the last term is referred as MFS'
      - Dedicated no-drop pool size = Dedicated no-drop high watermark + Std DV(TC)

   b. Step 1 dedicated drop TC:
      - Dedicated drop low watermark = 0
      - Dedicated drop high watermark = Dedicated drop pool size
      - Dedicated drop pool size = 2 x MFS(TC) + PCIRTT

   c. Step 1 shared:
      - Shared pool size = RPB_ESize(Port) – Sum of Dedicated pools' sizes
      - Shared high watermark = Shared pool size
      - Shared low watermark = 0
      - Shared low threshold = 0
      - Shared high threshold = Shared pool size
      - Exit condition: If shared pool size is not negative, then exit the flow

Figure 38-63.Normal Configuration at Step 1



2. **Downgrade TCs to semi independent TCs** — Downgrade a TC to a semi independent PB. Try a drop TC first, but if it is not enough to fulfill the following exit condition, then downgrade a no-drop TC instead, and so forth starting from lowest TCID and up.

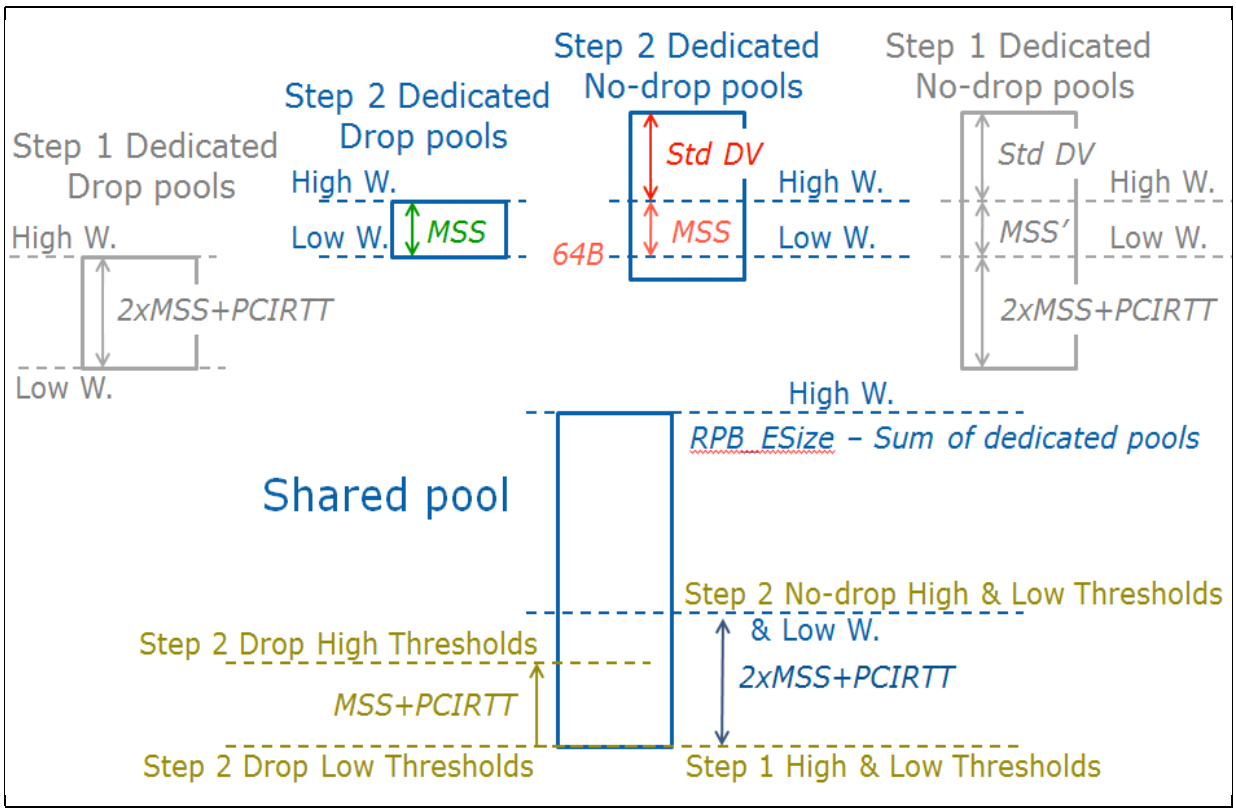
   a. Step 2 no-drop TC:
      - Dedicated no-drop low watermark = 64 bytes
      - Dedicated no-drop high watermark = MFS(TC) + 64 bytes
      - Dedicated no-drop pool size = Dedicated no-drop high watermark + Std DV(TC)
      - Shared no-drop low threshold = 2 x MFS(TC) + PCIRTT
      - Shared no-drop high threshold = 2 x MFS(TC) + PCIRTT

   b. Step 2 drop TC:
      - Dedicated drop low watermark = 0
      - Dedicated drop high watermark = MFS(TC)
      - Dedicated drop pool size = Dedicated drop high watermark
      - Shared drop low threshold = 0
      - Shared drop high threshold = MFS(TC) + PCIRTT

   c. Step 2 shared:
      - Shared pool size = RPB_ESize(Port) - Sum of dedicated pools' size of TCs left unchanged from Step 1 - Sum of Dedicated pools' size of TCs downgraded at Step 2.
      - Shared low watermark = 2 x MFS(max) + PCIRTT
      - Shared high watermark = Shared pool size
      - For TCs left unchanged from Step 1: Shared low threshold = Shared high threshold =0

- Exit condition: If Shared pool size > Sum over n1 of MFS(TC) + Sum over n2 of (MFS(TC)/2) + PCIRTT + MFS(max), where n1 = # of Step 2 no-drop TCs, n2 = # of Step 2 drop TCs; then exit the flow with no new iteration of Step 2

Figure 38-64.Normal Configuration at Step 2



3. **Downgrade TCs to shared pool only** — Downgrade a TC to shared pool only, meaning it does not have a dedicated pool. Try a drop TC first, but if it is not enough to fulfill the following exit condition, then downgrade a no-drop TC instead, and so forth starting from lowest TCID and up.

   a. Step 3 no-drop TC:
      - No dedicated pool
      - Shared no-drop low threshold = 2 x MFS(TC) + PCIRTT
      - Shared no-drop high threshold = Shared no-drop low threshold + MFS(TC)

   b. Step 3 drop TC:
      - No Dedicated pool
      - Dedicated drop low watermark = 0
      - Dedicated drop high watermark = 64 bytes
      - Dedicated drop pool size = Dedicated drop high watermark
      - Shared drop low threshold = 0
      - Shared drop high threshold = 2 x MFS(TC) + PCIRTT

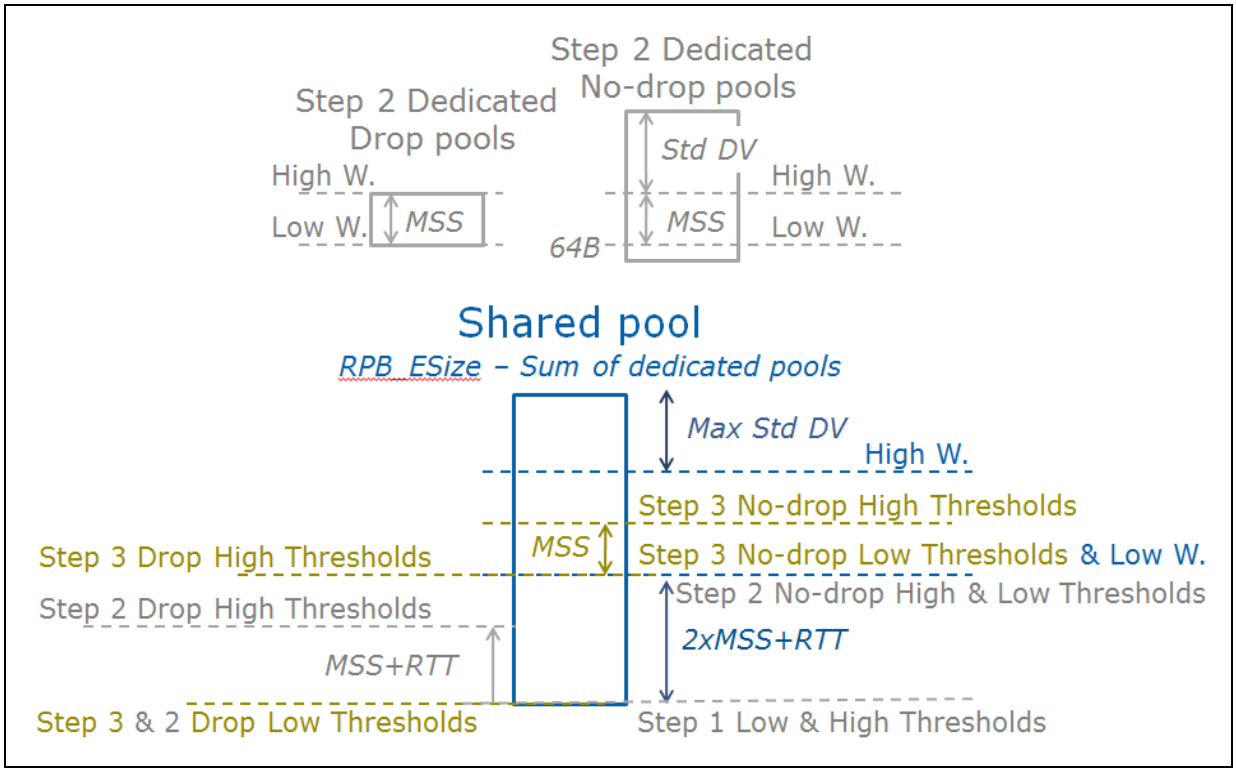
c. Step 3 shared:
   - Shared pool size = RPB_ESize(Port) - Sum of dedicated pools' size of TCs left unchanged from Step 2
   - Shared low watermark = 2 x MFS(max) + PCIRTT
   - Shared high watermark = Shared pool size – Max Std DV across no-drop TCs downgraded at step 3 (if any)

   Exit conditions:

   If Shared pool size > Sum over all no-drop TCs of MFS(TC) + Sum over all drop TCs of (MFS(TC)/2) + PCIRTT + MFS(max), AND

   If Shared pool size > Max {Std DV + 3 x MFS(TC) + PCIRTT} across no-drop TCs downgraded at step 3 (if any), then exit the flow with no new iteration of Step 3.

Figure 38-65.Normal Configuration at Step 3



38.24.1.2.11 Configuration Rules for Global Watermarks

See Section 38.24.1.2.10 for the definition of Std DV(TC).

- The global high watermark (GLRPB_GHW) = [968 KB — Sum over the 4 ports of (Max Std DV(TC) over all no-drop TCs of a port).
- The global low watermark (GLRPB_GLW) = [968 KB — 2 x (Sum over the 4 ports of (Max Std DV(TC) over all no-drop TCs of a port).
- The packet high watermark (GLRPB_PHW) = (968 KB / 128 B) — [(Sum over the 4 ports of (Max Std DV(TC) over all no-drop TCs of a port) / 64 B].
- The packet low watermark (GLRPB_PLW) = (968 KB / 128 B) — 2 x [(Sum over the 4 ports of (Max Std DV(TC) over all drop TCs of a port) / 64 B].

38.24.1.2.12 Examples of RPB Configuration Flows

- 10GBASE-T PHY is assumed, which includes the following interface delay contributors:
  - — XGMII MAC/RS interface: 8 192 + 2 * 2 048 = 12 288 bit times
  - — 10GBASE-T Delay: 25 600 bit times
- All MFS must be rounded up to 16 bytes
- All pool sizes or watermarks must be rounded up to 64 bytes

Table 38-289.Example #1: 1 No-drop TC + 3 Drop TCs

10 GbE Analysis	Bit Time	KB	µs
MSS(max)	77824	9.5	7.8
PFC frame	672	0.1	0.1
Cable delay	5556	0.7	0.6
Interface delay	37888	4.6	3.8
Higher layer delay	6144	0.8	0.6
Std DV(other no-drop TCs)	327296	40.0	32.7
PCIRTT (1 µs=10,000 bit times)	20000	2.4	1
Dedicated pool size (other no-drop TCs) at Step 1	581120	70.9	58.1
Dedicated pool size (drop TCs) at Step 1	176128	21.5	17.6
Number of other no-drop TCs at Step 1	1		
Number of drop TCs at Step 1	3		
Shared pool size at Step 1	466432	57	

Table 38-290.Example #2: 2 No-drop TCs + 3 Drop TCs

10 GbE Analysis	Bit Time	KB	µs
MSS(max)	77824	9.5	7.8
PFC frame	672	0.1	0.1
Cable delay	5556	0.7	0.6
Interface delay	37888	4.6	3.8
Higher layer delay	6144	0.8	0.6
Std DV(other no-drop TCs)	327296	40.0	32.7
PCIRTT (1 µs=10,000 bit times)	20000	2.4	1
Dedicated pool size (other no-drop TCs) at Step 1	581120	70.9	58.1
Dedicated pool size (drop TCs) at Step 1	176128	21.5	17.6
Number of other no-drop TCs at Step 1	2		
Number of drop TCs at Step 1	3		
Shared pool size at Step 1	-457216	-56	
Dedicated pool size of another no-drop TC downgraded at Step 2	405504	49.5	
Dedicated pool size of a drop TC downgraded at Step 2	77824	9.5	
Number of other no-drop TCs downgraded at Step 2	2		
Number of drop TCs downgraded at Step 2	3		
Minimum Shared pool size required at Step 2	406528	49.6	
Shared pool size at Step 2	421683	51	

Table 38-291.Example #3: 8 No-drop TCs

10 GbE Analysis	Bit Time	KB	µs
MSS(max)	77824	9.5	7.8
PFC frame	672	0.1	0.1
Cable delay	5556	0.7	0.6
Interface delay	37888	4.6	3.8
Higher layer delay	6144	0.8	0.6
Std DV(other no-drop TCs)	327296	40.0	32.7
PCIRTT (1 µs=10,000 bit times)	20000	2.4	1
Dedicated pool size (other no-drop TCs) at Step 1	581120	70.9	58.1
Dedicated pool size (drop TCs) at Step 1	176128	21.5	17.6
Number of other no-drop TCs at Step 1	8		
Number of drop TCs at Step 1	0		
Shared pool size at Step 1	-2731520	-333	
Dedicated pool size of another no-drop TC downgraded at Step 2	405504	49.5	
Dedicated pool size of a drop TC downgraded at Step 2	77824	9.5	
Number of other no-drop TCs downgraded at Step 2	8		
Number of drop TCs downgraded at Step 2	0		
Minimum Shared pool size required at Step 2	720896	88.0	
Shared pool size at Step 2	-1326285	-162	
Number of other no-drop TCs downgraded at Step 3	6		
Number of drop TCs downgraded at Step 3	0		
Minimum Shared pool size required at Step 3	720896	88.0	
Shared pool size at Step 3	1106739	135	110.7

Table 38-292.Example #4: 4 No-drop TCs + 2 Drop TCs (Sheet 1 of 2)

10 GbE Analysis	Bit Time	KB	µs
MSS(max)	77824	9.5	7.8
PFC frame	672	0.1	0.1
Cable delay	5556	0.7	0.6
Interface delay	37888	4.6	3.8
Higher layer delay	6144	0.8	0.6
Std DV(other no-drop TCs)	327296	40.0	32.7
PCIRTT (1 µs=10,000 bit times)	20000	2.4	1
Dedicated pool size (other no-drop TCs) at Step 1	581120	70.9	58.1
Dedicated pool size (drop TCs) at Step 1	176128	21.5	17.6
Number of other no-drop TCs at Step 1	4		
Number of drop TCs at Step 1	2		
Shared pool size at Step 1	-1443328	-176	
Dedicated pool size of another no-drop TC downgraded at Step 2	405504	49.5	
Dedicated pool size of a drop TC downgraded at Step 2	77824	9.5	
Number of other no-drop TCs downgraded at Step 2	4		

Table 38-292.Example #4: 4 No-drop TCs + 2 Drop TCs (Sheet 2 of 2)

10 GbE Analysis	Bit Time	KB	µs
Number of drop TCs downgraded at Step 2	2		
Minimum Shared pool size required at Step 2	523264	63.9	
Shared pool size at Step 2	-311501	-38	
Number of other no-drop TCs downgraded at Step 3	1		
Number of drop TCs downgraded at Step 3	1		
Minimum Shared pool size required at Step 3	581120	70.9	
Shared pool size at Step 3	623411	76	62.3

38.24.2 Transmit Path DCB

38.24.2.1 Transmit Path ETS

Transmit path ETS is basically handled in the Tx-scheduler.

This section describes the requirements on the Tx data path to avoid distortions on the ETS scheme performed by Tx-scheduler.

The approach relies on two principles:

1. Maintain the order of requests issued by the Tx-scheduler, as most as possible.
2. Avoid misleading the scheduler decisions by inexact reports.

Other ETS requirements on Tx data path:

1. Serve low latency traffic as fast as possible in any condition, as long as it is not overtaking its allocated bandwidth.
2. Guarantee that best effort TCs can be served at full blown, regardless to the XOFF/XON events history over loss less TCs.
3. Achieve the per port throughput performance goals.
4. Avoid starvation of one ETS TC by another, even in the case it is dripping packets while the other TCs offer sustained workload.

38.24.2.1.1 Identifying LL Traffic in Transmit

In order to handle sudden PFC XOFF notifications received from the link, the transmit data path is provided with buffers in its different stages. To reduce impact on die size, provision is made for only two types of traffic, LL and Bulk (B). This approach requires a priority differentiation between LL and B traffic in order to decide in which buffer a Tx data or request for data has to be stored. TC indexes are used for that purpose according to a setting made in LLTC bitmap in PRTDCB_TETSC_TCB and PRTDCB_TETSC_TPB registers.

The entity that runs DCBX configures all non-ETS TCs as LL traffic, and only those TCs.

38.24.2.1.2 UP-to-TCs in Transmit

Mapping of UP-to-TC in transmit

Register PRTDCB_TUP2TC controls the mapping of UPs-to-TCs in the transmit path with respect to PFC XOFF/XON notifications received from the link partner or internally from the loopback path. It defines which transmit TC is paused/released when a UP bit is set in a PFC XOFF/XON frame received (or in internal notification).

Note: PRT_TCTUPR, PRTDCB_TUP2TC, and PRTDCB_RUP2TC registers must be programmed identically.

Tx manageability traffic issued by the MC is bound to the lowest indexed drop TC, or to TC0 if all TCs are no-drop.

Remapping of the UP field in transmit

The remapping scheme also applies to the traffic issued by EMP (or MC), which is represented in the tables by its own four VSI numbers.

38.24.2.2 Transmit Path PFC

This section describes the guarantee of no packet loss at the link partner and inside the 10 GbE controller further to receiving PFC pause frames from the link partner or further to PFC notifications received from the internal switch.

TCs are classified in two types relatively to PFC: no-drop (loss less) TCs, and drop TCs (best effort delivery). PRTDCB_TC2PFC and PRTDCB_MFLCN.RPFCE bitmaps (and PRTMAC_HSEC_CTL_TX_PAUSE_ENABLE for 40 GbE links) control the use of PFC by a TC over a link.

38.24.2.2.1 Performance Goals for Transmit Path PFC

Basic assumptions

1. The 10 GbE controller must support optimally up to 2 TCs per port which are PFC-enabled. It might support more, but under sub-optimal performing concerning throughput.
2. One PFC-enable TC supports 9.5 KB jumbo packets.

XOFF performance goal

Once an internal/external XOFF notification is received on a TC:

1. Transmission from the TC is stopped on the wires within the DV specified in Annex O of the PFC specification. See Section 38.24.1.2.10 for more detail.
2. Other TCs (PFC-enabled or not) can be served at full blown. Some limitations can be tolerated for other PFC-enabled TCs if 2 PFC-enabled TCs are already paused for the port, depending on the history of PFC XOFF events. Worst case is when the accumulated amount of traffic (commands or data) that was in the data path at the moment XOFF events were received on a port has reached twice the Tx command/data path depth. In this case, all PFC-enabled TCs of the port are paused.
3. PFC XOFF events received on one port must not impact performance of other PFC-enabled TCs on other ports.

XON performance goal

1. **Over the wires** — Once an external XON notification is received for a TC or once the XOFF timer has ended for it, resuming transmission over the wires for this TC must be made possible within the *Higher Layer Delay* plus the *Interface Delay* specified in Annex O of the PFC specification. This corresponds to half of the DV used for XOFF performance goal above minus the cable delay.
2. **On the internal switch** — Once an internal XON notification is received for a TC, resuming transmission at the internal switch ingress port for this TC must be made possible within only the *Higher Layer Delay* specified in Annex O of the PFC specification. Internal switch ingress port is located at the entry of the Rx packet buffer.

Note: Interface delay must take in account the layers present and active inside the 10 GbE controller and those present in an external PHY connected to it on the board (if any).

38.24.2.2.2 Transmit PFC

The TC(s) concerned by an XOFF/XON notification, which is per UP, are identified by the setting made in PRTDCB_TUP2TC register.

38.24.2.2.3 PFC Dead-lock Prevention

Tx traffic that belongs to PFC-enabled TCs can be halted in the Tx pipe due to endless XOFF received either from the line or from the internal loopback path. Flushing out this traffic is a gating condition for the completion of the following operations: Tx queue disable, PFR, VFR, disabling the PFC of a TC or DCBX UP to TC remapping.

The device implements a mechanism that automatically flushes out Tx traffic that belongs to a port for which the link goes down. However, there is no similar mechanism for flushing out traffic halted in the Tx data pipe due to endless XOFF conditions. In the case of PFR, an endless XOFF condition is detected autonomously by the device and handled as described in the text that follows. For the other cases (previously listed), it is the PF(s) responsibility to detect that the Tx data-path is halted for a long time by periodically monitoring the eight Tx PFC timers attached to a port, one per TC (PRTDCB_TPFCTS.PFCTIMER). Each timer (per TC) is restarted by the device every time the TC is halted by an XOFF notification (received from the link). If any of these counters crosses a threshold defined by the ENDLESS_XOFF_THRESH parameter (in offset 0x15 of EMP setting module in the NVM), the PF(s) software should take the following actions:

- Post a PFC Ignore admin command (see Section 38.24.5.1) for the TC, requesting EMP to set IGNORE_FC bit(s) (32 bits - 1 per TC and per port) in the GLDCB_TFPFCI register. It causes the entire Tx data path to ignore PFC indications for the concerned TC/port, whether they were received from the line or from the internal loopback path. When in this state, the device flushes out the concerned frames without issuing them over the line or into the internal loopback path.
- When the endless XOFF condition disappeared, and/or when the Tx pipe is checked to have been cleaned up (read PRTDCB_TCWSTC and PRTDCB_TCMSTC arrays of registers), the PF clears the *IGNORE_FC* bit by posting a PFC Ignore admin command with *Ignore Flag* cleared.

Note: In case the port is operated in LFC, the functionality is controlled by the bit corresponding to TC0.

The eight Tx PFC timers of a port must be always operative, even if a TC has no more UP mapped to it (further to a UP to TC mapping change driven via DCBX).

38.24.2.3 Tx Path DCB Configuration Rules

Tx-scheduler registers are dynamic registers. They can be loaded from the NVM at initialization/reset time, and/or they can be modified at run time further to a change in DCBX resolution.

PRTDCB_TUP2TC and PRTDCB_GENC are also dynamic registers. Refer to the flow described in Section 38.24.3.3.1 for the way they are modified.

All other Tx DCB settings described in Section 38.24.2.1 and in Section 38.24.2.2 are static. They must not be modified at run time. They are loaded from the NVM only at initialization/reset time.

38.24.3 DCBX

DCBX protocol relies on the exchange with the peer of untagged LLDP packets over the physical link.

38.24.3.1 DCBX / LLDP Ownership

By default DCBX is handled by EMP, though when the device is operated in SFP mode, the host can decide to take the DCBX ownership once system boot has been completed. SFP mode software initiates the transition on a per port basis, by posting the Stop LLDP Agent command.

When software owns DCBX, it might give DCBX ownership back to EMP using the Start LLDP Agent AQ command. Each time the host which handles DCBX goes to sleep mode, DCBX is un-covered (EMP does not cover for DCBX). It is the host's responsibility to reset DCB configuration to its default settings (such as single TC, PFC disabled, etc.) prior to entering the sleep state. When software handles DCBX, it is also responsible to configure the DCB settings of the port, via setting the DCB registers. It must use the relevant flow described in Section 38.24.4. In such case, the end-station might be made by software as the DCBX master. For example, the entity that propagates its DCB settings to the peer.

When LLDP/DCB is handled by EMP, the 10 GbE controller behaves always as a DCBX slave, retrieving its DCB settings from recommendations or configurations received from the peer.

The LLDP agent embedded in EMP (including its DCBX agent) is reset by GLOBR. Further to such reset events, EMP restarts LLDP Agent and DCBX resolution for the ports on which LLDP was handled by EMP.

When LLDP is handled by EMP, following to a CORER event, EMP must reconfigure the Tx/Rx paths (and any relevant hardware that was reset) with the DCB configuration that prevailed before the reset occurred.

38.24.3.2 DCBX Version

There are two known versions of DCBX standard. IEEE and CEE. The 10 GbE controller supports both and can adopt at runtime its peer's mode of operation or to determine the mode of operation based on NVM configuration word: DCBX Mode.

The NVM configuration selects one of the three options; CEE, IEEE or adopt peer's mode of operation.

When it configured to adopt peer's mode of operation, it is done based on DCBX TLV's received from the link peer using the following detailed flow.

- Upon initialization (linkup, GLOBR or LLDP ownership hand-off from software to EMP, TLV counters expire) EMP runs IEEE/DCBX as default mode.
- Upon receiving a DCBX TLV's from the partner:
  - EMP determines the actual DCBX mode it runs. This is done based on the OUI field in the DCBX TLV. in IEEE the OUI TLV value is 00 80 C2. in CEE, this value is 00 1B 21.
  - If the message contains IEEE/DCBX TLV's, then
    - DCBX mode == IEEE
    - Ignore and CEE/DCBX TLV in the message
  - End if

  - Else If the message contains only CEE/DCBX TLV's, then
    - DCBX mode == CEE
  - End if
- When DCBX mode is changed between CEE and IEEE, the entire DCB configuration is re-initialized to the default DCB setting.

Method used by software to find which version EMP runs

Software sends down the GET_CEE_DCBX_OPER_CFG AQ command and waits for its completion.

```
If retval == success

    DCB Mode = CEE.

    Negotiated DCB arbitration available in response buffer for this command.

End if

If retval == ENOENT

    DCBX Mode = IEEE.

    SW sends down "GET_LLDP_MIB" AQ command to find out the DCB arbitration
negotiated using DCBX protocol.

End if

If retval == EPERM

    Software has taken control of DCBX. Query SW DCB agent for any DCBX
configuration

End if
```

38.24.3.2.1 CEE Vs. IEEE DCBX

This section highlights the main difference between CEE and IEEE modes that affects the DCBX resolution. Refer to the two standard specifications for more details.

Operational configuration

IEEE DCBX TLV's always advertise the operational setting of the feature. CEE DCBX on the other hand, computes the operational configuration based on local DesiredCfg and peer's DesiredCfg. If software wants to determine the OperCfg, it cannot use the GET_LLDP_MIB AQ command for CEE as OperCfg is not the wire in case of CEE. Software has to GET_CEE_DCBX_OPER_CFG AQ command for obtaining the OperCfg in CEE.

Priority groups

CEE DCBX has a concept of priority group that is not present in IEEE DCBX. EMP is required to equate the priority group and TC concepts. That is, treat the priority-to-priority group mapping received in the CEE DCBX TLV as the priority-to-TC mapping as well.

PFC

For CEE DCBX, if the PFC does not become operational via CEE DCBX state machine, then CEE DCBX disables PFC and enables link level flow control. When EMP runs the DCBX agent in IEEE mode this recommendation is irrelevant since it behave in slave mode and align itself to the peer.

38.24.3.3 DCBX Offload Flow

DCBX TLVs embedded in the LLDP packets are identified by a TLV type value of 127 followed by the IEEE 802.1 OUI field value of 0x0080C2 for IEE and 0x001B21 for CEE.

There are four types of IEEE/DCBX TLVs that are handled by the 10 GbE controller, and they are classified into three categories as follow:

- ETS TLV's — ETS configuration TLV and ETS recommendation TLV.
- Priority-based FC configuration TLV.
- Application priority configuration TLV.

CEE/DCBX defines the following sub-TLV's:

- Control TLV — Contains fields controlling the operation of CEE DCBX.
- Priority Groups TLV — Similar to IEEE ETS feature.
- Priority Flow Control TLV — Similar to IEEE PFC feature.
- Application Protocol TLV — Similar to IEEE application priority feature.

CEE/DCBX transmits all DCB TLV's packed inside a single LLDP TLV. IEEE/DCBX uses a separate LLDP TLV for each of the DCB features.

LLDP packets are periodically issued with DCBX TLV's inside. The following steps represent the specific handling required from the EMP-based DCBX agent, after receiving DCBX TLVs from the peer:

1. **Wait until the first DCBX TLV is received or for a change in the received DCBX TLV**.
2. **EMP identifies LLDP/DCBX peer** — The LLDP peer identifier is made by the concatenation of its chassis ID and port ID fields. Compare these fields to a saved copy of the previous LLDP peer identifier received (if there is any and if it is not aged out).
   a. If the LLDP peer identifier is not the same, then store the new LLDP peer identifier into the Remote DCBX parameters saved in firmware data RAM, and start a timer with the longest 802.1AB TTL value of any of the peers. Go to next step even before waiting for timer's end.
   b. If the number of LLDP peers that run DCBX is greater than one at the timer ends, then a multiple peer condition is detected and reported via the *PRTDCB_GENS.DCBX_STATUS* field set to MULTIPLE_PEERS (as well as via a LLDP MIB change event if configured for). Restart the timer with the longest 802.1AB TTL value of any of the peers and exit the flow with updating the LLDP DCBX peer and treating DCBX parameters as NULL (no DCBX peer).
3. **If for some reason, the DesireCFG received from the peer does not match the DesiredCfg issued to it, the CEE/DCBX agent re-sends the DCB TLV's for the concerned feature(s) with the DesiredCfg copied from the peer and with the LocalParmaterCahnge bit set.**
   a. Refer to CEE DCBX specification for the way CEE/DCBX agent must handle the CEE control and feature state machines.

4. **EMP resolves DCBX and reconfigures DCB** — Update the RPB settings according to the flow described in Section 38.24.1.2.9 and to the rules described in Section 38.24.1.2.10 for the normal RPB settings. Update the Local DCB management objects of Section 38.24.3.4 according to the flows described in Section 38.24.3.3.1 and Section 38.24.3.3.3.

   a. If LLDP TLVs were aged out, EMP reconfigures the device with the default DCB settings, excepted for LFC which should be reverted to the previously known LFC mode that prevailed before DCBX was resolved.

   b. If some DCBX TLVs are missing or malformed, the concerned DCB settings are returned to their default configuration.

38.24.3.3.1 ETS TLVs Resolution

IEEE/DCBX ETS TLV's

Figure 38-66.ETS Recommendation FLV Format

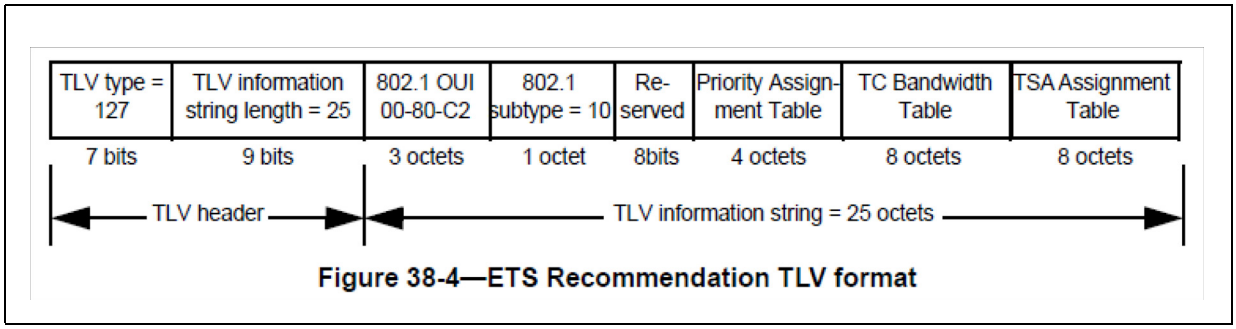


Figure 38-4—ETS Recommendation TLV format

Figure 38-67.ETS Configuration FLV Format

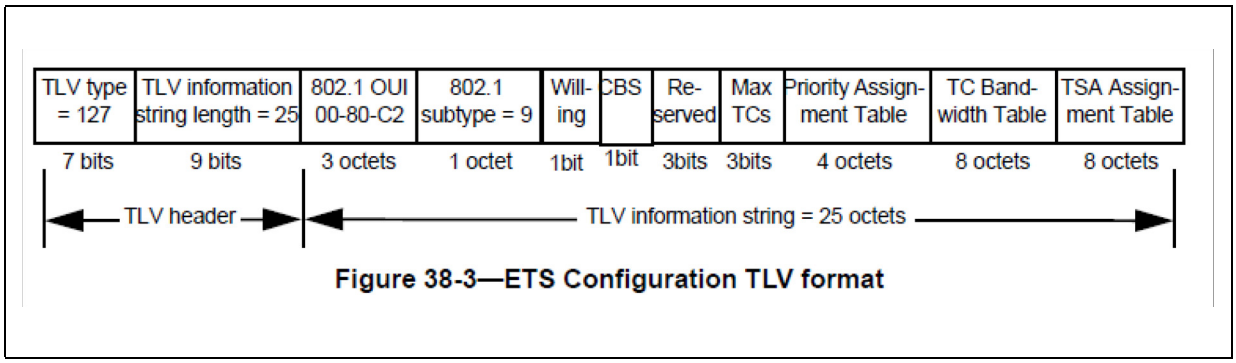


Figure 38-3—ETS Configuration TLV format

The ETS TLV sent by a node concerns its ETS setting in transmit direction. ETS parameters resolution uses the asymmetric attribute passing state machine described in Section 38.4.1 of DCBX standard. It allows a different setting for each Tx direction between two peers. When DCBX is handled by EMP, the 10 GbE controller acts as a DCBX slave and therefore the local *Willing* bit issued to the peer is always set.

If no ETS TLV is received from the peer, or if they are aged out, the default ETS setting is recovered, which assumes all the UPs are mapped to a single non-ETS TC, TC 0.

Otherwise, two scenarios whether or not an ETS recommendation TLV has been received from the peer:

1. If the ETS recommendation TLV is present in peer's TLVs, local Tx ETS settings are extracted from the ETS recommendation TLV received. It ise referred to as the

remote ETS TLV. Local Rx ETS settings are extracted from ETS configuration TLV of the peer.

2. If no ETS recommendation TLV or if it is aged out, local ETS settings (both Tx and Rx) are extracted from the ETS configuration TLV of the peer. It is referred as the remote ETS TLV.

CEE/DCBX ETS TLV's

Refer to corresponding section in the CEE ETS specification.

ETS Configuration Change Flow

EMP extracts the number of TCs from the ETS TLV. For IEEE/DCBX, it corresponds to the number of different TC indexes identified in the priority assignment table. It then performs the following tasks sequentially, according to the case:

1. If a there is a change in a PFC policy or in UP-to-TC mapping, go to the port draining flow.
   a. Reconfigure the DCB settings according to the following order: Load the number of TCs into PRTDCB_GENC.NUMTC.
   b. Configure the Rx commands FIFOs as described in Section 38.24.1.1.4.
   c. Update the PFC settings to TCs according to the new priority assignment table (if it was modified), using the flow described in Section 38.24.3.3.2.
   d. Load the priority assignment table extracted from the remote ETS TLV into PRTDCB_RUP2TC.
   e. Load the priority assignment table extracted from the remote ETS TLV into PRT_TCTUPR.
   f. Mark non-ETS TCs as LL TCs in the *LLTC* field of PRTDCB_TETSC_TCB and PRTDCB_TETSC_TPB registers.
   g. Assign MC pass-through traffic to the lowest indexed drop TC, or to TC0 if all TCs are no-drop.
2. Go to the Tx-scheduler configuration flow where TC bandwidth table and TSA assignment table extracted from the remote ETS TLV is loaded there into Tx-Scheduler.

Note: When changing the UP-to-TC mapping and/or PFC policy, it might be that PFC and ETS behaviors be perturbed during the transition time until traffic in the pipes are emptied.

The 10 GbE controller always advertises support to eight TC's, but in CEE/DCBX it locally operates with a number of TC's that equal to the number of priority groups received from the peer. It is a one-to-one device's TC-to-priority group assignment made locally by the EMP. So, from TC index 0 and up, from PG index 0 and up: TC0 to PG0, TC1 to PG1 etc. Excepted PGID 15,which is a special value dedicated to the strict priority group, such as the priority group with unlimited bandwidth allocation. PGID 15 must be mapped to the lowest free TC index once all other priority groups have been mapped

38.24.3.3.2 PFC Configuration TLV Resolution

IEEE/DCBX PFC TLV

Figure 38-68.Priority-based Flow Control TLV Format

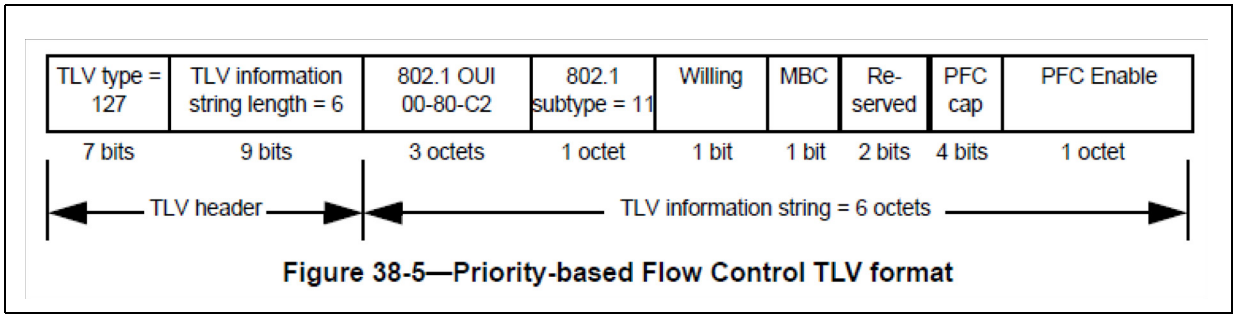


Figure 38-5—Priority-based Flow Control TLV format

PFC parameters resolution uses the symmetric attribute passing state machine described in Section 38.4.2 of DCBX standard. It means that the same setting is expected in both directions Tx/Rx, up to the edge of the DCB network.

Two scenarios follows, assuming the local *Willing* bit is always set:

1. Remote *Willing* bit is cleared. The local PFC configuration is copied from the peer.
2. Remote *Willing* bit is set as well. The local PFC configuration is taken from the link partner with the lower numerical MAC address.

With scenario 1 and scenario 2, if the peer has the lower numerical MAC address, the PFC configuration change flow described in PFC Configuration Change Flow is performed by the EMP.

Note: Since PFC is disabled by default on all UPs, Intel recommends that the remote *Willing* bit not be set, so that the device for sure inherits the PFC setting from the peer.

CEE/DCBX PFC TLV

Refer to the corresponding section in the CEE specification.

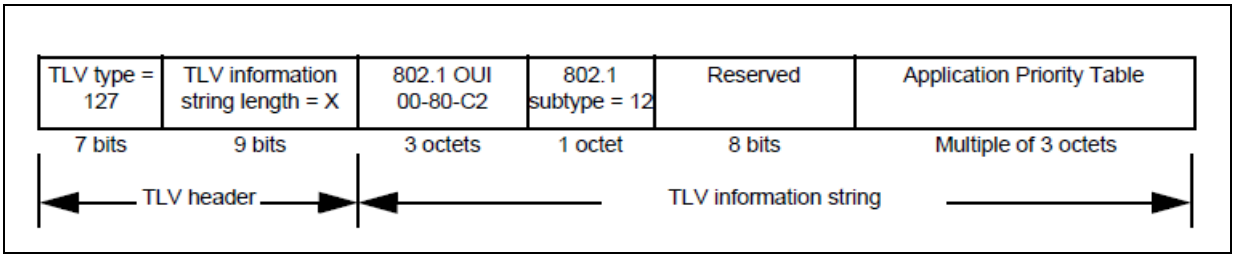
PFC Configuration Change Flow

- Load the PFC *Enable* bit vector of the peer into *TC2PFC* field of PRTDCB_TC2PFC and into *RPFCE* field in PRTDCB_MFLCN (and into PRTMAC_HSEC_CTL_RX_PAUSE_ENABLE and PRTMAC_HSEC_CTL_TX_PAUSE_ENABLE for 40 GbE links), making use of the PRTDCB_RUP2TC.UP2TC settings as follows:
  a. If one of the UPs attached to a TC has its bit set in the PFC *Enable* bit vector, then set to 1b the TC2PFC bit that corresponds to the TC.
  b. If all the UPs attached to a TC have their bit cleared in the PFC *Enable* bit vector, then clear to 0b the TC2PFC bit that corresponds to the TC.
- Set the *PRTDCB_RUP.NOVLANUP* field with the lowest indexed no-drop UP. If there is not a no-drop UP, use the lowest indexed drop UP instead.

38.24.3.3.3 Application Priority TLV

IEEE/DCBX Application Priority TLV

Figure 38-69.Application Priority Table TLV Format

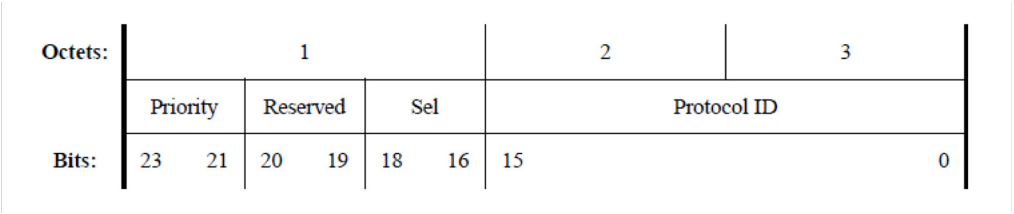


Content of this TLV is mainly informational. Two sets are stored by firmware in the DCBX database: one local set and one remote set received from the peer.

It is assumed that the peer always sends the application TLV and that the local set is copied from the remote set.

The local/remote tables stored by firmware are accessible to software via the Get LLDP MIB command (see PFCLinkDelayAllowance).

Figure 38-70.Application Priority Table



The UP attached to iSCSI traffic is the priority field in the application priority table row for which:

- Sel = 2 or 4
- Protocol ID = iSCSI TCP port (3260)

CEE/DCBX application protocol TLV

Refer to the corresponding section in the CEE specification.

38.24.3.4 DCB Managed Objects

38.24.3.4.1 DCBX Managed Objects

When a DCBX port is handled by the EMP, the DCBX managed objects are the DCBX parameters stored internally in EMP data RAM and/or in device registers. All DCBX objects are per-LAN port. Two instances are stored per port, one issued by the device to the link partner referred as Local DCBX parameters, and one received from the link partner referred as remote DCBX parameters.

When DCBX is handled by the EMP, the PFs must be restrained from any write access to the DCB registers listed in Table 38-293. Note that RW in the table refers to the EMP capability to write the object.

Table 38-293.Local DCBX Managed Objects

IEEE Object Name *CEE Object Name	Data Type	Width In Bits	Admin	Default	Related Local Configuration Register
ETS Configuration TLV					
Willing	Boolean	1	RO	1	Firmware only
Credit-based Shaper (CBS)	Boolean	1	RO	0	Firmware only
Max TCs *Num TCs Supported	Unsigned integer	3	RW	0[1]	PRTDCB_GENC.NUMTC
Priority Assignment Table *Priority Allocation Table	Unsigned integer [0..7]	8x4	RW	0	PRTDCB_TUP2TC, PRT_TCTUPR, PRTDCB_RUP2TC
TC Bandwidth Table *Priority Group Allocation Table	Unsigned integer [0..7]	8x7	RW	0	
TSA Assignment Table	Unsigned integer [0..7]	8x8	RW	0[2]	
PFC Configuration TLV					
Willing	Boolean	1	RO	1	Firmware only
MBC	Boolean	1	RO	0	Hardcoded
PFC Cap *Num TC PFC Supported	Unsigned integer	4	RO	0x8[3]	Firmware only
PFC Enable *PFC Config Table	Boolean [0..7]	8x1	RW	0	TC2PFC in PRTDCB_TC2PFC RPFCE in PRTDCB_MFLCN PRTMAC_HSEC_CTL_RX_PAUSE_ENABLE PRTMAC_HSEC_CTL_TX_PAUSE_ENABLE PRTMAC_HSEC_CTL_RX_ENABLE_GPP PRTMAC_HSEC_CTL_RX_ENABLE_PPP
Application Priority Configuration TLV					
Application Priority Table *App Protocol Config Table	Unsigned integer	32x24	RW	0	Firmware only
MFS per TC Table	Unsigned integer	8x15[4]	RW	1536	Firmware only

Notes:

1. 0 is the encoding for 8.
2. Strict priority algorithm is assumed by default (indicated by a zero value) on each user priority. Setting the value of 2 selects ETS bandwidth allocation scheme.
3. The 10 GbE controller is always able to support up to eight PFC-enabled TCs though in some cases (when jumbo frames are enabled) fully independent PFC behavior is partially achieved.
4. Max frame size is defined in bytes.

38.24.3.4.2 PFC Managed Objects

The PFC managed objects are PFC parameters that do not concern DCBX. Note that RW in Table 38-294 refers to the software capability to write the object.

Table 38-294.PFC Managed Objects

Object Name	Data Type	Width In Bits	Admin	Default	Specification Reference	Related Local Configuration Register
PFC Control Objects					Table 12.1	
PFCLinkDelayAllowance[1]	Unsigned integer	16[2]	RW	0	12.18	PRTDCB_GENC.PFCLDA
PCIRTT[a]	unsigned integer	16	RW	0	N/A	GLDCB_GENC.PCIRTT
PFCRequests	unsigned integer	16	RO	0	12.18	GLPRT_PXOFFTXCNT
PFCIndications	unsigned integer	16	RO	0	12.18	GLPRT_PXOFFRXCNT

Notes:

1. These parameters are relevant only when DCBX is handled by EMP. Any modification made by the PF driver to these register fields take effect on the RPB settings only once a DCB Updated admin command is posted.
2. In PFC specification the parameter is expressed in link bits, but in the 10 GbE controller it is expressed in 16-byte units. SNMP agent handled in the operating system is responsible to convert the parameter in bits units when returning it in the corresponding MIB object.

PFCLinkDelayAllowance

The value of PFCLinkDelayAllowance is configurable by software per port via the PRTDCB_GENC.PFCLDA register field. It is expressed in 16-byte time units. It might be more accurate if the EMP selects by itself the PFCLinkDelayAllowance value that corresponds to the PHY, without referring to this register field.

Firmware uses this parameter to compute the Rx packet buffer settings. See Section 38.24.1.2.10 for more detail.

Referring to Annex O in PFC Std, PFCLinkDelayAllowance is also referred as the DV, which is computed per TC as follows:

DV= 2*(Max Frame) + (PFC Frame) + 2*(Cable Delay) + 2*(Interface Delay) + (Higher Layer Delay)

- *2*(Max Frame) = MFS (TC) + MFS (max)* is in fact formed by the sum of the MFS (TC) over the TC for which DV is computed and the maximum MFS (max) over all TCs. See Section 38.24.3.3.3 for the MFS per TC table to be used. The PFCLinkDelayAllowance loaded to the PFCDLA register field must assume MFS is 9.5 KB for both. Firmware is responsible for handling a different PFCLinkDelayAllowance per each TC according to the MFS per TC table it handles; referred to as DV(TC).
- *(PFC Frame)* duration is equal to 672 bit times.
- *2*(Cable Delay)* is proportional to the cable length and inversely proportional to the bit time duration (like link speed). A 100 m Cat6 cable operating at 10 GbE link speed is the worst case for all the supported cable length, medium type, and link speeds across 10 GbE, 1 GbE, and 100 Mb/s. Under this worst case conditions the term equals to 2*(5,556) bit times. Firmware is not responsible for optimizing the term contribution if the link speed is below 10 GbE. However, it is firmware's responsibility to multiply this value internally by 4 for 40 GbE link.

- *2* (Interface Delay) + (Higher Layer Delay)* must take in account the interface delay at each side of the link, and higher layer delay at the peer side. The PFCLinkDelayAllowance value loaded to the PFCLDA register field must be computed assuming at the peer the worst case values tolerated by the standard over the medium type. Table O-1 in Annex O of the PFC standard gives the interface delay contributors for the different layers that might be present between the 10 GbE controller and the physical medium.

Figure 38-71.IEEE 802.3 Interface Delays

Sublayer	Maximum RTT (bit times)	Maximum RTT (pause quanta)	Reference (subclause of 802.3)
10G MAC Control, MAC, and RS	8 192	16	46.1.4
XGXS and XAUI	2 048	4	48.5
10GBASE-X PCS	2 048	4	49.2.15
10GBASE-R PCS	3 584	7	50.3.7
LX4 PMD	512	1	53.2
CX4 PMD	512	1	54.3
Serial PMA and PMD	512	1	52.2
10GBASE-T	25 600	50	55.11

The *higher layer delay* at 10 GbE is 614.4 ns, which is equivalent to 6,144 bit times. As before, firmware is not responsible for optimizing the contribution of this term if the link speed is below 10 GbE.

PCIRTT

The value of PCIRTT is configurable by software for the entire device via the GLDCB_GENC.PCIRTT register field. It is expressed in 16-byte time units.

Firmware uses this parameter to compute the Rx packet buffer settings. See Section 38.24.1.2.10 for more detail.

It represents the maximum PCIe round trip time supported with no performance penalty (2 µs by default. A 10 GbE link speed is the worst case for all the supported link speeds across 10 GbE, 1 GbE, and 100 Mb/s. At 10 GbE speed the term is equal to 20,000 bit times. Firmware is not responsible for optimizing the term contribution if the link speed is below 10 GbE.

38.24.4 Initialization of DCB Functionality

38.24.4.1 DCB Initialization Flow

DCB initialization flow is always handled by the EMP, starting from pre-boot time, further to any of the following events: GLOBR, PCIR, PERST, EMPR, and POR.

1. **Auto-load** — The device auto loads the default configuration of the ports from the NVM, which might include some changes to the DCB/RPB registers'. See Table 38-295 for the default DCB/RPB settings, since the registers hardware default are tuned to four ports enabled at 10 GbE (or lower) link speed. In any case, DCB/RPB defaults loaded to registers at this stage must guarantee basic connectivity for the enabled ports.

It must equally partition RPB across the enabled ports by setting an identical value into their shared pool buffer (PRTRPB_SPS register) so that their sum equals 968 KB. For disabled ports, zero is expected to be loaded from the NVM into PRTRPB_SPS. DCB defaults must assume one single (non-ETS) TC for the port (like TC0) to which all UPs are mapped. No dedicated pools are used at this stage. PFC is disabled for all UPs. The control port of the internal switch must (by default) be addressed to EMP. It guarantees untagged LLDP packets are forwarded to the EMP in Rx, and in Tx, untagged LLDP packets (wrongly) issued by the host are filtered out by the device before reaching the wire.

2. **Auto-negotiation** — If auto-negotiation is enabled by default, the device performs auto-negotiation with the peer, which might result in a link speed change and/or in a LFC status change. Any DCB/RPB configuration change that requires further to auto-negotiation is handled by the device at pre-boot stage, as follows:

   a. *Further to a link speed change* — Update receive port arbiter. The device updates the credits allocated to the port in the Rx port arbiter. Other link speed registers settings are handled by the EMP. Firmware is responsible to multiply internally the value of PFCLinkDelayAllowance by 4 for 40 GbE link.

   b. *Further to a LFC state change* — Update RPB and LFC registers. The EMP computes and loads the high/low watermark of the shared pool allocated to the port as if it was a single fully independent buffer. See Section 38.24.1.2.10. Other LFC registers settings are handled by the EMP.

3. **EMP runs DCBX** — Once EMP has re-loaded, the LLDP filters in the internal switch, and if the LLDP agent was not disabled via NVM settings or via the Stop LLDP Agent admin command (the later case is not relevant for POR triggering events), the EMP performs a DCBX exchange with the link partner from scratch, and updates the device registers related to DCB managed objects as per the flows described in Section 38.24.3.2.

   a. This step starts as soon as the link is up (auto-negotiation has completed) and any time the link goes up again later on.

4. **PFs read the DCB configuration** — PFs issue a Get LLDP MIB command to the device. From the response posted by the EMP, it identifies the UP used for storage as well as the UP to TC mapping used at the link level.

5. **PFs configure VEBs and VSIs according to the flow.**

   a. If there are several UPs attached to a TC, and if there is at least one Rx queue for each, the EMP might attach one corresponding Tx queue set per each UP under the TC. This is done upon availability of the Tx queue set resources. In this case, a Tx queue set belongs to a single UP and to a single TC. Otherwise, it belongs to a single TC. In any case, an Rx queue might not carry traffic from a single UP or a single TC, as Rx filters assign traffic to Rx queues according to many decision parameters that might not take in account only UP/TC.

   b. The EMP sets the per VSI UP translation tables used in Tx and Rx, VSI_TUPR and VSI_RUPR tables, respectively.

Note: On CORER events, the EMP must reconfigure the core blocks with the DCB settings saved from the last DCBX resolution.

38.24.4.2 Transfer of DCB Ownership Between EMP and Operating System

38.24.4.2.1 Transfer of DCB Ownership to Operating System

In SFP mode, it might be that the operating system decides to take ownership over DCBX from the EMP. This is done per port according to the following flow:

1. **Operating system disables DCBX offload** — The operating system posts a Stop LLDP Agent command. This notifies the EMP to stop handling DCBX.
   a. If the Shutdown LLDP Agent command variant is used, the EMP LLDP agent issues a last LLDP packet to the peer with TTL=0 before returning the port to its default DCB setting; one single non-ETS TC (like TC 0). This variant is used by the PF when it does not plan to take LLDP ownership. It has the advantage to gracefully shutdown the LLDP with the peer without requiring the host to get involved in the last LLDP packet transmission.
2. **EMP completes the Stop LLDP Agent command**.
3. **Operating system redirects LLDP to the host** — Operating system moves the LLDP forwarding rules of the internal switch to the control VSI.
4. **Operating system runs DCBX** - Operating system performs a DCBX exchange with the link partner, and updates the device registers related to DCB managed objects as per the flows described in Section 38.24.3.2. It might be that the operating system re-runs DCBX in master mode, propagating to the peer the local DCB settings, which were made to its DCBX agent via some DCB application.
5. **When the operating system detects a DCBX change**, it posts a DCB updated event in the admin command queue to notify the EMP that the untagged traffic mapping must be modified.

Note: If the Stop LLDP Agent command is issued when the LLDP agent is already off, the command is silently dropped.

The LLDP agent might be disabled by default via clearing the Factory LLDP Admin Status word in the NVM LLDP configuration module. This mode can be useful on nodes that provide NAT and other network edge services.

On any reset event other than POR, the EMP does not retake LLDP ownership on ports where ownership has been moved to the PF. It means that the PF must not perform the transfer of DCB ownership to the operating system again further to such reset events.

38.24.4.2.2 Return of DCB Ownership to EMP

At operating system reboot, it is possible that the system goes through a complete initialization cycle, including a BIOS pre-boot phase. In such a case, it is required that LLDP ownership returns to the EMP so that it supports pre-boot activities such as operating system boot over the network. Pre-boot software has to first initiate a GLOBR to re-initiate the device hardware, and then to signal the EMP to take over the LLDP agent. The later is done by posting a Start LLDP Agent command.

Further to receiving the Start LLDP agent, the EMP starts the LLDP agent from scratch, making use of the hardware defaults of registers.

38.24.4.3 Initial DCB Settings

Table 38-295 lists the changes in DCB registers that must be made via NVM settings. Hardware default settings for DCB registers correspond to ports are operated at 10 GbE or lower speeds.

Table 38-295.Changes in DCB Registers Made via NVM Settings When Port Operated at Different Link Speed

Register	Field	1x10 GbE	2x10 GbE	4x10 GbE
PRTDCB_GENC	PFCLDA	0x079D [1]	0x079D [1]	0x079D [1]
GLDCB_GENC	PCIRTT	0x009C	0x009C	0x009C
PRTDCB_RPPMC	RX_FIFO_SIZE	0x10	0x10	0x08
PRT_SWR_PM_THR	THRESHOLD	0x0F	0x0F	0x09
GLRPB_PHW	PHW	0x1246	0x1246	0x1246
GLRPB_PLW	PLW	0x0846	0x0846	0x0846
PRTRPB_SPS	SPS	0x3C800	0x3C800	0x3C800
GLDCB_TGENC_RLPM	TCPM_DIS	0x0	0x0	0x0
	CWLB_MODE	0x0	0x0	0x0
PRTDCB_TCPFCPC_RLPM	PORTOFFTH	0xC64	0x65C	0x34A
PRTDCB_ TCPFCTCC_RLPM	TCOFFTH	0x624	0x320	0x19E
PRTDCB_TCPMC_RLPM	CPM	0x624	0x320	0x19E
	TCPM_MODE	0x0	0x0	0x0
PRTDCB_TCPFCTCC_RLPM	LL_PRI_EN	0x0	0x0	0x0
	LL_PRI_TRESH	0x608	0x304	0x182

Note:
1. *The value put in the NVM image must be optimized to reflect the PHY type and the internal delays in the 10 GbE controller.

38.24.5 DCB Admin Commands

All parameters in the admin commands are defined in little endian.

38.24.5.1 PFC Ignore

This command is used to request the device to ignore a PFC condition present on a Tx path for a TC. The same command is used to release PFC ignore request. When PFC packets are ignored they are considered as regular packets and might be forwarded to the host.

Table 38-296.PFC Ignore Response (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0301	Command opcode.
Datalen	4-5		Must be zeroed.
Return Value	6-7		Return value. 0x0 - No error (success).
Cookie High	8-11	Cookie	Opaque value, is copied by the EMP into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by the EMP into the completion of this command.

Table 38-296.PFC Ignore Response (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Completion_flags	16		Bits 7:0 = TC status bitmap. Returns the TCs for which PFC is currently ignored. Bit n set to 1b means a PFC condition on Tx path for TC n is ignored by the device.
Reserved	17-19		Reserved, must be zeroed.
Reserved	20-23		
Reserved	24-27		
Reserved	28-31		

Table 38-297.PFC Ignore Command

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0301	Command opcode.
Datalen	4-5		Must be zeroed by driver.
Return Value/ VFID	6-7		Must be zeroed by driver.
Cookie High	8-11	Cookie	Opaque value, is copied by the EMP into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by the EMP into the completion of this command.
Command_flags	16-17		Byte 16, bits 7:0 = TC bitmap. A bitmap of the TCs concerned by the command. Bit n set to 1b means TC n is concerned by the request. When LFC is used instead of PFC, TC index 0 is used to request ignoring LFC. Byte 17, bits 6:0 = Reserved, must be zeroed. Byte 17, bit 7 = Ignore flag. When set to 1b, the PF requests to ignore PFC conditions on the TC indexes set to 1b in the TC bitmap. When cleared to 0b, the PF requests to release any ignore PFC condition request issued for the TC indexes set to 1b in the TC bitmap.
Reserved	18-19		Reserved, must be zeroed.
Reserved	20-23		
Reserved	24-27		
Reserved	28-31		

38.24.5.2 LLDP/DCBX Admin Commands

Refer to the LLDP Protocol commands section for more details.

38.24.5.3 DCB Updated Command

When LLDP is handled by the PF, this command is used to notify the EMP that a DCB setting has been modified.

When LLDP is handled by the EMP, it is used by PF to notify the EMP that one of the following parameters has been modified:

- PFCLinkDelayAllowance set by PRTDCB_GENC.PFCLDA
- PCIRTT set by PRTDCB_GENC.PCIRTT

In return, the EMP might modify the mapping of untagged traffic (via PRTDCB_RUP). A command completion is posted once the mapping has been changed.

Table 38-298.DCB Updated Response

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0302	Command opcode.
Datalen	4-5		Must be zeroed.
Return Value	6-7		Return value. 0x0 = if needed, RPB or untagged traffic mapping was modified accordingly.
Cookie High	8-11	Cookie	Opaque value, is copied by the EMP into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by the EMP into the completion of this command.
Reserved	16		Reserved.
Reserved	17-19		Reserved, must be zeroed.
Reserved	20-23		
Reserved	24-27		
Reserved	28-31		

Table 38-299.DCB Updated Command

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x0302	Command opcode.
Datalen	4-5		Must be zeroed by driver.
Return Value	6-7		Must be zeroed by driver.
Cookie High	8-11	Cookie	Opaque value, is copied by the EMP into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by the EMP into the completion of this command.
Reserved	16-19		Reserved, must be zeroed.
Reserved	20-23		
Reserved	24-27		
Reserved	28-31		

38.24.5.4 LAN Queue Overflow Event

See Section 38.24.1.2.8 for more details. This event is sent from the device to a PF. It does not generate a response.

Table 38-300.LAN Queue Overflow Event (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	0-1	0	
Opcode	2-3	0x1001	Command opcode.
Datalen	4-5	0x00	N/A
Return Value	6-7	0x00	N/A
Cookie High	8-11	Cookie	Opaque value, is copied by the EMP into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by the EMP into the completion of this command.
PRTDCB_RUPTQ	16-19	See Section 38.24.1.2.8	Contains a copy of the PRTDCB_RUPTQ register reporting the absolute index (in the device space) of the reported receive queue.
QTX_CTL	20-23	See Section 38.24.1.2.8	Contains a copy of the QTX_CTL register of the matched transmit queue pair of the reported receive queue.

Table 38-300.LAN Queue Overflow Event (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Reserved	24-31		Reserved, must be zeroed.

38.25 Transmit Scheduling

38.25.1 Bandwidth Management Hierarchy

38.25.1.1 Hierarchy of Switching Elements

The 10 GbE controller contains a number of switching elements that might be arranged in a hierarchical manner. Depending on how the 10 GbE controller is configured, the hierarchy might consist of zero, one or two layers. Each switching element contains a single uplink (egress) port and one or more virtual (ingress) ports. The egress port of the switching element can be connected to physical port or to the virtual ingress port of another switching element. For a detailed description of the switching components, terminology and hierarchy rules supported by the 10 GbE controller see Section 38.21.7.2.1.

Figure 38-72.Hierarchy of Switching Elements

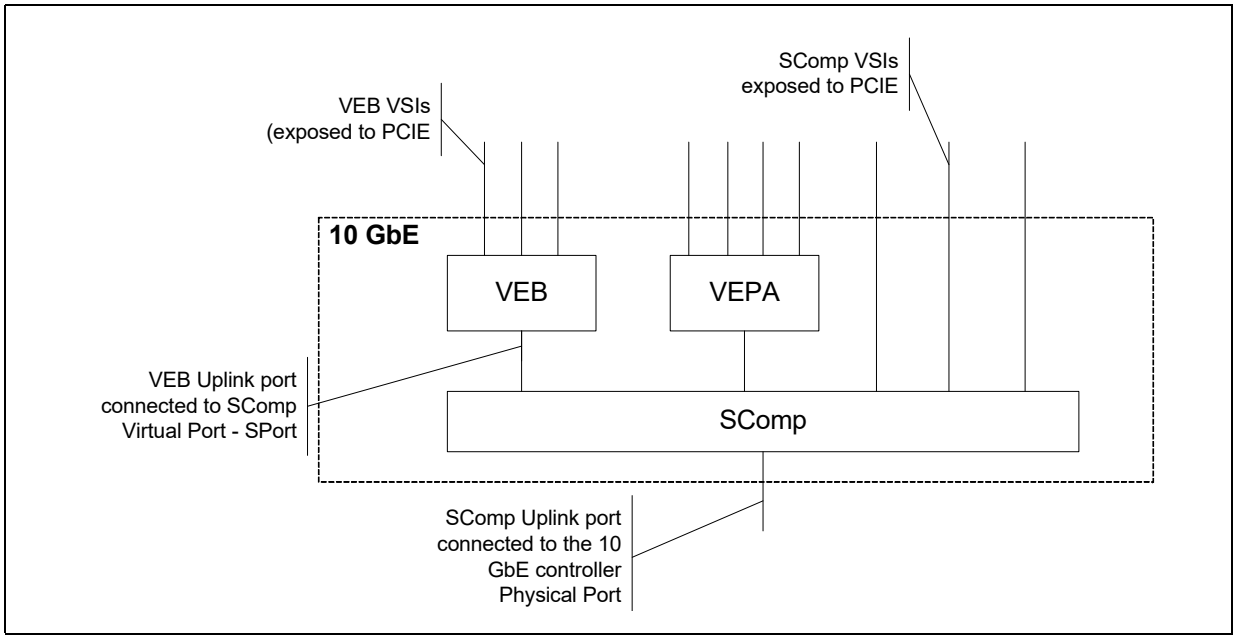


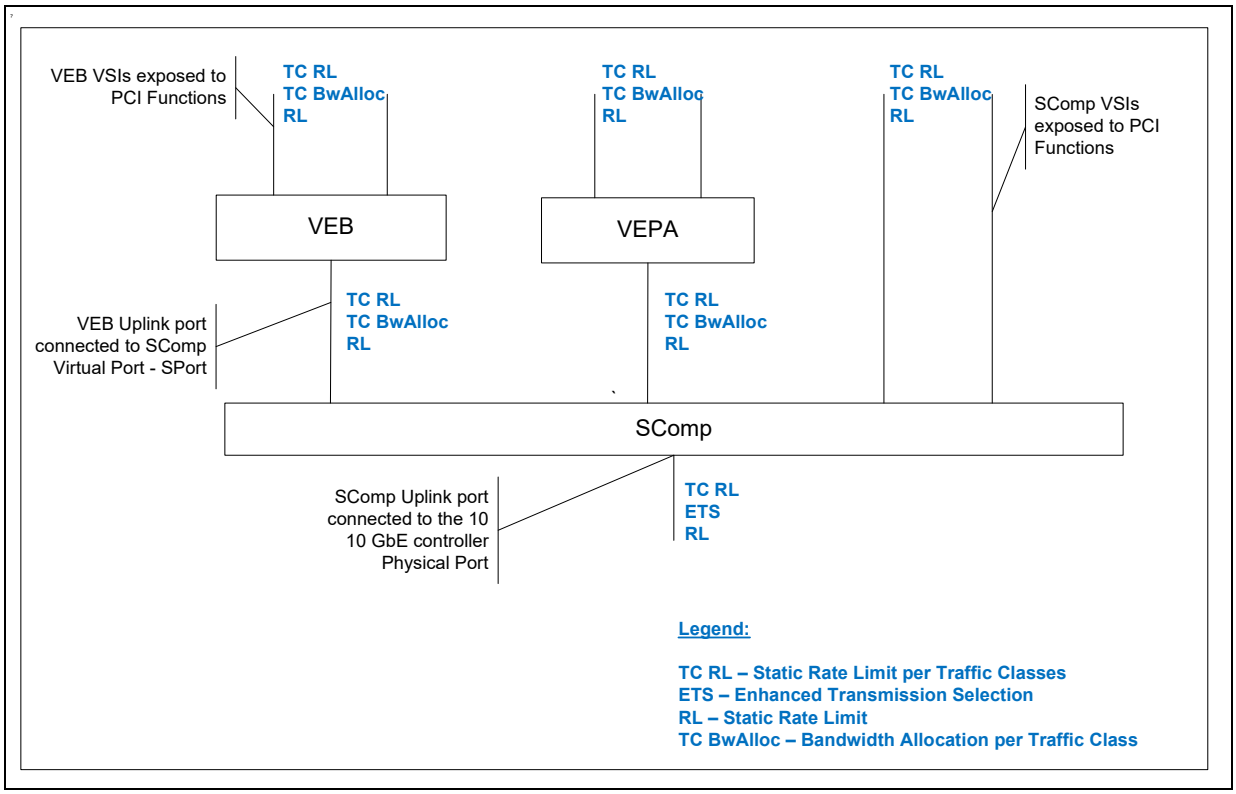
Figure 38-72 shows an example of a two level switching hierarchy. The S-comp's ingress port is connected to the physical port. Several of the S-comp's virtual ports (S-channels) are exposed as PCIe functions. Two of the S-comp's egress ports are connected to VEB and VEPA switching elements. Both the VEB and VEPA switching elements expose multiple virtual ports as PCIe functions or VSIs.

A virtual port that is not connected to the egress of a switching element is exposed as a virtual port to the PCIe interface and called a VSI. In this chapter, the term virtual port is used to refer to all ports of a switching component, and the term VSI is used for virtual ports that are exposed as PCIe functions. Each of the VSIs is assigned to either

the PF or VF and connected to the virtual egress port of the respective PCIe function. A VSI can be owned by a single PCIe function at any one time, but might transition from one PCIe function to another. A single PCIe function can own multiple VSIs.

By default, a VF owns a single VSI. A VM that needs connectivity to multiple virtual ports or switching elements can either use an emulated path through a VMM (using VSIs owned by the physical function) or use multiple VFs, one for each VSI.

Figure 38-73.10 GbE Ethernet Controller Bandwidth Management Attributes



The 10 GbE controller transmit scheduler configuration follows topology of the internal switching components. Figure 38-73 shows various bandwidth management attributes and their association with the switching components. A subset of bandwidth management controls shown on this diagram is available depending on the transmit scheduler configuration scheme. See Section 38.25.2 for description of supported transmit scheduler configuration schemes.

- VSI/Switching Component Bandwidth Limit — Each VSI and switching component can be configured to limit maximum bandwidth used by that VSI and switching component. Limiting bandwidth of the switching component effectively limits bandwidth all VSIs associated with that switching component. For details, see Section 38.25.1.3. Bandwidth limit of VSIs and switching components can be configured in all supported transmit scheduler configuration schemes. See Section 38.25.2.
- ETS Configuration of Physical Port — The 10 GbE controller physical port can be configured with ETS configuration.This enables designers to specify distribution of bandwidth among TCs enabled for the physical port. Each physical port can be configured with independent ETS. TCs configured to the physical port can be enabled for the switching components and VSIs allocated for the port. Allocation, association and bandwidth distribution within each one of the TCs must be

consistent with the physical port ETS configuration and bandwidth distribution within the switching hierarchy. For example, TCs enabled for VSI must be enabled for respective switching components and physical ports, and a total bandwidth allocated to all VSIs for particular TC must be equal to the bandwidth allocated for that TC on the switching component. For details on ETS configuration, see Section 38.25.1.4.

- Per Traffic Type Bandwidth Allocation — Each VSI and switching component can have relative bandwidth allocated per TC. Such bandwidth allocation is hierarchical, similar to the VSI/switching component bandwidth allocation previously described. But it is done within a particular TC. For example, each VSI of the switching component can be configured with bandwidth share with respect to other VSIs of the same switching component for the particular TC. Such bandwidth allocation is mutually exclusive with the VSI bandwidth allocation previously described and depends on the transmit scheduler configuration scheme. See Section 38.25.2 for more detail. The bandwidth allocation principles are described in Section 38.25.1.2.
- Per Traffic Type Bandwidth Limit — Each VSI that has an ETS enabled can be configured to limit the maximum bandwidth available for one or more TCs configured for that port. For details, see Section 38.25.1.4.

Configuring the 10 GbE controller scheduler is done by firmware. Software can use the Admin Queue interface to modify a default allocation of bandwidth attributes. See Section 38.25.3.1 for more detail.

Table 38-301 lists bandwidth management attributes supported for each switching component and VSI depending on the transmit scheduler configuration scheme. For a description of both transmit scheduler configuration schemes, see Section 38.25.2.

Table 38-301.10 GbE Ethernet Controller bandwidth Management Attributes

Configuration Scheme	Switching Element Name	Bandwidth Attribute	Description
ETS-Based Scheme	SComp	Bandwidth Limit	This applies to any switching component or VSI directly attached to the physical port. Bandwidth limit for entire physical port regardless type of TC.
		ETS	Relative bandwidth allocation between TCs within switching component.
		Traffic Type Bandwidth Limit	Bandwidth limit for individual TC.
	VEB/PA	Bandwidth Limit	Bandwidth limit for entire switching component regardless type of TC.
		Traffic Type Bandwidth Allocation	Relative bandwidth allocation of switching component within each TC type.
		Traffic Type Bandwidth Limit	Bandwidth limit for switching component per TC type.
	VSI	Bandwidth Limit	Bandwidth limit for the entire VSI regardless of TC type.
		Traffic Type Bandwidth Allocation	Relative bandwidth allocation of VSI within each UP type.
		Traffic Type Bandwidth Limit	Bandwidth limit for VSI per traffic type (TC or UP).

38.25.1.2 Bandwidth Allocation

The 10 GbE controller supports configurable allocation of bandwidth to the switching elements and their virtual (ingress) ports. Each switching element gets certain portion of bandwidth allocated to its uplink port. The total bandwidth available for the switching component is shared across its ingress virtual ports. This effectively enables distribution of the egress switch port bandwidth between ingress ports (VSIs in case of virtual port exposed to PCIe functions). Each virtual port is configured with its share of the switching element bandwidth.

The default configuration assumes equal bandwidth distribution between ingress ports. This default configuration can be changed using admin queue commands defined in Chapter 38.25.4. By default, bandwidth distribution between switch virtual ports is relative, and if some of virtual ports did not use their bandwidth allocation, the remaining bandwidth can be used by other virtual ports relatively to their original bandwidth share. For example, if bandwidth allocation of three VSIs is configured to 10%, 30% and 60% respectively; and if VSI_2 (configured to 60%) does not use its bandwidth; then VSI_0 and VSI_1 can share the remaining 60% of wire bandwidth relative to their original bandwidth allocation (which was 10% for VSI_0 and 30% for VSI_1). The new bandwidth distribution would be 25% for VSI_0 and 75% for Port1.

Note that bandwidth ratio between two VSIs remains the same (1:3). In the relative bandwidth allocation mode, all entities that can be scheduled are assigned non-zero bandwidth share, see Figure 38-74 for an example of bandwidth distribution.

Bandwidth allocation is not accumulative. If a virtual port did not use its bandwidth share, for any reason (such as no work available, bandwidth limit, priority flow control, etc.), it does not accumulate any additional bandwidth and next time it becomes active, it is allowed to consume its allocated bandwidth share, regardless a period of staying idle and not using bandwidth.

Granularity of the bandwidth allocation per virtual link is 1% of the bandwidth available for that virtual link.The virtual port propagates its bandwidth allocation to the uplink egress port of the switching element, or to the PCIe function owning this virtual port. This bandwidth allocation can be in turn shared by virtual ports/VSIs of the switching element or by transmit queues of respective PCIe function.

Figure 38-74.Example of Bandwidth Distribution

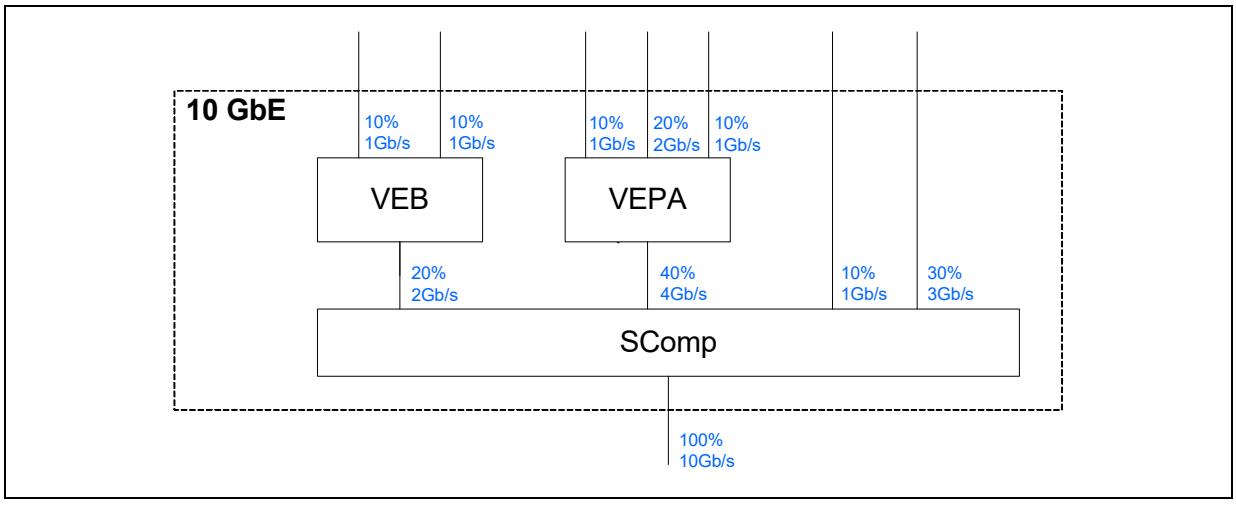


Figure 38-74 shows an example of bandwidth distribution within switching hierarchy. S-comp uplink port is connected to the 10 GbE physical port of the device, and owns all the bandwidth available on the wire. S-comp has four virtual ports: two VSIs is two S-channels connected to the VEB/VEPA switching components and bandwidth allocation for each port is as shown in Figure 38-74. Note that bandwidth distribution is relative, and each virtual port gets a percentage of the bandwidth available for the S-comp. For example if the total amount of bandwidth available for S-comp is decreased to half, then bandwidth available for each virtual port is effectively halved, or if some virtual ports will not use a bandwidth allocated for them, other virtual ports can use the remaining bandwidth. VEB switching component is configured to get 20% of the bandwidth available for the S-comp. Bandwidth available for the VEB in turn is equally distributed between VEB VSIs.

38.25.1.2.1 Arbitration Schemes

The 10 GbE controller supports three types of arbitration schemes: weighted strict priority, weighted round robin and combination of both. The arbitration scheme can be specified by software as a part of the bandwidth distribution definition.

A weighted strict priority scheme enables software to assign different bandwidth share priorities to scheduled entities. Multiple entities can be configured with weighted strict priority. Entity with higher priority is allowed to fully consume its bandwidth share before virtual port with lower priority can use a bandwidth share allocated to it. This scheme enables one or more lightly loaded ingress virtual ports with high bandwidth share priority to use its bandwidth as soon as data becomes available on those ports independent of the ports with lower priority. If a virtual port is configured to weighted strict priority, software can specify its bandwidth share to be unlimited by allocating 127 bandwidth share credits to the virtual port. Strict priority arbitration scheme is a weighted strict priority with an unlimited bandwidth allocation credits.

A weighted round robin scheme enables virtual ports to use allocated portions of bandwidth in round robin fashion. In this configuration, all virtual ports are assumed to have the same bandwidth allocation priority and ports are interleaving in round robin sequence while using a portion of the bandwidth allocated to the port.

A combined configuration enables software to specify per switching component two sets of ingress virtual ports: one sharing bandwidth using weighted strict priority scheme and another sharing bandwidth using weighted round robin scheme. Virtual ports belonging to the weighted round robin group can use their bandwidth share only if all virtual ports in weighted strict priority group either fully used their bandwidth share or cannot use their share due to other restrictions (such as no data available for the port, or the port exceeded its static rate limit).

Software is allowed to modify bandwidth allocation of the virtual port on the fly using the Admin Queue commands described in Section 38.25.4. New bandwidth allocation can take effect after a few scheduling cycles.

38.25.1.3 Static Rate Limiting

The 10 GbE controller enables the configuration of a maximum bandwidth limit for each virtual port of switching element (both ingress and egress ports) and for the TCs/UPs on ETS/SLA enabled ports. This limit specifies the maximum amount of bandwidth that can be consumed by the virtual port. Limiting maximum bandwidth of the uplink (egress) port of the switching element effectively limits the total bandwidth that can be used by all virtual (ingress) ports of that switching element. This causes propagation of the maximal bandwidth limit through the switching hierarchy. Maximum bandwidth allocated for a virtual port should not exceed the maximum bandwidth of the respective uplink port (if configured). The sum of the bandwidth limits of all virtual ports of the switching element can exceed a bandwidth limit of uplink port to enable efficient bandwidth utilization.

Maximum bandwidth limit can be configured for the ETS TC groups and TCs. See Section 38.25.1.4 for more detail.

Maximum bandwidth limits can be used to share bandwidth between VSIs. This is not the most efficient method, but it is still a supported bandwidth distribution scheme. In this case, relative bandwidth allocation should match a maximum bandwidth limit for each port, and the sum of bandwidth limits should add up to the total bandwidth available for the switching element.

Figure 38-75.Example of Limiting Maximum Bandwidth

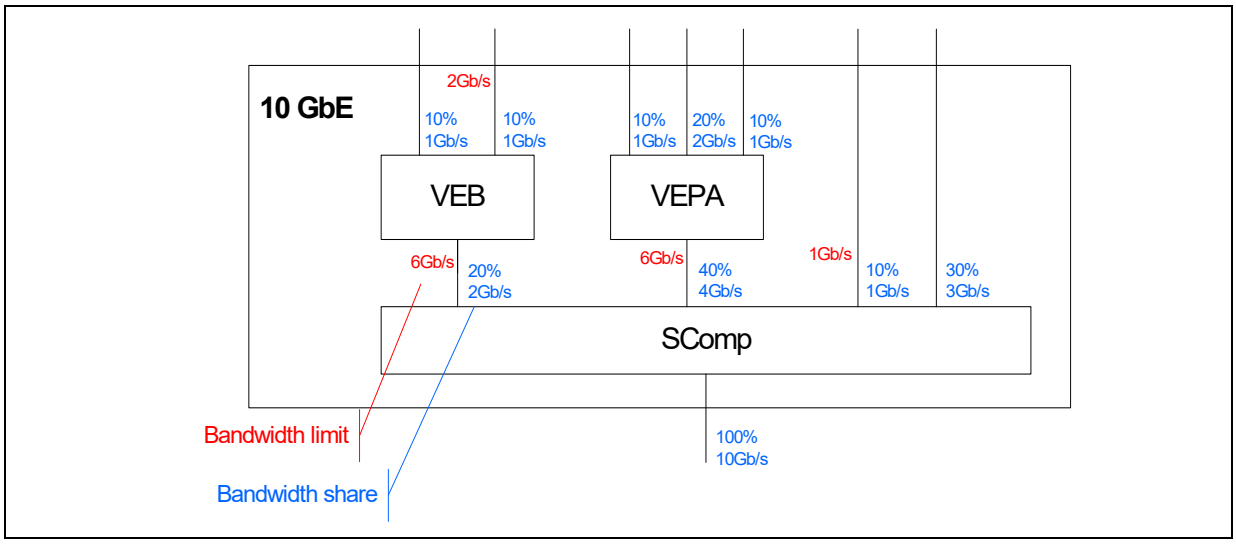


Figure 38-75 shows an example of bandwidth distribution between switching elements and their virtual ports in conjunction with limiting the maximum bandwidth available for some of virtual ports. Bandwidth limit on the uplink port of VEB limits total bandwidth available for that switching element (such as 6 Gb/s for the VEB uplink port on Figure 38-75). Some virtual ports of the switching element might be bandwidth limited as well (like the right side VSI of VEB is limited to 2 Gb/s). Bandwidth limit of an individual virtual port should be less or equal to the bandwidth limit of the switching element or its uplink port. However, the sum of bandwidth limits of the virtual ports should (for the efficient bandwidth use) exceed a bandwidth limit of the switching element. For example, the sum of bandwidth limits of S-comp virtual ports is 13 Gb/s, while the total available bandwidth is
10 Gb/s.

The 10 GbE controller enables configurable accumulation of the bandwidth limit credits. If a virtual port, with static rate limit enabled, does not use allowed bandwidth, the port can accumulate rate limiting credits and use more bandwidth than allowed by static rate limiter up to the configurable limit for the burst of quanta bytes. The 10 GbE controller maintains single quanta for the entire device; this is configured via the TSCDQUANTA register.

The 10 GbE controller static rate limiters can be configured to the minimal rate limit of 50 Mb/s with a granularity of 50 Mb/s. The maximum rate limit supported by the 10 GbE controller is 40 Gb/s.

Static bandwidth limits can be configured to VSIs and switching components in all supported configuration schemes. See Section 38.25.2 for more detail.

38.25.1.4 ETS

In a DCB enabled environment, a single physical port or VSI can be shared by multiple traffic types. The ETS standard defines sharing of the virtual or physical link bandwidth by multiple types of traffic. Different types of traffic are segregated in TCs based on UPs. Each TC is mapped to one or more UPs. Mapping of UPs to the TCs and sharing of bandwidth of VSI by TCs is defined by the ETS specification. Though ETS bandwidth allocation is defined for the egress ports of the networking devices, it also might apply to the ingress port configuration of adjacent devices sharing the same physical or virtual link.

The 10 GbE controller enables ETS to be enabled and independently configured for the uplink ports and VSIs. When enabled, ETS defines the number of TCs allocated per port, UP-to-TC mapping and distribution of port bandwidth between TCs. Due to limited amount of scheduling resources, the 10 GbE controller limits the number of TCs/UPs that can be installed to average of two per VSI.

Per-traffic type bandwidth allocation of VSIs and physical port ETS are independent, each VSI or switching component can be configured with different number of TCs and different bandwidth distribution. However, ETS configuration of the entire internal switching fabric must be consistent. For example, TCs enabled for VSI, must be enabled for respective switching components and physical port, and a total bandwidth allocated to all VSIs for particular TC must be equal to the bandwidth allocated for that TC on switching component.

The 10 GbE controller supports a mixed bandwidth allocation for the TCs, when some TCs are configured as strict priority TCs and others are configured as ETS TCs. Strict priority TCs have a scheduling priority over ETS TCs and can consume unlimited amount of bandwidth. Strict priority TCs are intended to be lightly loaded TCs carrying high priority traffic that does not consume much bandwidth. ETS TCs are sharing bandwidth unused by strict priority TCs relatively to their bandwidth share allocation. See Section 38.25.1.2.1 for the description of arbitration schemes.

38.25.1.5 Transmit Queues and Queue Sets Assignment

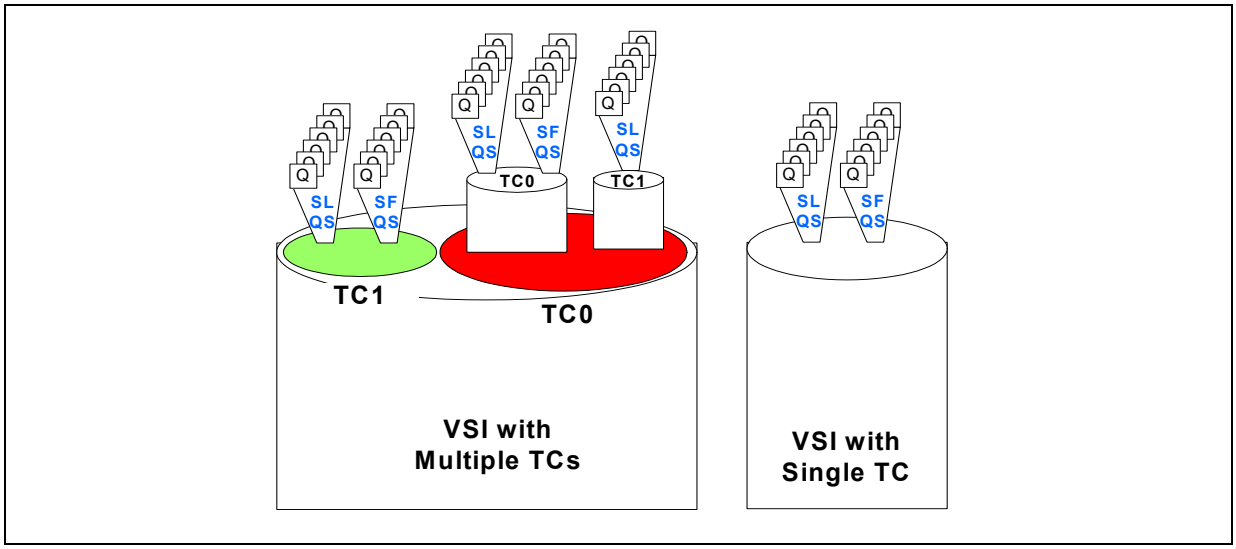
The 10 GbE controller supports two types of queue sets: stateful and stateless queue sets. Each queue set can be associated with multiple transmit queues. A stateful queue set is a set of transmit queues carrying accelerated RDMA or User-space Direct Access (UDA) traffic. A stateless queue set is a set of transmit queues carrying unaccelerated traffic generated by the networking stack.

A VSI can have multiple sets of transmit queues associated with it.

A VSI configured to the carry single type of traffic (one TC) has a one pair of transmit queue sets assigned to it: one stateful queue set and one stateless queue set.

A VSI configured to carry multiple types of traffics has one pair of queue sets allocated for each TC configured for the port.

Figure 38-76.Queue Qet Assignment Diagram



Allocation of queue sets to VSIs is managed by firmware. This is done either at VSI creation time, or when ETS/SLA configuration of the VSI is enabled or modified by software. See Section 38.25.4.8 and Section 38.25.4.17.

Bandwidth allocation between queue sets in the pair is configured separately in addition to bandwidth distribution between VSIs and ETS bandwidth distribution within VSI.

All queues assigned to the same queue set belong to the same TC, and have even bandwidth allocation.

Assignment of the queues to queue sets is performed by software.

To comply with function-level reset requirement, all queues associated with the same queue set must belong to the same PCIe function.

38.25.1.5.1 Queue Set Reassignment

Certain modifications of scheduler configuration such as changing number of TCs enabled for VSI (Section 38.25.4.8) can result in reassignment of active transmit queues from one queue set to another. Transmit queue reassignment involves changing of the transmit queue context to carry a new queue set handle.

If both queue sets remain active after configuration change, then software can perform transmit queue reassignment lazily in the background in parallel with the regular scheduling operation. Transmit queues might get scheduled once via old queue set before switching to the scheduling using a new queue set.

If an old queue set should be deallocated, then software should:

- Upon completion of the configuration change, start reassigning transmit queues to the new queue set.

- Meanwhile hardware might keep scheduling transmit queues using old queue set. The re-assigned transmit queue might be scheduled using an old queue set at most once.
- When the transmit queue reassignment completes, software should notify firmware using the Release Queue Set AQ command (Section 38.25.4.9) that it has completed its part of transition and that firmware can proceed with queue set deallocation.

Software should not request release of the queue set, if it still has transmit queues assigned to it.

38.25.2 Transmit Scheduler Configuration Schemes

The 10 GbE Ethernet Controller scheduler supports wide variety of configuration schemes. Programming interfaces described in Section 38.25.4 enable software to configure ETS-based configuration schemes described in Section 38.25.2.1.

38.25.2.1 ETS-based Scheduler Configuration Scheme

Figure 38-77 shows a logical diagram of ETS-based configuration scheme.

Figure 38-77.ETS-based Scheduler Configuration Scheme

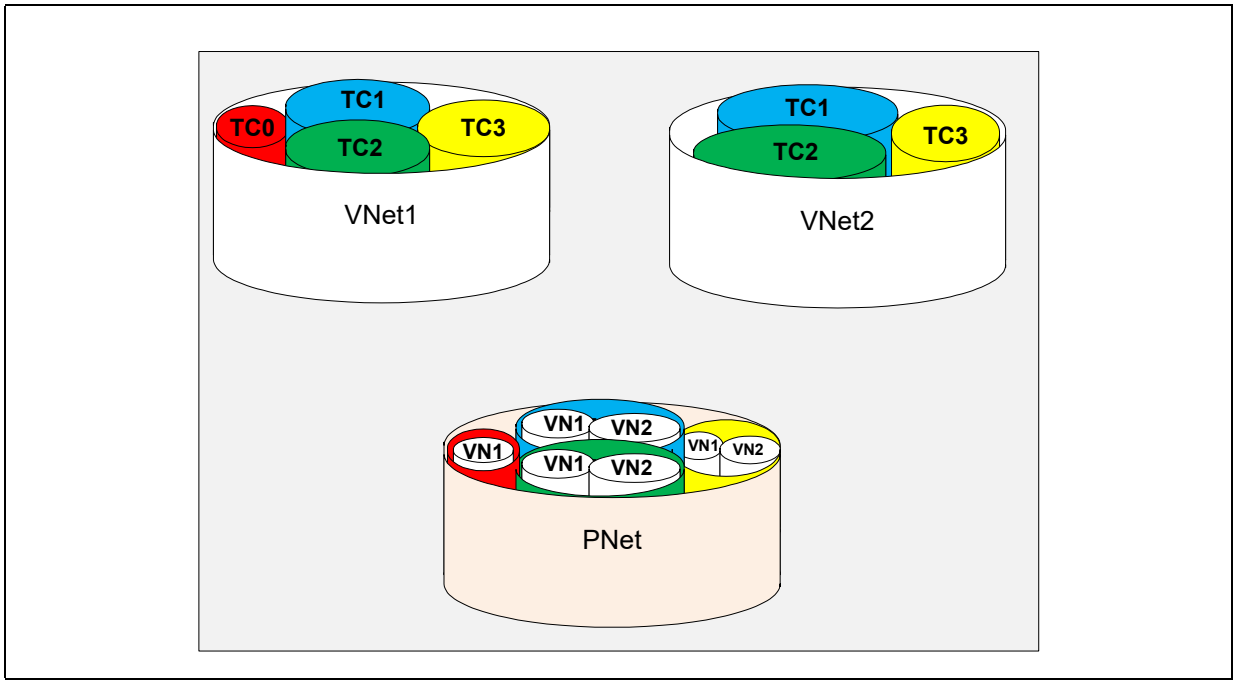


Figure 38-77 shows two virtual vetworks. Each virtual network carries multiple types of traffic. Each traffic type can be identified by its TC or UP. Virtual networks are merging to the physical network. Physical network shows a bandwidth distribution between virtual networks and their traffic types in an ETS-based scheduler configuration scheme.

In the ETS-based configuration, bandwidth is first distributed between types of traffic on the wire and then between virtual networks within each traffic type. Note that traffic type bandwidth distribution might be more complex than shown in Figure 38-77 and might include multiple bandwidth distribution layers. These might include bandwidth

distribution between TCs and then between UPs within a TC. Similarly, bandwidth distribution between virtual networks might involve multiple bandwidth distribution levels (such as virtual switches and their virtual ports).

If a virtual network does not have traffic currently available of a particular traffic type (like VNet2.TC3), then the physical network attempts to preserve ratio of bandwidth allocated for that traffic type and the remaining bandwidth is distributed between other virtual networks having the same traffic type available (like VNet1.TC3).

Such per-traffic type bandwidth distribution does not guarantee any bandwidth allocation for the virtual network, but does allow full control over traffic distribution on the wire per traffic type (ETS). This scheme does guarantee virtual network bandwidth allocation per type of traffic (TC or UP).

This configuration is very suitable for the DCB-enabled fabric. As long as traffic is available for each traffic type enabled on the virtual network the bandwidth distribution on the physical wire matches the ETS configuration of the physical port.

The ETS-based configuration scheme enables designers to configure an ETS for the switching component or VSI connected to the physical port, and hierarchically distribute per-traffic type bandwidth between all other switching components (VEB/PA and port extender) and VSIs.

Figure 38-78 shows an example of bandwidth distribution in an ETS-based scheduler configuration scheme.

Figure 38-78.ETS-based Bandwidth Configuration Example

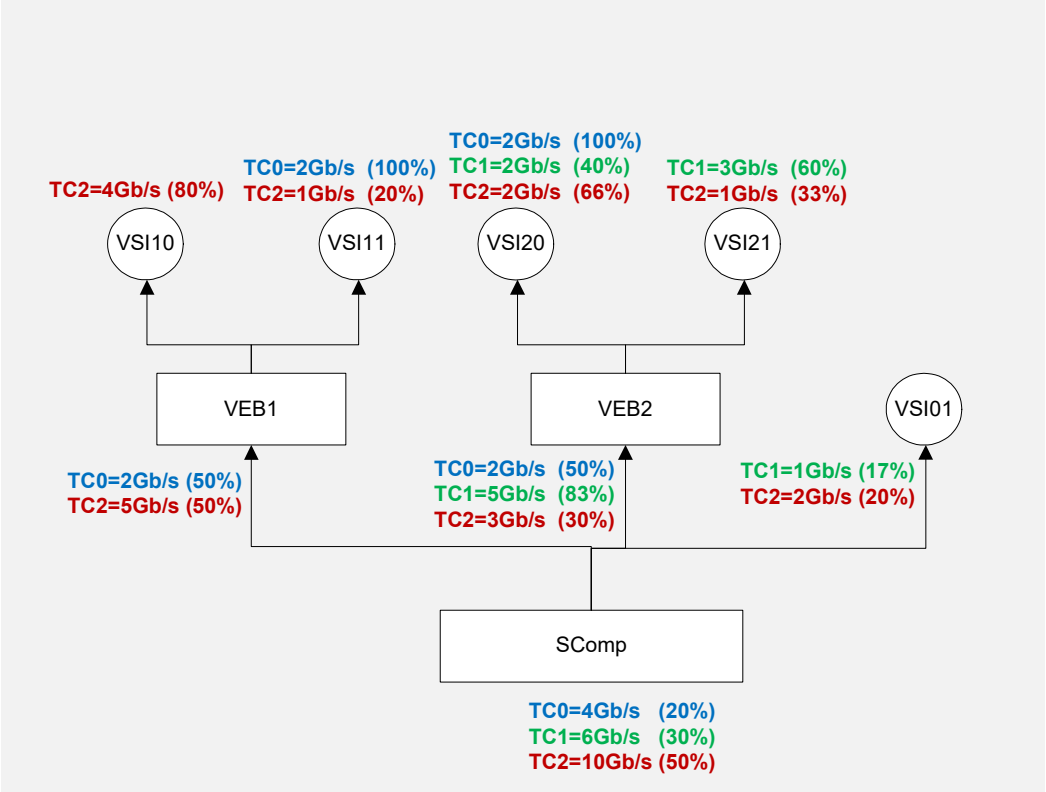


Figure 38-78 shows a single port configuration with S-comp. Two VEB switching components are connected to S-comp (VEB1 and VEB2), along with VSI (VSI01). Each VEB has two VSIs (VSI10 and VSI11, and VSI20 and VSI21, respectively). Each one of the switching components is configured with bandwidth allocation per traffic type. The physical port is configured with ETS, showing one level of bandwidth distribution (per TC). The physical port can be configured with two level ETS including bandwidth allocation per TC, and bandwidth allocation per UP within respective TC.

Bandwidth allocation between switching components and VSIs on each level of switching topology is relative within same TC and hierarchical. Figure 38-78 shows an example bandwidth allocation in Gb/s and relative percentage. Color coding helps to visualize bandwidth distribution between switching components and VSIs within TC.

- S-comp level.
- Physical wire bandwidth is 20 Gb/s. This bandwidth is distributed between three TCs enabled for S-comp (TC0, TC1 and TC2) accordingly with wire ETS configuration.
- TC0=20%, TC1=30% and TC2=50%, total of 100% bandwidth available on the wire
- VEB Level (VEB1, VEB2 VSI01).
- Per TC bandwidth is distributed between VEBs and VSI. Bandwidth distribution is relative. Total of 100% of bandwidth per TC.
  - TC0: VEB1=50%, VEB2=50%
  - TC1: VEB2=83%, VSI01=17%
  - TC2: VEB1=50%, VEB2=30%, VSI01=20%

Total effective bandwidth (in Gb/s) allocated for TCs on VEB level equals to the bandwidth allocated per TC for S-comp. (such as VEB1.TC0 + VEB2.TC0 = SComp.TC0, VEB2.TC1 + VSI01.TC1 = SComp.TC1, and VEB1.TC2 + VEB2.TC2 + VSI01.TC2 = SComp.TC2).

- VSI Level (VSI10, VSI11) and (VSI20, VSI21).
- Per TC bandwidth is distributed between VSIs within respective VEB. Bandwidth allocation for each TC is relative within VEB.
- VEB1 VSIs:
  - TC0: VSI11=100%
  - TC1: 0%
  - TC2: VSI10=80%, VSI11=20%
- Total effective bandwidth (in Gb/s) allocated for TCs on VEB1 VSI level (VSI10 and VSI11) equals to the bandwidth allocated per TC for VEB1. (such as VSI10.TC0 + VSI11.TC0 = VEB1.TC0, and VSI11.TC2 = VEB1.TC2)
- VEB2 VSIs:
  - TC0: VSI20=100%
  - TC1: VSI20=40%, VSI21=60%
  - TC2: VSI20=66%, VSI21=33%

Total effective bandwidth (in Gb/s) allocated for TCs on VEB2 VSI level (VSI20 and VSI21) equals to the bandwidth allocated per TC for VEB2. (such as VSI20.TC0 = VEB2.TC0, VSI20.TC1 + VSI21.TC1 = VEB2.TC1, and VSI20.TC2 + VSI21.TC2 = VEB2.TC2).

Bandwidth allocation must be consistent. Bandwidth allocated to the TC on the S-comp level, must be equal to the total bandwidth allocated for the same TC on the level of VEBs and VSIs connected directly to the S-comp (VEB1, VEB2 and VSI01). Bandwidth allocated to the TC on VEB (VEB1) level must be equal to the total bandwidth allocated to that TC on VSI level (VSI10 and VSI11).

The ETS-based scheduler configuration scheme does not support bandwidth allocation per switching component or VSI. It supports only bandwidth allocation for switching component or VSI within particular TC.

ETS-based scheduler configuration scheme enables software to instantiate bandwidth limits for each of the switching components and VSIs. Bandwidth limit can be programmed to restrict total bandwidth available for switching component and VSI, or limit bandwidth available for the switching component or VSI for the particular type of traffic, identified by UP or TC.

38.25.3 Scheduling

The The 10 GbE controller scheduler operates with queue sets. Every scheduling cycle scheduler selects a next queue set to be served. The amount of traffic that can be generated on the wire by transmit queues associated with the selected queue set is limited to the configurable value (Quanta). Selection of the queue set is based on the bandwidth share, bandwidth limit and ETS configuration of the VSI and uplink ports of the internal switching components. The scheduler fairly distributes available bandwidth among queue sets that have transmit queues with work available staying within limits of the bandwidth allocation dictated by bandwidth limit, bandwidth share and ETS constraints. Bandwidth distribution between queues within same queue set is intended to be even.

The scheduler limits amount of data that can be transmitted by selected queue set to the configurable value (Quanta). Single quanta might enable transmission of multiple Ethernet frames. Quanta constrains the amount of burst of the traffic generated by the 10 GbE controller. Independent of the rate limit implied on the queue set, the 10 GbE controller transmits a Quanta worth of data with the wire speed of the associated physical port.The default Quanta value is 4 KB.

Scheduler configuration is derived from the configuration of internal switching elements, and in most cases follows hierarchy of the switching elements.

The 10 GbE controller does not schedule individual packets. All packets belonging to the same transmit queue follow same route within internal switching hierarchy and are subject to the same set of bandwidth constraints.

When the 10 GbE controller transmits packet from one of the transmit queues of the selected queue set, it effectively uses bandwidth of all switching components in internal switching hierarchy leading from the virtual port that queue set is associated with to the physical port of the device. Bandwidth constraints applied to the switching component on the route from the queue set to physical port of the device automatically apply to the queue set. This means that a particular queue set can be scheduled only if there is enough bandwidth available in all switching components on the transmission route.

The scheduler is aware of the PFC, and stops scheduling queue sets associated with a paused TC to avoid head of line blocking in pipeline.

38.25.3.1 Scheduler Configuration Process

The scheduler implements one set of configuration tables shared by all PCIe functions. To synchronize access to the shared configuration table and keep configuration consistent, direct access to the internal scheduler configuration is restricted to firmware.

The scheduler does not expose its internal configuration tables to the standard deployment software. A set of registers enabling a direct access to the internal scheduler structures is exposed to firmware to enable programming of the scheduler configuration tables. Those registers are exposed in debug mode to bring up and other privileged software. A default configuration of the scheduling tables is done by firmware and based on the internal switch and scheduling configuration profiles kept in NVRAM. Software is allowed to modify the default configuration by proxying its requests through firmware using Admin Queue commands. Software should use an admin queue interface to communicate its scheduler configuration requests with firmware. Firmware is responsible to constrain access to the scheduling configuration tables based on the software privilege level.

The 10 GbE controller maintains single Quanta for entire device, configured via the TSCDQUANTA register. Quanta accounts for payload and headers generated by software for the LAN traffic and payload and headers generated by PE for RDMA traffic. Quanta does not account for the L2 tags, Ethernet, MPA, CRCs, MPA markers and RDMA padding inserted by hardware on the fly.

The scheduler is intended to be configured by a PF driver using the Admin Queue commands described in Section 38.25.4. A PF driver is considered to be a trusted software, and firmware should accept requested configuration changes. Most of the switching components are owned either by single PF or by one of the VFs associated with same PFs. If a switching component is shared among physical functions, this component is either configured by an EVB agent running in firmware, or by service running in one of the PFs. In the later case, firmware relies on software to coordinate scheduler configuration among different PFs.

Most of the scheduler configuration is fully controlled by a PF driver in the standard deployment environment. If a VF driver or configuration software running in VFs is granted control over one of the scheduling configuration parameters (such as the ETS/SLA configuration of the virtual port, scheduler configuration commands should be proxied via PF driver).

The scheduler supports up to 768 pairs of queue sets, regardless of the number of queue sets populated in each pair.

38.25.4 Admin Queue Commands

The internal structure of the scheduler configuration tables is not exposed to software. Firmware is responsible for the scheduler configuration and provides software with admin queue interface enabling alter scheduler configuration. In some configurations, a privileged software is allowed to perform direct programming of the scheduler configuration tables bypassing firmware. Firmware or privileged software should maintain a mapping table of the scheduling resources assigned to each PCIe function.

This section defines admin queue commands that should be used by software to program scheduler configuration attributes.

A standard Product Name scheduler configuration is based on the configuration of the internal switch and its components including instantiated switching components, their VSIs, S-channels, and connectivity between internal switching component. Scheduler configuration adds bandwidth management attributes to the configured switching components, including allocation of bandwidth for the switching components, and their VSIs, enablement and configuring ETS or bandwidth allocation per TC, and instantiation of rate limiters.

The AQ commands described in this section are intended to be used by PF driver only. A VF driver is not allowed to directly participate in scheduler configuration. The PF driver is considered to be a trusted software component within this PF. Firmware validates that AQ commands impacting configuration of the components owned by the PF driver issuing those commands. This is done using association of the AQ with particular PF and information kept in the switch configuration tables. When PF performs configuration on behalf of VF, corresponding VF must be provided within an AQ command.

Table 38-302 lists the AQ commands enabling configuration of internal scheduler structures.

Firmware must avoid partial command execution. All command validation and resource availability checks must be done prior to updating scheduler configuration tables. Note that aborting a command with a partially updated scheduler configuration table might lead to unpredictable hardware behavior.

Table 38-302.Scheduler Configuration Admin Queue Commands (Sheet 1 of 2)

Command	Opcode	Brief Description	Detailed Description
Configure VSI Bandwidth Limit	0x0400	This command enables software to configure bandwidth limit to the specified VSI of the specified switch component. This is a global bandwidth limit that applies to all TCs enabled for VSI.	Section 38.25.4.6
Configure VSI Bandwidth Limit Per Traffic Type	0x0406	This command enables configuration bandwidth limits assigned to TCs of the specified VSI exposed to PCIe interface.	Section 38.25.4.7
Configure VSI Bandwidth Allocation Pper Traffic Type	0x0407	This command enables specifying traffic types enabled for VSI and configure relative bandwidth allocation of VSI within each TC type. This command is valid for ETS-based configuration only.	Section 38.25.4.8
Query VSI Bandwidth Configuration	0x0408	This command enables retrieving configuration of the specified VSI of the specified switching component. This configuration should include bandwidth limit and bandwidth allocation.	Section 38.25.4.15
Query VSI Bandwidth Configuration Per Traffic Type	0x040A	This command enables retrieving current bandwidth configuration of VSI within each TC type. This command is valid for ETS-based configuration only.	Section 38.25.4.16
Configure Switching Component Bandwidth Limit	0x0410	This command enables software to enable, disable or modify bandwidth limit of the egress port of specified switching component. This is a global bandwidth limit that applies to all TCs enabled for a switching component.	Section 38.25.4.9
Enable Physical Port ETS	0x0413	This command enables ETS configuration of the egress port of the specified switching component or VSI directly connected to the physical port. It carries full specification of ETS configuration for the port, including number of TCs. user priority to TC mapping, TC bandwidth allocation, TC arbitration scheme, and bandwidth allocation. This command is valid for switching components directly connected to the physical port only. This command is valid in ETS-Based configuration only.	Section 38.25.4.10

Table 38-302.Scheduler Configuration Admin Queue Commands (Sheet 2 of 2)

Command	Opcode	Brief Description	Detailed Description
Modify Physical Port ETS	0x0414	This command enables modifying ETS configuration of the egress port of the specified switching component or VSI directly connected to the physical port. It carries full specification of ETS configuration for the port, including number of TCs. UP to TC mapping, TC bandwidth allocation, TC arbitration scheme, and bandwidth allocation. This command is valid for switching components directly connected to the physical port only. This command is valid in ETS-Based configuration only.	Section 38.25.4.10
Disable Physical Port ETS	0x0415	This command enables disabled ETS configuration of the egress port of the specified switching component or VSI directly connected to the physical port. It carries full specification of ETS configuration for the port, including number of TCs. user priority to TC mapping, TC bandwidth allocation, TC arbitration scheme, and bandwidth allocation. This command is valid for switching components directly connected to the physical port only. This command is valid in ETS-based configuration only.	Section 38.25.4.10
Configure Switching Component Bandwidth Limit Per Traffic Type	0x0416	This command enables enabling, disabling or modifying bandwidth limits assigned to TCs of the egress port of the specified switching component. This command is valid in ETS-based configuration only.	Section 38.25.4.13
Configure Switching Component Bandwidth Allocation Per Traffic Type	0x0417	This command enables specifying traffic types enabled for switching components and configure relative bandwidth allocation of switching components within each TC type. This command is valid for ETS-Based configuration only.	Section 38.25.4.14
Suspend Port's TX Traffic	0x041B	This command enables PF to suspend a port's transmit traffic. The command is completed after all Qsets belonging to the port are suspended and the Tx pipe of the port is drained. This enable software to modify a port's configuration like DCB setting. This command is valid only under SFP mode.	Section 38.25.4.11
Resume PF Traffic	0x041C	This command is used to resume suspended Qsets. It resumes all Qsets belonging to the PF.	Section 38.25.4.12
Query Switching Component Bandwidth Configuration	0x0418	This command enables retrieving current configuration of the switching component.	Section 38.25.4.17
Query Physical Port ETS Configuration	0x0419	This command enables retrieving current ETS configuration of the switching component. This should include number of valid TCs bandwidth allocation and bandwidth limit for TCs. This command is valid for switching components directly connected to the physical port only. This command is valid in ETS-based configuration only.	Section 38.25.4.18
Query Switching Component Bandwidth Configuration per Traffic Type	0x041A	This command enables retrieving current bandwidth configuration of a switching component within each TC type. This command is valid for ETS-Based configuration only. This command is valid only under SFP mode.	Section 38.25.4.19
Add VSI	0x0210	This is a switch configuration command. Execution of this command affects scheduler configuration. This command allocates VSI and returns a VSI SEID that should be used to modify VSI bandwidth configuration.	Section 38.25.4.2
Add VEB, Add_PE	0x0210	This is a switch configuration command. Execution of this command affects scheduler configuration. This command allocates switching component, and returns SEID that should be used to modify switching component bandwidth configuration.	Section 38.25.4.2
Delete Element	0x0210	This is a switch configuration command. Execution of this command affects scheduler configuration.This command deallocates switching elements including VSIs. Deallocation of the switching element results in deallocation of he scheduling resources associated with this switching element, and cleanup of scheduler configuration tables.	Section 38.25.4.2

38.25.4.1 Scheduler Initialization Flow

Main scheduler configuration and initialization flows are initiated by software and performed by firmware using AQ interface. The sections that follow define various AQ commands that can be used by software to configure the transmit scheduler.

Prior to initialization of AQ command interface, firmware can use information available in NVRAM to perform default configuration of the scheduler. Similar to the standard configuration, default scheduler configuration follows a default configuration of the switching components.

38.25.4.2 Allocation of Switching Elements

A default configuration of the scheduler tables is performed as a part of the internal switch configuration. Each time software adds a new switching component or VSI to the internal switch, the configuration tables of the scheduler are updated respectively by firmware. A new added switching component or VSI is configured with a default bandwidth allocation and bandwidth limit disabled. The allocated switching component or VSI is created with single TC enabled (TC0). Software can enable TCs either by using the bitmask in the Add VSI or Add VEB/PA commands, or by using the AQ command described in Section 38.25.4.7 and Section 38.25.4.14. Along with default bandwidth, distribution firmware allocates a pair of queue sets for each configured TC of the new allocated VSI.

With completion of a switching element allocation request, software is provided with a handle per allocated queue set. The queue set handle should be used to associate a queue with a queue set.

A default bandwidth allocation provides a new instantiated switching component and VSI with a minimal bandwidth share with respect to other switching components or ports sharing same virtual link. Software can modify the default bandwidth allocations using the Admin Queue command described in Section 38.25.4.8 and Section 38.25.4.14.

Initial configuration of the scheduler tables is done as a part of the initial configuration of the internal switch and is based on the device profiles. No software involvement is required to perform initial scheduler configuration. As soon as firmware completes initial configuration, scheduler is enabled and ready to schedule transmit work.

Software is allowed to change internal switch configuration or modify scheduler configuration by adding/removing internal switching components and/or VSIs, or by changing bandwidth distribution, bandwidth limits and ETS configuration of physical port, switching components or VSIs. Some configuration changes, causing modification of the internal switching structure, might require firmware to partially or fully suspend scheduler operation until it completes a local or global modification of the internal scheduler tables. Certain scheduler configuration changes, such as bandwidth redistribution, or modification of the bandwidth limit, can be done without suspending normal scheduler operation.

Firmware is responsible to track allocated resources and their configuration, and manage configuration of internal scheduler tables. If software performs incremental modification of the internal switching structure, it should rely on firmware to maintain consistency of internal scheduler tables, and fit requested modifications to the existing configuration. Each allocated switching element, including instance of internal switch or VSI is assigned a unique within the device identification number (SEID). SEID should be used by software when referencing to the switching component during scheduler configuration.

38.25.4.2.1 Queue Set Assignment

As a part of the default scheduler configuration, each instantiated VSI is supplied with a pair of queue sets per configured TC: one for stateless and one for the stateful queues. Firmware manages the shared pool of the queue sets. Allocation and deallocation of queue sets is hidden from software, and done based on the software resource allocation/deallocation requests. Handles of allocated queue sets are returned to software as a part of completion of Admin Queue command and should be used by software to associate transmit queues and queue sets. Firmware allocates a single queue set handle to the LAN/PE queue sets pair.

Software can change a default configuration of the VSI. (like change the number of configured TCs). A change in VSI configuration results in changing the number of TCs allocated for the port, causing firmware to reallocate queue sets assigned to the VSI to match number of TCs.

Allocated queue set handles are provided in completion of respective AQ command. Firmware always returns eight queue set handles. Order of handles returned matching an order of TCs enabled for VSI (from 0 to 7). Invalid queue set handles carry value of 0xFFFF. Queue set handles are returned for all AQ commands that can be result in allocation or change of allocation of queue sets (create VSI, configure VSI bandwidth allocation per traffic type, configure VSI bandwidth limit per traffic type). For more detail see Section 38.25.4.7 and Section 38.25.4.8. Software can also use AQ command to query queue sets allocated for particular VSI, see Section 38.25.4.16.

Queue set assignment is completely hidden from software. Software should use a queue set handle when referring to the particular queue set.

Assignment of queues to queue sets is not described here, and outside of the scope of the scheduler configuration definition. If software changes configuration of the VSI and this change results in reallocation of queue sets, as long as TC remains assigned to the VSI, all queues associated with the queue set identified by the VSI and TC remain associated with the queue set. If software modifies the VSI configuration and eliminates the TC, which has an active queue associated with, software must resolve contention and reassign queues.

38.25.4.3 Release of the Switching Elements

Software can use the AQ Commands Delete Element and the Configure Bandwidth Allocation/Limit of the VSI/Switching Component per Traffic Type to remove switching components and VSIs or change their configuration per traffic type. Removal of the switching component or VSI is allowed only if the respective uplink switching element has been previously removed or a traffic type configuration has been previously adjusted to the uplink switching element. For example, if software decides to change VEB configuration per traffic type and reduce number of traffic types enabled for VEB, it must first respectively adjust traffic types for all VSIs allocated for that VEB. Software might request to remove VEB, only after it removed all VSIs allocated for that VEB.

Software might also request to remove VSI only if the queue set associated with that VSI has all transmit queues removed, see Section 38.25.4.3.1.

38.25.4.3.1 Queue Set Release

Certain modifications to the scheduler configuration, such as change in number of TCs enabled for the VSI, or change in TC mapping or VSI deallocation, might lead to release of the queue set previously allocated for VSI.

Prior to performing any operation that might require release of the queue set, software must remove from the queue set all queues that were previously associated with the queue set. Operation of the queue removal depends on the queue type. For the LAN, this operation requires Q disable. For the PE Qs, Q might have transmission suspended. In any case, software must either suspend or disable all Qs that were previously associated with the queue set. Completion of Q suspend or Q disable operation must be confirmed by hardware.

Scheduler configuration firmware validates that a queue set subject to release does not have Qs associated with, and fails the requested operation, reporting EBUSY error, if it does.

38.25.4.4 Common Processing and Error Handling

All Transmit Scheduler Admin Queue Commands described in the sections that follow are derived from the generic Admin Queue commands, and using a generic Admin Queue Command structure as a baseline.

Sections describing Transmit Scheduler Admin Queue commands are relating to the Admin Queue command fields that are specific to the particular command.

The transmit scheduler uses various types of Admin Queue commands. Some of them Direct Admin Queue commands (all command information is provided within the body of command) and some indirect (additional data is provided within the buffer referred by Admin Queue command). Some commands report completion within Admin Queue command body and others carry completion information within buffer provided by original command. Each Transmit Scheduler Admin Queue command indicates its type in the command description.

Table 38-303 lists errors specific to Transmit Scheduler Admin Queue commands and reported in Admin Queue Completions, providing their cause and a reported error code.

Table 38-303.Transmit Scheduler Admin Queue Completion Errors

Error Name	Error Code	Description
Invalid Handle	ENOENT	Upon allocation of the transmit scheduler resources software is provided with various handles, such as TC handle, UP handle, QS handle. Those handles must be used by software for the future reference to the allocated resources. This error indicates that handle provided by software is not valid.
Invalid SEID	ENOENT	SEID provided within Admin Queue command is not valid.
Parameter Out Of Range	ERANGE	Provided argument does not match definition (not within allowed range).
Resource Allocation Failure	ENOSPC	Failed to allocate transmit scheduler resource.
Operation Not Permitted	EPERM	Depending on the scheduler configuration scheme, certain AQ commands should not be used by software (like in ETS-based configuration software should not attempt configure bandwidth allocation to VSI or switching component). This error indicates that software attempted AQ command that is not valid in the current operation.
Resource Is Busy	EBUSY	Software attempted to release a queue set that had Qs associated with it.

38.25.4.5 Function-level Reset

Transmit scheduler configuration tables are updated by firmware as a part of the function-level reset processing by the 10 GbE controller. All scheduling resources allocated for a particular function are released during this process, including VSIs and Queue Sets. Respective resources can be later on allocated to the different PCI Functions.

Scheduler operation is not suspended during function-level reset but its performance and bandwidth distribution might be intermittently affected. Upon completion of function-level reset processing, the transmit scheduler completes its normal operation.

38.25.4.6 Configure VSI Bandwidth Limit

This command enables software to configure a bandwidth limit for the specified VSI of the switch component that is exposed to PCIe interface. To configure bandwidth limit of the egress port, software should use the command described in Section 38.25.4.9. The 10 GbE controller provides hierarchical bandwidth limit, that apply to VSI and all associated TCs and UPs. See Section 38.25.1.3.

Software cannot have both a bandwidth limit and bandwidth limit per traffic type enabled for the same VSI. A request to enable bandwidth limit for VSI that has bandwidth limit per traffic type enabled should fail and report an EPERM error.

In that case, the PF uses more than one TC (such as an iSCSI PF that must be opened for Storage TC and LAN TC as well).

- PF is allowed to enable more than one TC. According enabled UP's and UP to TC mapping.
- Bandwidth relative allocation defined for this PF and the Alt RAM is configured for each one of the used TC's.
- Max bandwidth for the PF is declared as shared rate limiter.

This is a Direct Admin Queue command with completion reported within the completion descriptor structure. Table 38-304 lists the command format and defines command specific fields.

Table 38-304.Configure VSI Bandwidth Limit Command Fields (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	1:0	0	
Opcode	2-3	0x0400	Command opcode.
Datalen	4-5	0	Length of response buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
VSI SEID	16-17		Unique identifier of the VSI.
Reserved1	18-19	0	Reserved. Must be set to 0.
Bandwidth Limit Credits	20-21		Bandwidth limit in Mb/s. Supported range is 50 Mb/s - 40 Gb/s, in increments of 50 Mb/s. 0 indicates that bandwidth limit is disabled. One credit corresponds to bandwidth limit of 50 Mb/s.

Table 38-304.Configure VSI Bandwidth Limit Command Fields (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Bandwidth Limit Max	24.0-24.2		Maximum bandwidth limit indicates how much bandwidth limit credits can be accumulated due to inactivity. Single scheduling Quanta consumes multiple bandwidth limit credits. The 10 GbE controller enables accumulating of following discrete values of Quanta-worth credits: 0,1, 2, and 4 Quanta worth credits. This field is valid, only if bandwidth limit is enabled.
Reserved2	24.3-31	0	Reserved. Must be set to 0.

38.25.4.7 Configure VSI Bandwidth Limit per Traffic Type

This command enables software to specify TCs enabled for VSI and configure a bandwidth limits of TCs enabled for the specified VSI of the switching component.

In an ETS-based configuration scheme, software can configure either VSI bandwidth limit or VSI bandwidth limit per traffic type.

Software should keep track of already configured bandwidth limits and provide complete bandwidth limit configuration, and not just do incremental modifications.

Software is allowed to change the bandwidth limit per traffic type and TCs enabled for VSI that has a VSI bandwidth allocation configured. A new TCs enabled for VSI has a default bandwidth allocated. Software should use a Configure VSI Bandwidth Allocation per VSI AQ command to modify a default bandwidth allocation.

Software should not request change of TC configuration of VSI leading to release or remapping of TCs, prior to disassociating Qs with affected queue sets. Operation fails with EBUSY error, if software requested to release queue sets having Qs associated with.

The only exception for the previous rule is when remapping of only one TC is required. In this case, software is allowed to use this command when this queue set is suspended and the 10 GbE controller transmit pipeline is drained from packets belongs to this queue set. After the TC remapping takes place, software can resume its transmit activity using the command Resume PF Traffic Section 38.25.4.12. EMP firmware is required to keep remapped TCs queue set handles for its new location. If TC re-configuration is needed while the VSI is suspended then new generated TC nodes of this VSI is suspended too.

Software cannot have both bandwidth limit and bandwidth limit per traffic type enabled for the same VSI. Request to enable bandwidth limit per traffic type for VSI that has bandwidth limit enabled should fail and EPERM error reported in completion.

This is an Indirect Admin Queue command. Software should provide a valid buffer carrying additional command attributes as listed in Table 38-306.

Table 38-305 lists the command format, and defines fields specific to the command.

Table 38-305.Configure VSI Bandwidth Limits per Traffic Type Command Fields (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	1:0	0	
Opcode	2-3	0x0406	Command opcode.
Datalen	4-5		Length of response buffer.

Table 38-305.Configure VSI Bandwidth Limits per Traffic Type Command Fields (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
VSI SEID	16-17		Unique identifier of the VSI.
Reserved1	18-23	0	Reserved. Must be set to 0.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

Table 38-306 lists the format of the command buffer and additional command attributes.

Table 38-306.Configure VSI Bandwidth Limit per Traffic Type Command Buffer (Sheet 1 of 2)

Category	Byte/Bit	Field	Description
TC Valid	0.0-0.7	TC0-TC7 Valid	Valid bit per TC. Should list all TCs enabled for VSI. Must be consistent with physical port and uplink switching component.
Reserved1	1-15	0	Reserved. Must be set to 0.
TC Bandwidth Limit Credits	16-17	TC0 BW Limit Credits	This field specifies a bandwidth limit per TC in Mb/s. Supported bandwidth limit range is 50 Mb/s-40 Gb/s, with increments of 50 Mb/s. Bandwidth limit of 0 Mb/s indicates that bandwidth limit is disabled. Bandwidth limit should be provided for the valid TC handles only. One credit corresponds to bandwidth limit of 50 Mb/s.
	18-19	TC1 BW Limit Credits	
	20-21	TC2 BW Limit Credits	
	22-23	TC3 BW Limit Credits	
	24-25	TC4 BW Limit Credits	
	26-27	TC5 BW Limit Credits	
	28-29	TC6 BW Limit Credits	
	30-31	TC7 BW Limit Credits	
TC Max Bandwidth Limit	32.0-32.2	TC0 Max BW Limit	Maximum bandwidth limit indicates how much bandwidth limit credits can be accumulated due to inactivity. Single scheduling Quanta consumes multiple bandwidth limit credits. The 10 GbE controller enables accumulating of following discrete values of Quanta-worth credits: 0,1, 2, 4, 8,16 32 and 64 Quanta worth credits.
	32.3	Reserved	
	32.4-32.6	TC1 Max BW Limit	
	32.7	Reserved	
	33.0-33.2	TC2 Max BW Limit	
	33.3	Reserved	
	33.4-33.6	TC3 Max BW Limit	
	33.7	Reserved	
	34.0-34.2	TC4 Max BW Limit	
	34.3	Reserved	
	34.4-34.6	TC5 Max BW Limit	
	34.7	Reserved	
	35.0-35.2	TC6 Max BW Limit	
	35.3	Reserved	
	35.4-35.6	TC7 Max BW Limit	
	35.7	Reserved	

Table 38-306.Configure VSI Bandwidth Limit per Traffic Type Command Buffer (Sheet 2 of 2)

Category	Byte/Bit	Field	Description
Reserved2	36-63	0	Reserved. Must be set to 0.

Table 38-307 lists the format of completion returned in response buffer.

Table 38-307.Configure VSI Bandwidth Limit per Traffic Type Completion Buffer

Category	Byte/ Bit	Field	Description
QueueSet Handles	16-17	QS0 Handle	Handle per queue set. Completion always carries eight queue set handles. Only TCs that were marked as a valid TCs in the command have valid queue set handles returned in completion. Invalid handles are marked by 0xFFFF. Order of queue set handles matches order of TCs. Queue set handles must be used by software to associated transmit Q with queue set.
	18-19	QS1 Handle	
	20-21	QS2 Handle	
	22-23	QS3 Handle	
	24-25	QS4 Handle	
	26-27	QS5 Handle	
	28-29	QS6 Handle	
	30-31	QS7 Handle	

38.25.4.8 Configure VSI Bandwidth Allocation per Traffic Type

This command enables software to specify TCs enabled for VSI, and configure relative bandwidth allocation of VSIs within each TC.

This command can be used for VSI configured in ETS-based mode only. Otherwise the command is ignored and error is reported.

To configure VSI bandwidth allocation per traffic type, software should provide complete relative bandwidth allocation for all TCs enabled for VSI.

This is an Indirect Admin Queue command. Software should provide a buffer that would be used both to provide additional command attributes and for the command completion.

Software is allowed to change bandwidth allocation per traffic type and TCs enabled for VSI that has a VSI bandwidth limit enabled. It is responsibility of firmware to adjust hardware configuration to reflect this change.

Software should not request a TC configuration change of a VSI leading to release or remapping of TCs, prior to disassociating Qs with affected queue sets. Operation fails with a EBUSY error, if software requested to release queue sets having Qs associated with.

The only exception for the previous rule is when remapping of only one TC is required. In this case, software is allowed to use this command when this queue set is suspended and the 10 GbE controller transmit pipeline is drained from packets belongs to this queue set. After the TC remapping takes place, software can resume its transmit activity using the command Resume PF Traffic described in Section 38.25.4.12. EMP firmware is required to keep remapped TCs queue set handle for its new location. Currently the 10 GbE controller supports transmit pipe draining in port's granularity. This is used by DCBX flows described in Section 38.25.5.6.1.

In that case, the PF uses more than one TC (such as an iSCSI PF that must be opened for Storage TC and LAN TC as well).

- PF is allowed to enable more than one TC. According to enabled UP's and UP to TC mapping.
- Bandwidth relative allocation defined for this PF in the Alt RAM is configured for each one of the used TC's.
- Maximum bandwidth for the PF is declared as a shared rate limiter.

If TC re-configuration is needed while the VSI is suspended then new generated TC nodes of this VSI is also suspended.

Table 38-308 lists the command format and defines a command specific fields.

Table 38-308.Configure VSI Bandwidth Allocation per Traffic Type Command Fields

Name	Bytes.Bits	Value	Remarks
Flags	1:0	0	
Opcode	2-3	0x0407	Command opcode.
Datalen	4-5		Length of response buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by fFirmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
VSI SEID	16-17		Unique identifier of the VSI.
Reserved1	18-23	0	Reserved. Must be set to 0.
Data Address High	24-27		Address of buffer. This buffer is used both to convey configuration to firmware, and provide software with a list of TC handles.
Data Address Low	28-31		

Table 38-309 lists the format of the command buffer and defines command attributes.

Table 38-309.Configure VSI Bandwidth Allocation per Traffic Type Command Buffer

Category	Byte/Bit	Field	Description
TC Valid	0.0-0.7	TC0-TC7 Valid	Valid bit per TC. Should list all TCs enabled for VSI. Must be consistent with the physical port and uplink switching component. At least one TC musts be valid.
Reserved2	1-3	0	Reserved. Must be set to 0.
TC Bandwidth Share Credits	4	TC0 Bandwidth Credits	Relative VSI credits within same TC with respect to other VSIs or the switching components Valid range of credits is 1-127 credits, in increments of the single credit. 127(indicates infinite credits).
	5	TC1 Bandwidth Credits	
	6	TC2 Bandwidth Credits	
	7	TC3 Bandwidth Credits	
	8	TC4 Bandwidth Credits	
	9	TC5 Bandwidth Credits	
	10	TC6 Bandwidth Credits	
	11	TC7 Bandwidth Credits	
Reserved4	12-32	0	Reserved. Must be set to 0.

Table 38-310 lists the format of completion returned in response buffer.

Table 38-310.Configure VSI Bandwidth Allocation per Traffic Type Completion Buffer

Category	Byte/ Bit	Field	Description
QueueSet Handles	16-17	QS0 Handle	Handle per queue set. Completion always carries eight queue set handles. Only TCs that were marked as a valid TCs in the Command have valid queue set handles returned in completion. Invalid handles are marked by 0xFFFF. Order of queue set handles matches order of TCs. Queue set handles must be used by software to associated transmit Q with a queue set.
	18-19	QS1 Handle	
	20-21	QS2 Handle	
	22-23	QS3 Handle	
	24-25	QS4 Handle	
	26-27	QS5 Handle	
	28-29	QS6 Handle	
	30-31	QS7 Handle	

38.25.4.9 Configure Switching Component Bandwidth Limit

This command allows software to enable, disable or modify a bandwidth limit for the specified switch component. By enabling bandwidth limit of switching component, software effectively enables a bandwidth limit on the egress port of the switching component.

This command allows software to enable a global bandwidth limit regardless the traffic type. to enable a bandwidth limit per traffic type, software should use command described in Section 38.25.4.13.

Software cannot have both bandwidth limit and bandwidth limit per traffic type enabled for the same switching component. Request to enable bandwidth limit for a switching component that has bandwidth limit per traffic type enabled fails.

In that case, the PF uses more than one TC (such as an iSCSI PF that must be opened for Storage TC and LAN TC as well).

- PF is allowed to enable more than one TC. According enabled UP's and UP to TC mapping.
- Bandwidth relative allocation defined for this PF in the Alt RAM is configured for each one of the used TC's.
- Maximum bandwidth for the PF is declared as a shared rate limiter.

The 10 GbE controller provides a hierarchical bandwidth limit. Configuring a bandwidth limit to the switching component, software automatically limits the total bandwidth available for all VSI allocated for that switching component. See Section 38.25.1.3.

This is a Direct Admin Queue command with completion reported within the completion descriptor structure. Table 38-311 lists the command format and defines command specific fields.

Table 38-311.Configure Switching Component Bandwidth Limit Command Fields

Name	Bytes.Bits	Value	Remarks
Flags	1:0	0	
Opcode	2-3	0x0410	Command opcode.
Datalen	4-5		Length of response buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Switching Component SEID	16-17		Unique identifier of the switching component.
Reserved1	18-19	0	Reserved. Must be set to 0.
Bandwidth Limit Credits	20-21		Bandwidth limit in Mb/s. Supported range is 50 Mb/s - 40 Gb/s, in increments of 50 Mb/s. 0 indicates that bandwidth limit is disabled. One credit corresponds to bandwidth limit of 50Mb/s.
Bandwidth Limit Max	24.0-24.2		Maximum bandwidth limit indicates how much bandwidth limit credits can be accumulated due to inactivity. Single scheduling Quanta consumes multiple bandwidth limit credits. The 10 GbE controller enables accumulating of following discrete values of Quanta-worth credits: 0, 1, 2, and 4 Quanta worth credits.
Reserved2	24.3-31	0	Reserved. Must be set to 0.

38.25.4.10 Enable, Disable, or Modify Physical Port ETS

This command allows software to enable, disable or modify ETS configuration of the specified switching component connected directly to the physical port. Configuring ETS for the switching component effectively results in configuring ETS for the switching component egress port.

This command can be used only for switching components directly connected to the physical port configured to ETS-based scheduler configuration. Otherwise, the command is ignored and an error is reported.

Software is expected to store a current ETS configuration of the switching component and fully specify the ETS configuration every time it requests its modification. Software cannot provide incremental changes to the ETS configuration.

Firmware configures TC arbitration to be a weighted round robin arbitration scheme. Software can chose to specify a different arbitration scheme (strict priority or a combination of WRR and WSP), it also has an option to provide an infinite number of credits to strict priority TCs.

This is an Indirect Admin Queue command with additional command attributes and completion attributes are provided within the data buffer. Table 38-312 lists the command format and defines command specific fields.

Table 38-312.Configure Physical Port ETS Command Fields

Name	Bytes.Bits	Value	Remarks
Flags	1:0	0	
Opcode	2-3	0x0413 0x0414 0x0415	Command opcode. 0x0413 = Enable ETS. 0x0414 = Modify ETS. 0x0415 = Disable ETS.
Datalen	4-5		Length of response buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Physical Port SEID	16-17		Unique identifier of the physical port.
Reserved	18-23	0	Reserved. Must be set to 0.
Data Address High	24-17		Address of buffer.
Data Address Low	28-31		

Table 38-313 lists the format of the data buffer carrying additional command attributes. The majority of the fields in this table are valid for ETS enable and modify operations only. ETS disable operation should refer to the switching component SEID only.

Table 38-313.Configure Physical Port ETS Command Buffer

Category	Byte/Bit	Field	Description
Reserved1	0-3		Reserved to match VSI configuration format.
TC Valid	4.0-4.7	TC0-TC7 Valid	Valid bit per TC. Should list all TCs enabled for physical port. At list one TC must be enabled.
Reserved1	5.0-5.7		Reserved. Must be set to 0.
TC Strict Priority	6.0-6.7	TC0-TC7 Strict Priority Flag	Strict priority flag per TC. If set then TC should be arbitrated based on its priority. If software decides to configure one or more TCs to be a strict priority TC, then TC with higher TC number has a higher priority.
Reserved2	7.0-7.7		Reserved. Must be set to 0.
Reserved3	8-23	0	Reserved to match VSI configuration format.
TC Bandwidth Share Credits	24	TC0 Bandwidth Credits	Relative credits per TC. Valid range of credits is 1-127 credits, in increments of the single credit. 127 (indicates infinite credits if TC is configured to strict priority).
	25	TC1 Bandwidth Credits	
	26	TC2 Bandwidth Credits	
	27	TC3 Bandwidth Credits	
	28	TC4 Bandwidth Credits	
	29	TC5 Bandwidth Credits	
	30	TC6 Bandwidth Credits	
	31	TC7 Bandwidth Credits	
Reserved4	32-127		Reserved to match VSI configuration format.

38.25.4.11 Suspend Port's Tx Traffic

This command enables a PF to suspend a port's transmit traffic. Note that this command is completes immediately. After all queue sets belonging to the port are suspended and the TX pipe of the port is drained, EMP firmware uses a Send Link Status Event notification to update the PF. During the time period between processing this command and getting the event notification about port draining, the AQ commands Set PHY Config, MAC Config, and Set Link and Restart AN for this port, are not allowed and are rejected with an error code EAGAIN (8).

Prior to calling this Admin command, a PF must use Set Event Mask to register itself to link events notifications.

This command enables software to modify a port's configuration like DCB setting when it owns the LLDP agent.

Port draining is part of DCB configuration flow. This command is used to verify that the Tx pipe is drained from any Tx packet belong to the port. This is a pre-condition to some configuration changes of the port TC or DCB. This command is valid only under SFP mode. See Section 38.25.5.6.1 for more detail.

This is a Direct Admin Queue command with completion reported within the direct command completion structure. Table 38-314 lists the command format and defines its specific fields.

Table 38-314.Suspend Port's Tx Traffic

Name	Bytes.Bits	Value	Remarks
Flags	1:0	0	
Opcode	2-3	0x041B	Command opcode.
Datalen	4-5	0	Length of response buffer.
Return Value	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Physical Port SEID	16-17		Unique identifier of the physical port.
Reserved	18-31	0	Reserved. Must be set to 0.

38.25.4.12 Resume PF Traffic

This command is used to resume suspended Tx traffic for the PF. It resumes all queue sets belonging to the PF (all its VEBs and VSIs for all their TCs). This command is called by the PF driver after the completion of a configuration flow, which requires a port's Tx pipe to be suspended. See Section 38.25.5.6.1 for more detail.

This is a Direct Admin Queue command with completion reported within the direct command completion structure. Table 38-315 lists the command format and defines its specific fields.

Table 38-315.Resume PF Traffic

Name	Bytes.Bits	Value	Remarks
Flags	1:0	0	
Opcode	2-3	0x041C	Command opcode.
Datalen	4-5	0	Length of response buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Reserved	16-31	0	Reserved. Must be set to 0.

38.25.4.13 Configure Switching Component Bandwidth Limit per Traffic Type

This command enables software to specify TCs enabled for a switching component and configures bandwidth limits of TCs enabled for the specified switch component.

Software should keep track of already configured bandwidth limits and provide complete bandwidth limit configuration, and not just incremental modifications.

This command can be used only for switching components belonging to the physical port configured to ETS-based scheduler configuration. Otherwise, the command is ignored and an error is reported.

Software is allowed to change bandwidth limit per traffic type and TCs enabled for a switching component that has a bandwidth allocation configured. New TCs enabled for a switching component has a default bandwidth allocated. Software should use a configure switching component bandwidth allocation per VSI AQ command to modify a default bandwidth allocation.

This command can be used to enable bandwidth limit per TC for the physical port.

Software cannot have both bandwidth limit and bandwidth limit per traffic type enabled for the same switching component. A request to enable bandwidth limit per traffic type for the switching component that has bandwidth limit enabled should fails and an EPERM error is reported in completion.

Software can configure either the switching component bandwidth limit or switching component TC bandwidth limit.

In that case, the PF uses more than one TC (such as an iSCSI PF that must be opened for Storage TC and LAN TC as well).

- PF is allowed to enable more than one TC. According enabled UP's and UP to TC mapping.

- Bandwidth relative allocation defined for this PF in the Alt RAM is configured for each one of the used TC's.
- Maximum bandwidth for the PF is declared as a shared rate limiter.

If TC re-configuration is needed while the VSI is suspended then new generated TC nodes of this VSI is also suspended.

This is an Indirect Admin Queue command. Software should provide a valid buffer carrying additional command attributes as listed in Table 38-316. Command completion does not provide any additional information beyond status, and does not use a data buffer provided by software for command attributes.

Table 38-316 lists the command format and defines command specific fields.

Table 38-316.Configure Switching Component Bandwidth Limits per Traffic Type Command Fields

Name	Bytes.Bits	Value	Remarks
Flags	1:0	0	
Opcode	2-3	0x0416	Command opcode.
Datalen	4-5		Length of response buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Switching Component SEID	16-17		Unique identifier of the switching component or physical port.
Reserved	18-23	0	Reserved. Must be set to 0.
Data Address High	24-17		Address of buffer.
Data Address Low	28-31		

Table 38-317 lists the format of the data buffer carrying additional command attributes.

Table 38-317.Configure Switching Component Bandwidth Limit per Traffic Type Command Buffer (Sheet 1 of 2)

Category	Byte/Bit	Field	Description
TC Valid	0.0-0.7	TC0-TC7 Valid	Valid bit per TC. Should list all TCs enabled for a switching component. Must be consistent with the physical port configuration. At least one TC must be enabled.
Reserved0	1-15	0	Reserved. Must be set to 0.
TC Bandwidth Limit Credits	16-17	TC0 Bw Limit Credits	This field specifies a bandwidth limit per TC in Mb/s. Supported bandwidth limit range is 50 Mb/s-40 Gb/s, with increments of 50 Mb/s. Bandwidth limit of 0 Mb/s indicates that bandwidth limit is disabled. Bandwidth limit should be provided for the valid TC handles only. One credit corresponds to bandwidth limit of 50 Mb/s.
	18-19	TC1 Bw Limit Credits	
	20-21	TC2 Bw Limit Credits	
	22-23	TC3 Bw Limit Credits	
	24-25	TC4 Bw Limit Credits	
	26-27	TC5 Bw Limit Credits	
	28-29	TC6 Bw Limit Credits	
	30-31	TC7 Bw Limit Credits	

Table 38-317.Configure Switching Component Bandwidth Limit per Traffic Type Command Buffer (Sheet 2 of 2)

Category	Byte/Bit	Field	Description
TC Max Bandwidth Limit	32.0-32.2	TC0 Max Bw Limit	Maximum bandwidth limit indicates how much bandwidth limit credits can be accumulated due to inactivity. Single scheduling Quanta consumes multiple bandwidth limit credits. The 10 GbE controller allows accumulating of following discrete values of Quanta-worth credits: 0,1, 2, 4, 8,16 32 and 64 Quanta worth credits.
	32.3	Reserved	
	32.4-32.6	TC1 Max Bw Limit	
	32.7	Reserved	
	33.0-33.2	TC2 Max Bw Limit	
	33.3	Reserved	
	33.4-33.6	TC3 Max Bw Limit	
	33.7	Reserved	
	34.0-34.2	TC4 Max Bw Limit	
	34.3	Reserved	
	34.4-34.6	TC5 Max Bw Limit	
	34.7	Reserved	
	35.0-35.2	TC6 Max Bw Limit	
	35.3	Reserved	
	35.4-35.6	TC7 Max Bw Limit	
	35.7	Reserved	
Reserved2	36-63	0	Reserved. Must be set to 0.

38.25.4.14 Configure Switching Component Bandwidth Allocation per Traffic Type

This command enables software to specify TCs enabled for a switching component and configure relative bandwidth allocation of switching components within each TC.

This command can be used for switching components configured in ETS-based mode only. Otherwise, the command is ignored and an error is reported.

To configure switching component bandwidth allocation per traffic type, software should provide complete relative bandwidth allocation for all TCs enabled for the switching component.

Software is allowed to change bandwidth allocation per traffic type and TCs enabled for a switching component that has a switching component bandwidth limit enabled. It is responsibility of firmware to adjust hardware configuration to reflect this change.

In that case, the PF uses more than one TC (such as an iSCSI PF that must be opened for Storage TC and LAN TC as well).

- PF is allowed to enable more than one TC. According enabled UP's and UP to TC mapping.
- Bandwidth relative allocation defined for this PF in the Alt RAM is configured for each one of the used TC's.
- Maximum bandwidth for the PF is declared as a shared rate limiter.

If TC re-configuration is needed while the VSI is suspended then new generated TC nodes of this VSI is also suspended.

This is an Indirect Admin Queue command. Software should provide a buffer that would be used both to provide additional command attributes and for the command completion.

Table 38-318 lists the command format and defines a command specific fields.

Table 38-318.Configure Switching Component Bandwidth Allocation per Traffic Type Command Fields

Name	Bytes.Bits	Value	Remarks
Flags	1:0	0	
Opcode	2-3	0x0417	Command opcode.
Datalen	4-5		Length of response buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Switching Component SEID	16-17		Unique identifier of the VSI.
Reserved1	18-23	0	Reserved. Must be set to 0.
Data Address High	24-27		Address of buffer. This buffer is used both to convey configuration to firmware, and provide software with a list of TC handles.
Data Address Low	28-31		

Table 38-319 lists the format of the command buffer and defines command attributes.

Table 38-319.Configure Switching Component Bandwidth Allocation per Traffic Type Command Buffer

Category	Byte/Bit	Field	Description
TC Valid	0.0-0.7	TC0-TC7 Valid	Valid bit per TC. Should list all TCs enabled for a switching component. Must be consistent with the physical port configuration. At least one TC must be enabled.
Reserved2	1-3.6	0	Reserved. Must be set to 0.
Absolute Credits	3.7	Absolute Credits Enable	If set indicates that credits provided are absolute credits, and not relative to the bandwidth allocated to other switching components on the same TC. Software should avoid use of absolute credits.
TC Bandwidth Share Credits	4	TC0 Bandwidth Credits	Relative switching component credits within same TC with respect to other switching components or VSIs enabled and connected to s-channels on the same physical port. Valid range of credits is 1-127 credits, in increments of the single credit. 127 (indicates infinite credits).
	5	TC1 Bandwidth Credits	
	6	TC2 Bandwidth Credits	
	7	TC3 Bandwidth Credits	
	8	TC4 Bandwidth Credits	
	9	TC5 Bandwidth Credits	
	10	TC6 Bandwidth Credits	
	11	TC7 Bandwidth Credits	
Reserved4	12-32	0	Reserved. Must be set to 0.

38.25.4.15 Query VSI Bandwidth Configuration

This command enables software retrieve current bandwidth configuration of the specified VSI of the specified switching component.

This is an Indirect Admin Queue command. Software should provide a buffer for command completion. Command completion carries VSI bandwidth configuration attributes, listed in Table 38-321.

DCB flow might require VEB configuration changing followed by VSI configuration changing. Calling to query VSI bandwidth configuration between the previous two steps is responded with an error code EBUSY (12).

Table 38-320 lists the command format and defines a command specific fields.

Table 38-320.Query VSI Bandwidth Configuration Command Fields

Name	Bytes.Bits	Value	Remarks
Flags	1:0	0	
Opcode	2-3	0x0408	Command opcode.
Datalen	4-5		Length of response buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware. This should carry a status and an indication whether ETS is enabled or not.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
VSI SEID	16-17		Unique identifier of the VSI.
Reserved	18-23	0	Reserved. Must be set to 0.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

Table 38-321 lists the format of the completion buffer and defines completion attributes. In addition to the bandwidth characteristics of VSI, this completion carries UP handles and QS handles of UPs and queue sets allocated for VSI.

Table 38-321.Query VSI Configuration Completion Buffer (Sheet 1 of 2)

Category	Byte/Bit	Field	Description
TC Valid	0.0-0.7	TC0-TC7 Valid	Valid bit per TC.
TC Suspended	1.0-1.7	TC0-TC7 Suspended	Marks per TC if this TC transmission is suspended.
Reserved0	2-15	0	Reserved. Must be set to 0.
QueueSet Handles	16-17	QS0 Handle	Queue set handles of configured queue sets. Invalid handles carries the value of 0xfFFFF Order of queue set handles matches order of TCs (0-7).
	18-19	QS1 Handle	
	20-21	QS2 Handle	
	22-23	QS3 Handle	
	24-25	QS4 Handle	
	26-27	QS5 Handle	
	28-29	QS6 Handle	
	30-31	QS7 Handle	
Reserved1	32-35	0	Reserved. Must be set to 0.

Table 38-321.Query VSI Configuration Completion Buffer (Sheet 2 of 2)

Category	Byte/Bit	Field	Description
Bandwidth Limit	36-37		Bandwidth limit configured for the VSI in Mb/s. 0 indicates that bandwidth limit was not configured for the VSI. One credit corresponds to bandwidth limit of 50 Mb/s.
Reserved2	38-39	0	Reserved. Must be set to 0.
Bandwidth Limit Max	40.0-40.2		Maximum bandwidth limit indicates how much bandwidth limit credits can be accumulated due to inactivity. Single scheduling Quanta consumes multiple bandwidth limit credits. The 10 GbE controller allows accumulating of following discrete values of Quanta-worth credits: 0,1, 2, and 4 Quanta worth credits.
Reserved3	40.3-63	0	Reserved. Must be set to 0.

38.25.4.16 Query VSI Bandwidth Configuration per Traffic Type

This command enables software to retrieve a bandwidth configuration of the specified VSI within each TC.

This command can be used for VSI configured in ETS-based mode only. Otherwise, the command is ignored and an error is reported.

If the physical port ETS is configured to single level ETS, and defines bandwidth distribution between TCs only. Software can assume one-to-one mapping of UPs-to-TCs from the bandwidth allocation perspective, and use this command to retrieve a bandwidth allocation for VSI within each TC.

This is an Indirect Admin Queue command. Software should provide a buffer for additional command attributes and command completion. Command completion carries VSI ETS/SLA configuration attributes, listed in Table 38-322.

DCB flow might require VEB configuration changing followed by VSI configuration changing. Calling to query VSI bandwidth configuration between the previous two steps is responded with error code EBUSY (12).

Table 38-322 lists the command format and defines command specific fields.

Table 38-322.Query VSI Bandwidth Allocation per Traffic Type Command Fields

Name	Bytes.Bits	Value	Remarks
Flags	1:0	0	
Opcode	2-3	0x040A	Command opcode.
Datalen	4-5		Length of response buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware. This should carry a status and an indication whether ETS is enabled or not.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
VSI SEID	16-17		Unique identifier of the VSI.
Reserved	18-23	0	Reserved. Must be set to 0.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

Table 38-323 lists the format of completion buffer and its attributes.

Table 38-323.Query VSI Bandwidth Allocation per Traffic Type Completion Buffer

Category	Byte/Bit	Field	Description
TC Valid	0.0-0.7	TC0-TC7 Valid	Valid bit per TC.
TC Suspended	1.0-1.7	TC0-TC7 Suspended	Marks per TC if this TC transmission is suspended.
Reserved1	2-3	0	Reserved. Must be set to 0.
TC Bandwidth Share Credits	4	TC0 Bandwidth Credits	Relative VSI credits within same TC with respect to other VSIs enabled on the same switching component Valid range of credits is 1-127 credits, in increments of the single credit. 127 (indicates infinite credits).
	5	TC1 Bandwidth Credits	
	6	TC2 Bandwidth Credits	
	7	TC3 Bandwidth Credits	
	8	TC4 Bandwidth Credits	
	9	TC5 Bandwidth Credits	
	10	TC6 Bandwidth Credits	
	11	TC7 Bandwidth Credits	
TC Bandwidth Limit Credits	12-13	TC0 Bandwidth Limit Credits	This fiend specifies a bandwidth limit per TC in Mb/s. Supported bandwidth limit range is 50 Mb/s-40 Gb/s, with increments of 50 Mb/s. Bandwidth limit of 0 Mb/s indicates that bandwidth limit is disabled. Bandwidth limit should be provided for the valid handles only. One credit corresponds to bandwidth limit of 50 Mb/s.
	14-15	TC1 Bandwidth Limit Credits	
	16-17	TC2 Bandwidth Limit Credits	
	18-19	TC3 Bandwidth Limit Credits	
	20-21	TC4 Bandwidth Limit Credits	
	22-23	TC5 Bandwidth Limit Credits	
	24-25	TC6 Bandwidth Limit Credits	
	26-27	TC7 Bandwidth Limit Credits	
TC Max Bandwidth Limit	28.0-28.2	TC0 Max Bandwidth Limit	Maximum bandwidth limit indicates how much bandwidth limit credits can be accumulated due to inactivity. Single scheduling Quanta consumes multiple bandwidth limit credits. The 10 GbE controller allows accumulating of following discrete values of Quanta-worth credits: 0,1, 2, 4, 8,16 32 and 64 Quanta worth credits.
	28.3	Reserved	
	28.4-28.6	TC1 Max Bandwidth Limit	
	28.7	Reserved	
	29.0-29.2	TC2 Max Bandwidth Limit	
	29.3	Reserved	
	29.4-29.6	TC3 Max Bandwidth Limit	
	29.7	Reserved	
	30.0-30.2	TC4 Max Bandwidth Limit	
	30.3	Reserved	
	30.4-30.6	TC5 Max Bandwidth Limit	
	30.7	Reserved	
	31.0-31.2	TC6 Max Bandwidth Limit	
	31.3	Reserved	
	31.4-31.6	TC7 Max Bandwidth Limit	
	31.7	Reserved	

38.25.4.17 Query Switching Component Configuration

This command enables software retrieve current bandwidth management configuration of the specified switching component.

This is an Indirect Admin Queue command. Software should provide a buffer for command completion. Command completion carries switching component bandwidth configuration attributes, listed in Table 38-325.

Table 38-324 lists the command format, and defines a command specific fields.

Result of this command is returned as a completion to the Admin Queue command.

Table 38-324.Query Switching Component Configuration Command Fields

Name	Bytes.Bits	Value	Remarks
Flags	1:0	0	
Opcode	2-3	0x0418	Command opcode.
Datalen	4-5		Length of response buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware. This should carry a status and an indication whether ETS is enabled or not.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Switching Component SEID	16-17		Unique identifier of the switching component.
Reserved	18-23	0	Reserved. Must be set to 0.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

Table 38-325 lists the format of completion buffer and its attributes.

Table 38-325.Query Switching Component Configuration Completion Buffer

Category	Byte/Bit	Field	Description
TC Valid	0.0-0.7	TC0-TC7 Valid	Valid bit per TC.
Reserved1	1-31	0	Reserved. Must be set to 0.
Reserved1	32-35	0	Reserved. Must be set to 0.
Bandwidth Limit	36-37		Bandwidth limit configured for the port in Mb/s. 0 indicates that bandwidth limit was not configured for the switching component. One credit corresponds to bandwidth limit of 50 Mb/s.
Reserved2	38-39	0	Reserved. Must be set to 0.
Bandwidth Limit Max	40.0-40.2		Max bandwidth limit indicates how much bandwidth limit credits can be accumulated due to inactivity. Single scheduling Quanta consumes multiple bandwidth limit credits. The 10 GbE controller allows accumulating of following discrete values of Quanta-worth credits: 0,1, 2, and 4 Quanta worth credits.
Reserved3	40.3-63	0	Reserved. Must be set to 0.

38.25.4.18 Query Physical Port ETS Configuration

This command enables software to retrieve ETS configuration of the specified switching component or VSI directly connected to the physical port.

This is an Indirect Admin Queue command. Software should provide a buffer for additional command attributes and command completion. Command completion carries switching component ETS/SLA configuration attributes, listed in Table 38-327.

This command can be used only for switching components directly connected to the physical port configured to ETS-based scheduler configuration. Otherwise, the command is ignored and an error is reported.

Table 38-326 lists the command format and defines command specific fields.

Table 38-326.Query Physical Port ETS Configuration Command Fields

Name	Bytes.Bits	Value	Remarks
Flags	1:0	0	
Opcode	2-3	0x0419	Command opcode.
Datalen	4-5		Length of response buffer.
Return Vvalue/ VFID	6-7		Return value. Zeroed by driver. Written by firmware. This should carry a status and an indication whether ETS is enabled or not.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
Physical Port SEID	16-17		Unique identifier of the physical port.
Reserved	18-23	0	Reserved. Must be set to 0.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

Table 38-327 lists the format of completion buffer and its attributes.

Table 38-327.Query Physical Port ETS/SLA Completion Buffer (Sheet 1 of 2)

Category	Byte/Bit	Field	Description
Reserved1	0-3		Reserved to match VSI configuration format.
TC Valid	4.0-4.7	TC0-TC7 Valid	Valid bit per TC. This field is valid for ETS enable operation only. For ETS modify operation TC handles must be used instead. Up to eight TCs can be valid at any time.
TC Strict Priority	6.0-6.7	TC0-TC7 Strict Priority Flag	Strict priority flag per TC. If set then TC should be arbitrated based on its priority. If software decides to configure one or more TCs to be a strict priority TC, then TC with higher TC number has a higher priority. Configuring TCs to the strict priority may be useful for SLA configuration.

Table 38-327.Query Physical Port ETS/SLA Completion Buffer (Sheet 2 of 2)

Category	Byte/Bit	Field	Description
TC Bandwidth Share Credits	8	TC0 Bandwidth Credits	Relative credits per TC. Valid range of credits is 1-127 credits, in increments of the single credit. 127 (indicates infinite credits if TC is configured to strict priority).
	9	TC1 Bandwidth Credits	
	10	TC2 Bandwidth Credits	
	11	TC3 Bandwidth Credits	
	12	TC4 Bandwidth Credits	
	13	TC5 Bandwidth Credits	
	14	TC6 Bandwidth Credits	
	15	TC7 Bandwidth Credits	
TC Bandwidth Limit Credits	16-17	TC0 Bandwidth Limit Credits	This field specifies a bandwidth limit per TC in Mb/s. Supported bandwidth limit range is 50 Mb/s-40 Gb/s, with increments of 50 Mb/s. Bandwidth limit of 0 Mb/s indicates that bandwidth limit is disabled. Bandwidth limit should be provided for the valid TC handles only. One credit corresponds to bandwidth limit of 50 Mb/s.
	18-19	TC1 Bandwidth Limit Credits	
	20-21	TC2 Bandwidth Limit Credits	
	22-23	TC3 Bandwidth Limit Credits	
	24-25	TC4 Bandwidth Limit Credits	
	26-27	TC5 Bandwidth Limit Credits	
	28-29	TC6 Bandwidth Limit Credits	
	30-31	TC7 Bandwidth Limit Credits	
TC Max Bandwidth Limit	32.0-32.2	TC0 Max Bandwidth Limit	Maximum bandwidth limit indicates how much bandwidth limit credits can be accumulated due to inactivity. Single scheduling Quanta consumes multiple bandwidth limit credits. The 10 GbE controller enables accumulating of following discrete values of Quanta-worth credits: 0,1, 2, 4, 8,16 32 and 64 Quanta worth credits.
	32.3	Reserved	
	32.4-32.6	TC1 Max Bandwidth Limit	
	32.7	Reserved	
	33.0-33.2	TC2 Max Bandwidth Limit	
	33.3	Reserved	
	33.4-33.6	TC3 Max Bandwidth Limit	
	33.7	Reserved	
	34.0-34.2	TC4 Max Bandwidth Limit	
	34.3	Reserved	
	34.4-34.6	TC5 Max Bandwidth Limit	
	34.7	Reserved	
	35.0-35.2	TC6 Max Bandwidth Limit	
	35.3	Reserved	
	35.4-35.6	TC7 Max Bandwidth Limit	
	35.7	Reserved	
Reserved3	36-67	0	Reserved to match VSI configuration format. Must be set to 0.

38.25.4.19 Query Switching Component Bandwidth Configuration per Traffic Type

This command enables software to retrieve a bandwidth configuration of the specified switching component within each TC.

This command can be used for a switching component configured in ETS-based mode only. Otherwise, the command is ignored and an error is reported.

This is an Indirect Admin Queue command. Software should provide a buffer for additional command attributes and command completion. Command completion carries VSI ETS/SLA configuration attributes, listed in Table 38-328.

Table 38-328 lists the command format, and defines command specific fields.

Table 38-328.Query Switching Component Bandwidth Allocation per Traffic Type Command Fields

Name	Bytes.Bits	Value	Remarks
Flags	1:0	0	
Opcode	2-3	0x041A	Command opcode.
Datalen	4-5		Length of response buffer.
Return Value/ VFID	6-7		Return value. Zeroed by driver. Written by firmware. This should carry a status and an indication whether ETS is enabled or not.
Cookie High	8-11	Cookie	Opaque value, is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value, is copied by firmware into the completion of this command.
SEID	16-17		Unique identifier of the VSI or switching component.
Reserved	18-23	0	Reserved. Must be set to 0.
Data Address High	24-27		Address of buffer.
Data Address Low	28-31		

Table 38-329 lists the format of completion buffer and its attributes.

Table 38-329.Query Switching Component Bandwidth Allocation per Traffic Type Completion Buffer (Sheet 1 of 2)

Category	Byte/Bit	Field	Description
TC Valid	0.0-0.7	TC0-TC7 Valid	Valid bit per TC
Reserved1	1-3.6	0	Reserved. Must be set to 0.
Absolute Credits	3.7	Absolute Credits Enable	If set indicates that credits provided are absolute credits, and not relative to the bandwidth allocated to other switching components on the same TC. Software should avoid use of absolute credits.
TC Bandwidth Share Credits	4	TC0 Bandwidth Credits	Relative switching component credits within same TC with respect to other switching components enabled on the same physical port. Valid range of credits is 1-127 credits, in increments of the single credit. 127 (indicates infinite credits).
	5	TC1 Bandwidth Credits	
	6	TC2 Bandwidth Credits	
	7	TC3 Bandwidth Credits	
	8	TC4 Bandwidth Credits	
	9	TC5 Bandwidth Credits	
	10	TC6 Bandwidth Credits	
	11	TC7 Bandwidth Credits	

Table 38-329.Query Switching Component Bandwidth Allocation per Traffic Type Completion Buffer (Sheet 2 of 2)

Category	Byte/Bit	Field	Description
TC Bandwidth Limit Credits	12-13	TC0 Bandwidth Limit Credits	This fiend specifies a bandwidth limit per TC in Mb/s. Supported bandwidth limit range is 50 Mb/s-40 Gb/s, with increments of 5 0Mb/s. Bandwidth limit of 0 Mb/s indicates that bandwidth limit is disabled. Bandwidth limit should be provided for the valid handles only. One credit corresponds to bandwidth limit of 50 Mb/s.
	14-15	TC1 Bandwidth Limit Credits	
	16-17	TC2 Bandwidth Limit Credits	
	18-19	TC3 Bandwidth Limit Credits	
	20-21	TC4 Bandwidth Limit Credits	
	22-23	TC5 Bandwidth Limit Credits	
	24-25	TC6 Bandwidth Limit Credits	
	26-27	TC7 Bandwidth Limit Credits	
TC Max Bandwidth Limit	28.0-28.2	TC0 Max Bandwidth Limit	Max bandwidth limit indicates how much bandwidth limit credits can be accumulated due to inactivity. Single scheduling Quanta consumes multiple bandwidth limit credits. The 10 GbE controller allows accumulating of following discrete values of Quanta-worth credits: 0,1, 2, 4, 8,16 32 and 64 Quanta worth credits.
	28.3	Reserved	
	28.4-28.6	TC1 Max Bandwidth Limit	
	28.7	Reserved	
	29.0-29.2	TC2 Max Bandwidth Limit	
	29.3	Reserved	
	29.4-29.6	TC3 Max Bandwidth Limit	
	29.7	Reserved	
	30.0-30.2	TC4 Max Bandwidth Limit	
	30.3	Reserved	
	30.4-30.6	TC5 Max Bandwidth Limit	
	30.7	Reserved	
	31.0-31.2	TC6 Max Bandwidth Limit	
	31.3	Reserved	
	31.4-31.6	TC7 Max Bandwidth Limit	
	31.7	Reserved	

38.25.5 Scheduler Configuration Flows

This section describes use of AQ commands defined in Section 38.25.4 to configure the scheduler. It is not intended to provide a complete coverage of all possible configuration flow, but rather cover a several main stream cases.

Table 38-330 lists all possible configuration changes of the internal switch or bandwidth configuration that have an impact on the transmit scheduler configuration, along with AQ commands that should be used.

Table 38-330.Scheduler Configuration Summary (Sheet 1 of 2)

Configuration Change	Description	AQ Command
Default Port ETS Configuration	Default ETS configuration is ETS disabled, and the only TC enabled is TC0. See Section 38.25.5.3.	NA
Disable Physical Port	Disabling previously configured physical port does not directly impact scheduler configuration. Firmware suspends the disabled port and software is responsible for resource deallocation using Delete Element AQ command.	NA
Change Physical Port ETS Configuration	Driven by agent implementing DCBX protocol. Agent can run in firmware or in software. In case of software AQ command should be used to perform required ETS configuration change. See Section 38.25.5.3.	Enable, Disable, Modify Physical Port ETS. See Section 38.25.4.10.
Add Default VSI	Default VSI is allocated at switch initialization time. SFP configuration - one default VSI per port. Default VSI is configured with default bandwidth configuration (even bandwidth allocation, no bandwidth limits), and with single TC enabled - TC0. See Section 38.25.5.1.	NA
Add VSI	Regular VSI can be added directly connected to the physical port, S-channel of PE, or VEB. Scheduler configuration is done as a part of the internal switch configuration. A new VSI is configured with a default bandwidth configuration (single bandwidth allocation credit, and no bandwidth limit enabled). A new VSI can be configured with an initial set of TCs enabled for VSI. TCs enabled for VSI must be a subset of TCs enabled to the parent switching element or physical port. See Section 38.25.5.3.	Add VSI.
Change TCs Enabled For Default VSI Or Regular VSI.	Default or regular VSI TC configuration can be changed using Configure VSI Bandwidth Allocation per Traffic Type or Configure VSI Bandwidth Limit per Traffic Type AQ commands. In either case TCs enabled for the default VSI must be a subset of TCs enabled for the corresponding physical port/PE. Update VSI CANNOT be used to change TC configuration of transmit scheduler.	Configure VSI Bandwidth Allocation per Traffic Type. See Section 38.25.4.8. Configure VSI Bandwidth Limit per Traffic Type. See Section 38.25.4.7.
Change VSI Bandwidth Configuration	To change VSI bandwidth configuration, software can use one of three AQ commands: Configure VSI Bandwidth Limit to enable/modify/disable bandwidth limit that applies to entire VSI (all enabled TCs) Configure VSI Bandwidth Limit per Traffic Type to enable/modify/disable bandwidth limit for individual TC enabled for VSI. Configure VSI Bandwidth Allocation per Traffic Type to modify bandwidth allocation for the individual TC enabled for VSI.	Configure VSI Bandwidth Limit See Section 38.25.4.5. Configure VSI Bandwidth Allocation per Traffic Type. See Section 38.25.4.8. Configure VSI Bandwidth Limit per Traffic Type. See Section 38.25.4.7.
Delete VSI	Software can remove VSI using Delete Element AQ command. This includes removing all TC nodes enabled for VSI.	Delete Element.
Add VEB to Default VSI	When VEB is added to the default VSI, it is effectively inserted between default VSI and physical port/PE. VEB can be configured with a set of TCs enabled for VEB. TCs enabled for VEB must include all TCs enabled for the default VSI. VEB inherits bandwidth configuration of default VSI. TCs that are enabled for VEB and not enabled for the default VSI must be configured with a default bandwidth configuration. See Section 38.25.5.2.	Add VEB/PA.

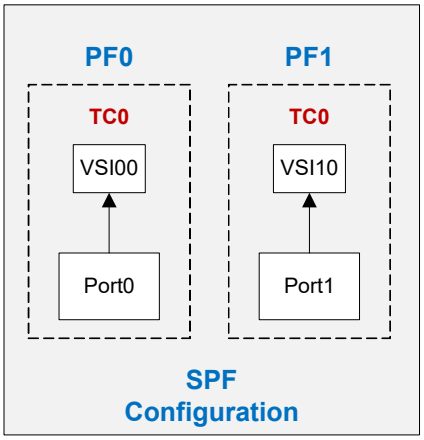
Table 38-330.Scheduler Configuration Summary (Sheet 2 of 2)

Configuration Change	Description	AQ Command
Add Floating VEB	Not supported by current scheduler definition. Transmit scheduler must provide a physical port and TC along with the scheduling request to enable proper device resource allocation by the pipeline. There are several ways how floating VEB can be enabled for the scheduler. All options result in modification of the Scheduler configuration tables.	Add VEB/PA.
Change TCs Enabled For VEB	To change number of TCs enabled for VEB software should use either Configure Switching Component Bandwidth Limit per Traffic Type, or Configure Bandwidth Allocation per Traffic Type AQ commands. TCs enabled for VEB must be a subset of TCs enabled for the parent switching component or physical port. Update VEB CANNOT be used to change TC configuration of transmit scheduler.	Configure Switching Component Bandwidth Limit per Traffic Type. See Section 38.25.4.14. Configure Switching Component Bandwidth Allocation per Traffic Type. See Section 38.25.4.13.
Change VEB Bandwidth Configuration.	To change VEB bandwidth configuration software can use one of three AQ commands: Configure Switching Component Bandwidth Limit to enable/ modify/disable bandwidth limit that applies to entire VEB (all enabled TCs). Configure Switching Component Bandwidth Limit per Traffic Type to enable/modify/disable bandwidth limit for individual TC enabled for VEB. Configure Switching Component Bandwidth Allocation per Traffic Type to modify bandwidth allocation for the individual TC enabled for VEB.	Configure Switching Component Bandwidth Limit. See Section 38.25.4.9. Configure Switching Component Bandwidth Limit per Traffic Type. See Section 38.25.4.14. Configure Switching Component Bandwidth Allocation per Traffic Type. See Section 38.25.4.13.
Delete VEB	Software can remove VEB using Delete Element AQ command. This includes removing all TC Nodes enabled for VEB. Software can remove VEB only after removing all VSIs attached to it after VEB creation. When VEB is removed, a default VSI is attached back to the S-channel or physical port.	Delete Element.

38.25.5.1 Default VSI

Default VSI is automatically created for each PF. In SFP environment single default VSI is allocated per physical port. Default VSI is always associated with TC0. Default VSI owns all bandwidth allocated for the physical port or a PF. Software can change TCs enabled for the default VSI and bandwidth allocation per Traffic Type using AQ command described in Section 38.25.4.8.

Figure 38-79.Default VSI Configuration



Firmware allocates a queue set for the each default VSI. Software can retrieve a queue set handle using Query VSI command. There is no additional configuration of default VSI is required, and once respective queue set handle is associated with transmit queue, that queue can be scheduled for transmission.

ETS is disabled for the default VSI. Software can enable ETS configuration for the physical port and then configure ETS for VSI. See Section 38.25.5.3.

Software can instantiate a switching component using Add VEB or Add PE commands, see Section 38.25.5.2.

38.25.5.2 Adding VEB/PA/Port Extender

Adding S-comp (port extender) does not impact transmit scheduler configuration. S-comp is effectively replacing a physical port in the transmit scheduler configuration hierarchy. All attributes configured for the port automatically apply for the S-comp (like ETS configuration, etc.).

VEB/PA added to the switching hierarchy can be inserted between a default VSI and a physical port or an S-comp, or can be instantiated using its own S-channel.

Regular VEB, attached to physical port, always inserted between port or port extender and VSI. If VSI is not a default VSI automatically allocated by firmware for each physical function or physical port, software is required to allocate VSI prior to adding VEB.

Figure 38-80.Adding VEB/PA

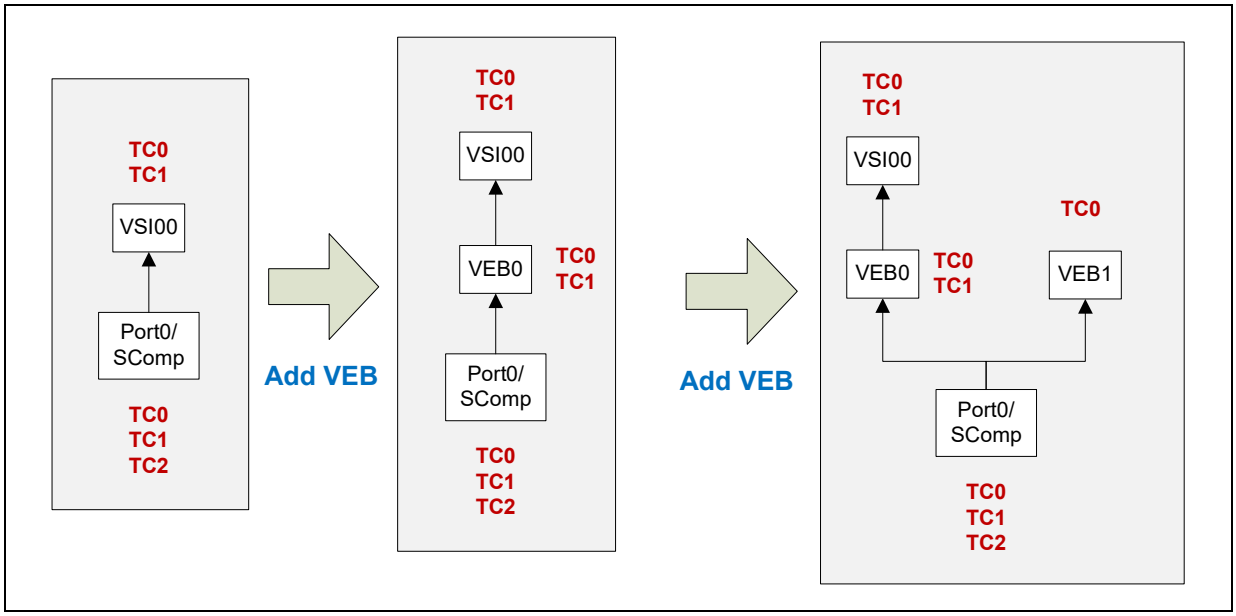


Figure 38-80 shows a transition of adding a VEB/PA switching component.

The first step shows insertion of the VEB between physical port and a default VSI (VSI00). In this example port has an ETS enabled, and a default VSI is configured with multiple TCs. Inserted VEB inherits VSIs configuration (such as TCs enabled for VSI, and bandwidth management attributes). VEB might have more TCs enabled than VSI. Software can change TCs enabled for VEB and relative bandwidth allocation within each Traffic Class using AQ command described in Section 38.25.4.14.

Insertion of VEB between default VSI and S-channel does not impact queue set allocated for VSI. Transmit queues, if any, previously associated with queue sets allocated for default VSI remains operational.

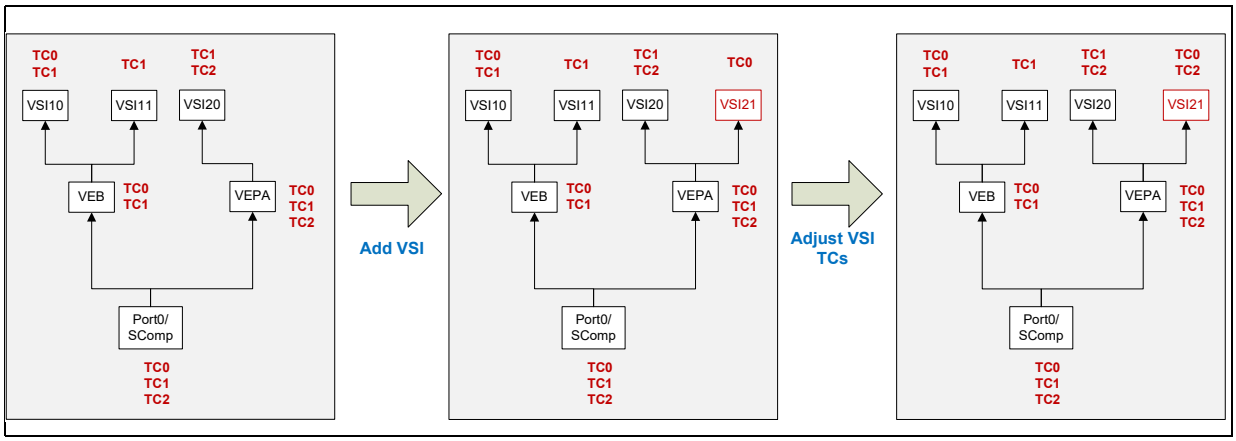
The second step shows a new VSI directly attached to the S-channel. Software is required to add VSI prior to adding a new VEB. In the example shown on the diagram, new VSI has TC0 enabled. Software can specify TCs enabled for VSI using a TC enable bitmap in Add VSI AQ Command, or can change it later using Configure VSI Bandwidth Allocation per Traffic Type, or Configure VSI Bandwidth Limit per Traffic Type AQ commands. Upon allocation of VSI, firmware allocates a queue set for each TC enabled for VSI, and provides it in the AQ command completion.

The third step shows a second VEB/PA inserted between VSI10 and the PE (VEB1). Process of creation and bandwidth configuration is similar to one previously described for the VEB0.

38.25.5.3 Adding VSI

In addition to the default VSI created for each port in SFP mode, software can request adding VSI to the previously allocated switching component. By default VSI is allocated with single TC enabled - TC0. Software can configure TCs enabled for VSI either using a bitmask in Add VSI AQ command, or using scheduler configuration AQ command described in Section 38.25.4.8.

Figure 38-81.Add VSI



Each allocated VSI is allocated a queue set per enabled TC. Queue set handles are returned in completion of Add VSI, or AQ command described in Section 38.25.4.8.

38.25.5.4 Bandwidth Limiting

Each allocated switching component and VSI might have a bandwidth limit configured. The 10 GbE controller supports two kinds of bandwidth limits: global, including traffic generated on all TCs enabled for VSI or switching component and per traffic type bandwidth limit. Global bandwidth limits can be enabled using AQ commands described in Section 38.25.4.9 and Section 38.25.4.5. Per TC bandwidth limits can be enabled using AQ commands described in Section 38.25.4.7.

Bandwidth limits are be default disabled. Software can enable only one kind of bandwidth limit to each switching component and VSI.

Software does not have to modify relative bandwidth allocation for the switching components and VSIs to enable bandwidth limits. The default (even) bandwidth allocation can be used, and bandwidth would be equally distributed between switching components and VSIs as long as they have not reached configured bandwidth limit.

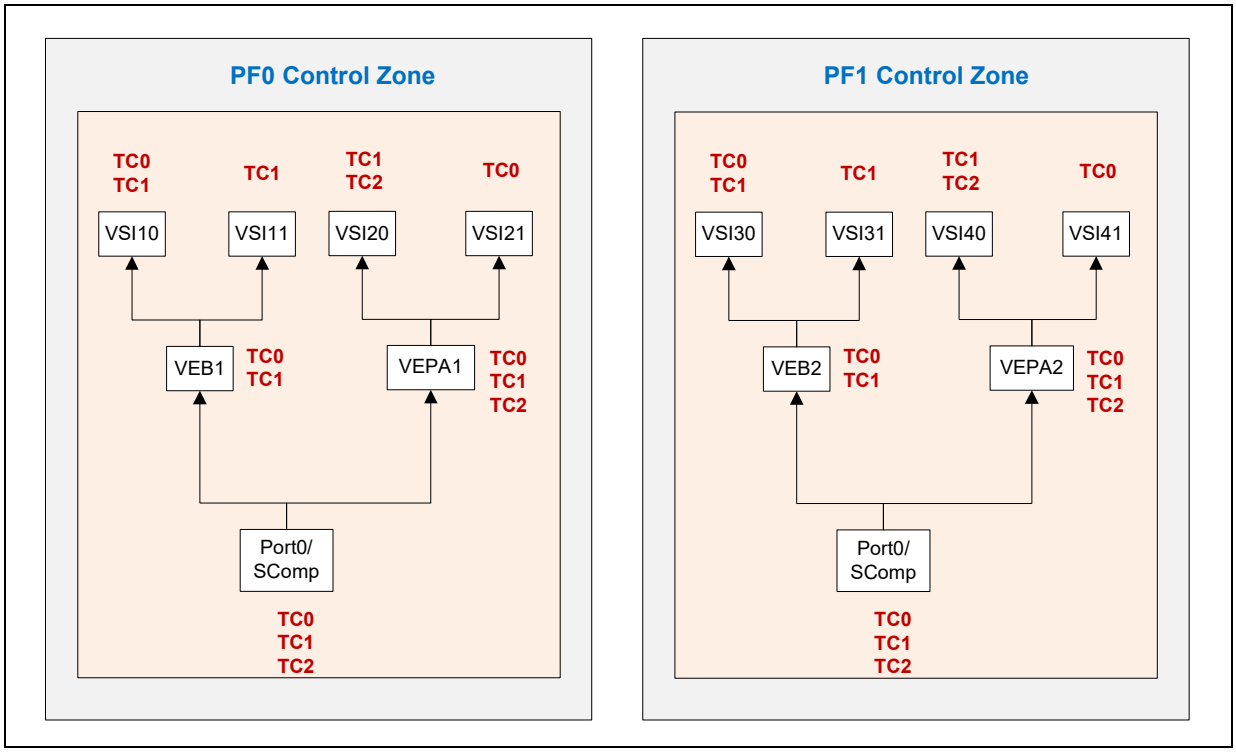
38.25.5.5 Bandwidth Distribution Ownership

38.25.5.5.1 SFP Mode

In SFP configuration one PF owns all resources allocated to the physical port.

ETS configuration of the physical port by default would be performed by firmware based on data provided by DCBX agent running in firmware. Software can take over the role of DCBX agent, and then it controls ETS configuration of the physical port as well. In either case, allocation or bandwidth management of the switching components and VSIs is completely owned by software, and performed using AQ commands described in Section 38.25.4.

Figure 38-82.Software Control Zone in SFP Mode



Bandwidth allocation should be done using relative credits only. The PF driver should have all information available to manage relative credits.

38.25.5.6 Enabling/Changing port DCB Configuration

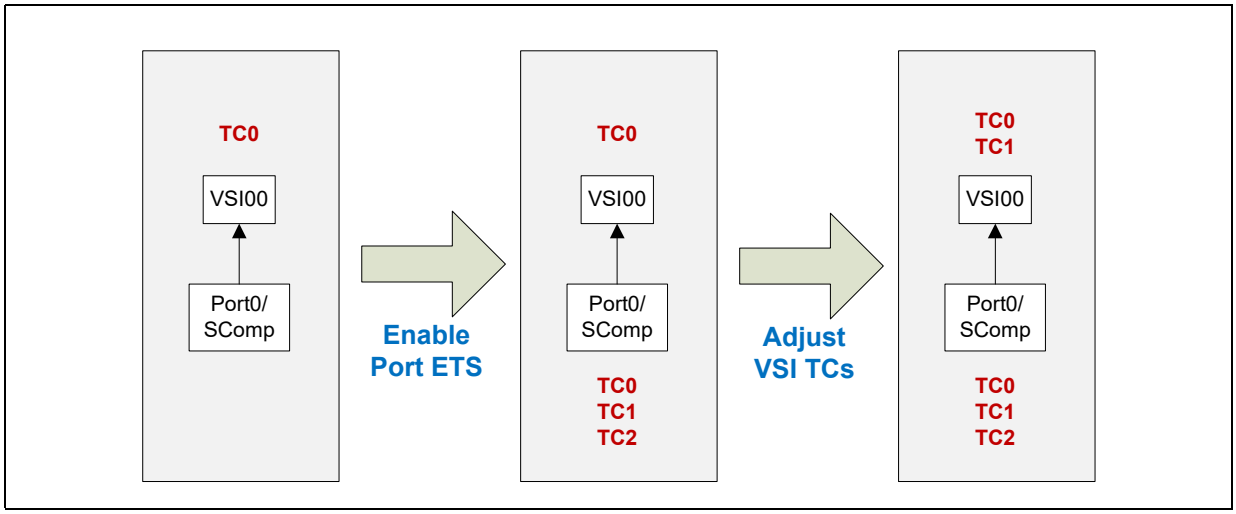
ETS configuration of a physical port can be enabled or modified at any point. This is not expected to be a frequent event, and therefore transition period to adjust scheduler configuration to reflect the change is acceptable.

At least one TC must be enabled. If DCB is disabled for the Port, TC0 should be used.

If port extender (S-comp) is enabled for the physical port, all attributes configured for the port automatically apply for the S-comp (like ETS configuration, etc.)

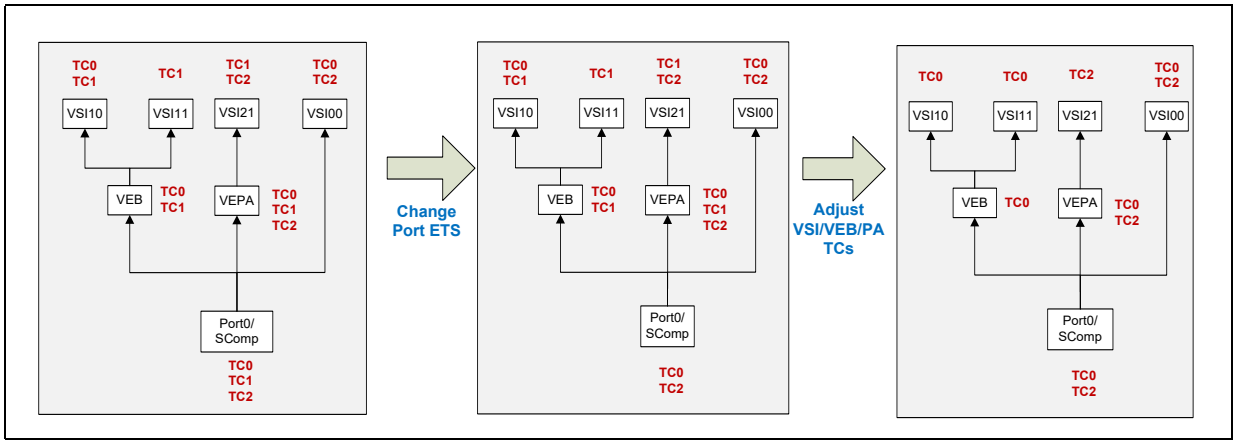
In the default flow, firmware negotiates ETS configuration of the physical port, and configures TCs enabled by ETS for the physical port. Default VSI, previously created for the port or PFs associated with the port, remains associated with TC0 only. Software should use AQ command described in Section 38.25.4.8 to modify TCs enabled for VSI and a relative bandwidth allocation of VSI within each enabled TCs.

Figure 38-83.Enable Physical Port ETS



If software instantiated additional VSIs or switching components prior to ETS is enabled for the port, or if ETS configuration has been changed, firmware is responsible to update ETS configuration of physical port, and add or remove TCs. All previously allocated VSIs and switching components remain associated with TCs previously configured TCs. Software should use AQ commands described in Section 38.25.4.8 and Section 38.25.4.14 to adjust TCs enabled for VSIs and switching components, and modify bandwidth allocation per traffic type as required.

Figure 38-84.Change Physical Port ETS



Scheduler starts scheduling requests for the new TCs added to the physical port configuration, only after this TC is enabled for one or more VSIs. Adding more TCs to VSI leads to allocation of new queue sets, which in turn must be associated with transmit queues.

Once TC is removed from the physical port configuration, transmit scheduler suspends scheduling traffic for all queue sets associated with that TC and then notifies software with an LLDP MIB change event. To resume scheduling traffic on transmit queues associated with such queue sets, software needs to adjust TC configuration of VSIs and switching components, and reassign affected transmit queues to queue sets associated with other TCs.

Port DCB configuration can be directly changed by firmware as a result of the DCBX exchange, this assumes a DCBX agent running in firmware (which is a default configuration). Or it can be configured by software using AQ command described in Section 38.25.4.10 if software took over duty of running DCBX agent. Transmit scheduler configuration does not depend on the location of the DCBX agent.

ETS reconfiguration of VSI might move Tx queues between TCs. Running this configuration flow while Tx pipe is loaded with Tx packets belong to this queue (either in the TCB or in the TPB) might cause out of order completion for this Tx queue.

To prevent this error case, it is required to verify that Tx pipe is drained from any Tx packet belonging to re-configured queue or queue set. There are two ways software can verify this draining.

1. PF can stop feeding the Tx ring and waits until all pending work completes.
2. PF can disable the Tx queue. This provides faster draining mechanism but flushes all pending work.

EMP firmware provides a service which suspends port's queue sets and drains the Tx pipe on the fly. This port draining service is used as part of DCBX change (see Section 38.25.4.11 for more detail).

The general flow DCBX exchange is: If ETS setting of VSI's is required to be changed then the EMP first drains the first the Tx pipe of the port (initiated by LLDP owner, done by EMP and Tx scheduler). After the port is drained, firmware notifies the PF that makes those required ETS changes (via Tx scheduler AQC). After VSI's ETS setting is adjusted, the PF resumes its queue sets using AQC described in Section 38.25.4.7 and in Section 38.25.4.8.

Some other DCB parameters require Tx pipe draining before re-configuration. DCBX owner (EMP firmware or software) uses the Tx scheduler draining service. More details are described in DCBX.

Some of DCBX flows require change of queue context configuration. In those cases, software must disable the queue first.

38.25.5.6.1 Tx Scheduler DCB Control Flows

General

DCBX and Tx scheduler control flows were declared under some assumptions and decisions as followed:

- For DCBX exchange event, software runs the similar flow under SFP.
- Usually DCBX agent runs in the EMP. Under SFP mode, software might take LLDP agent ownership.
- If software did not take LLDP ownership, the difference between DCBX software flows when running under SFP needs to be minimized.

- DCBX event might occur during first connectivity to the network as part of power up flow, as part of link down link up event, or in rare case, the switch had decided to change DCB setting.
- Three typical boot sequences are identified (DCBX runs at a different stage in each case):

Case I - Typical SFP boot sequence

a. EMP sets the initial scheduler configuration — One initial VSI per PF. Detailed flow in Reset/init flow.

b. DCBX runs. Detailed flow in DCBX.

c. Software boot. Detailed flow in Software Boot.

Case II - Late DCBX, DCBX parameters changed by the peer, or Link event.

a. EMP sets the initial scheduler configuration — One initial VSI per PF. Detailed flow in Reset/init flow.

b. Software boot. Detailed flow in Software Boot.

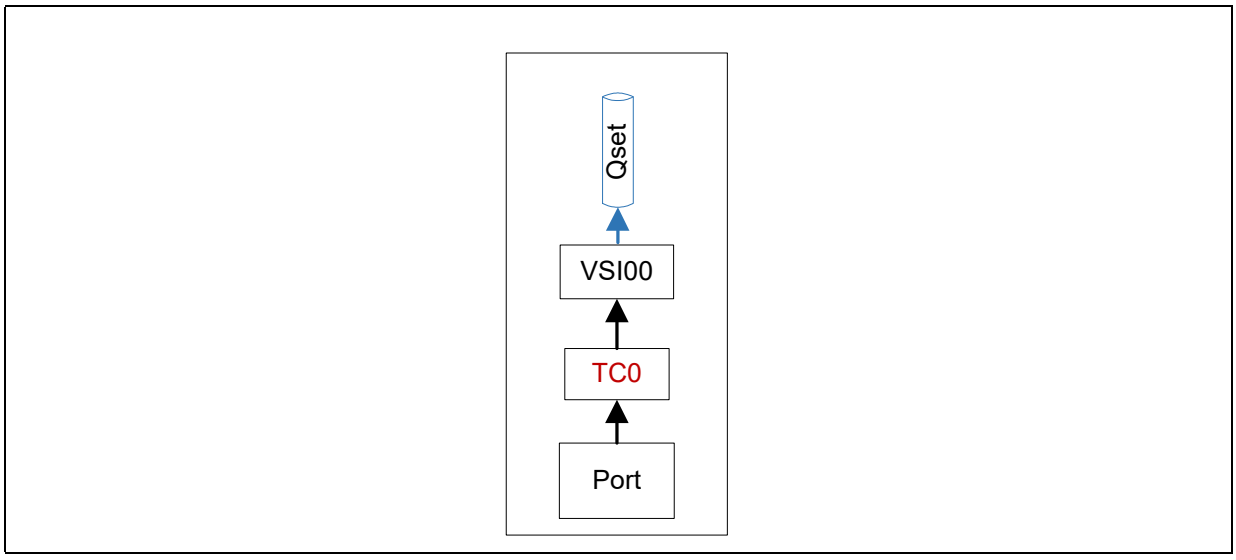
c. DCBX runs or makes changes in DCBX setting. Detailed flow in DCBX.

- When link goes up either while power up flow or while system already runs, all DCBX MIBs are reset to their default values (All TC are in DROP policy, all UPs are mapped to TC#0 and TC#0 is the only active TC for this port). So, after link up, the DCBX flow runs twice, a) Reset MIBs to default. b) New DCBX exchange with the new link partner.

Reset/init flow

- EMP initializes switch and Tx scheduler tables.
- For SFP, EMP builds one initial VSI per active port.
- EMP exposes the Qsets for pre-boot Tx (without active AQ)

Figure 38-85.initial Setting in SFP (One Initial VSI/Port)



DCBX exchange

- When EMP owns LLDP:

  - IF port draining is required (change in drop policy, application TLV, UP to TC mapping or port TC mapping)
    - EMP suspends all port's Tc (all TCs - no credits)
    - EMP drains port's Tx pipeline (waits until all pipe monitor counters are zeroed).
  - EMP re-configures needed DCB settings.
  - IF port is suspended then EMP resumes manageability Tx traffic.
  - EMP re-configures TX scheduler.
    - Port ETS setting (suspend the TC nodes of disabled TCs, resume TC nodes of enabled TCs).
    - TC bandwidth configuration.
  - EMP notifies DCBX event to all PFs that are registered for LLDP MIB notification.
    - This is done via posting LLDP MIB change event, after waking up the host if it was not in D0 state.
    - LLDP MIB change event notification includes *miscellaneous* field, which mark to the PF if the MIB change involved port draining and if per TC pipe flushing was used.
    - PF that is not registered for LLDP event must periodically check for LLDP MIBs update status.

Following is relevant only when software is up and running (case III):

- When software owns LLDP (applies only in SFP mode).
  - IF port draining is required, PF calls Suspend Port’s TX Traffic AQC (See Section 38.25.4.11).
    - EMP suspends all port's TX (all TCs - no credits).
    - EMP stops manageability Tx traffic.
    - EMP drains port's Tx pipeline (waits until all pipe monitor counters are zeroed).
    - EMP drains port's Tx pipeline (waits until all pipe monitor counters are zeroed).
  - PF re-configures needed DCB settings as needed.
  - PF instructs EMP to configure Tx scheduler with port's TC setting (via Configure Physical Port ETS AQC).
- PF computes the needed tree topology changes according the new UP to TC mapping and PFs' UP settings.
- PF marked as an iSCSI service, defined in Alt RAM: PF protocol (Offset 0xB), needs to be opened for both iSCSI TC and LAN TC.
- PF adjusts VEBs and VSIs per TC setting:
  - Done via Configure VSI Bandwidth Limit per Traffic Type, Configure VSI Bandwidth Allocation per Traffic Type, Configure Switching Element Bandwidth Limit per Traffic Type or Configure Switching Element Bandwidth Allocation per Traffic Type commands.
    - If the commands Configure VSI Bandwidth Limit per Traffic Type, Configure VSI Bandwidth Allocation per Traffic Type are called and remapped to only one TC node, EMP gives this TC node its original queue set handle. This enables the PF to skip the step of disabling and enabling all queues.
    - The commands Configure VSI Bandwidth Limit per Traffic Type, Configure VSI Bandwidth Allocation per Traffic Type can be used to expand VSI’s TC

setting. The EMP returns the queue set handle for each one of the enabled TCs including the new used TCs.
    - PF copies the new queue set handles to the contexts of the Tx queues belonging to the new established queue sets. When a queue context needs to change its queue set (moving between queue sets), software must disable the queue, reconfigure and enable back.
- After configuration completes, the PF resumes its suspended Tx traffic using AQC Resume PF Traffic (see Section 38.25.4.12).

Software Boot

- PF reads the DCBX MIBs (including UP-TC mapping).
  — In order to get consistent MIB data, the PF is required to verify that the LLDP owner is not processing LLDP messages right now by polling PRTDCB_GENS.DCBX_STATUS until it is set to DONE.
- Before making any change to VSI settings, the PF reads initial configuration and configures all queue sets.
  — Tree topology reading done via Get Switch Configuration command.
  — For each VSI, the PF needs to call Query VSI Bandwidth Configuration (see Section 38.25.4.15).
    - The response buffer includes:
    - A bitmap, which TCs are enabled in this VSI.
    - A queue set handle for each enabled TC.
    - A bitmap, which of the enabled TCs is suspended due to DCBX event.
    - PF must verify that a queue set is established for each enabled TC.
    - For each enabled TC, PF is required to copy the queue set handle value to all queues belonging to the queue set (before enabling the queue).
- PF determines whether to move/add any VSI to a different TC. Done via Configure VSI Bandwidth Limit per Traffic Type, Configure VSI Bandwidth Allocation per Traffic Type commands.
  — If the commands Configure VSI Bandwidth Limit per Traffic Type, Configure VSI Bandwidth Allocation per Traffic Type are called and moved to only one TC node, the EMP gives this TC node, its original queue set handle. This enables the PF to skip the step of disabling and enabling all queues.
    - PF resumes its suspended VSI's using AQC Resume PF Traffic (see Section 38.25.4.12).
- When the PF needs to change a TC setting of a VSI (not as a response to DCBX event, This VSI is NOT suspended), PF MUST drain all Tx queues belonging to moved TCs (in the VSI) before changing its TC settings. Changing any setting of a queue set while it has packets in the Tx pipe might cause out of order completion.
- Any topology change is done via the Add VSI or Add VEB commands (*Enabled TCs* field) that defines the TCs for the VSI or VEB.
  — EMP adds VEB, VSI, queue sets accordingly.
  — As a response to ADD VSI command, the EMP provides the queue set handles of the enabled TCs.
  — For each enabled TC, the PF is required to copy the queue set handle value to all queues belonging to the queue set.

38.25.5.7 Transmit Scheduler Resource Allocation Control

Each PF is configured with limited number of resources. The 10 GbE controller scheduler enables flexible resource allocation, but due to limited number of queue set supported, it is allowed to allocate an average of two queue sets per VSI.

Number of VSIs dedicated per PF is a switch configuration parameter configured via NVRAM. The remainder of available VSIs are shared on the first-come-first-serve bases.

Scheduler configuration firmware only controls resources allocated for PF, and enables the PF driver to distribute those resources within PF.

Allocation of queue sets per PF is proportional to the allocation of VSIs, assuming average of two queue sets per VSI. Shared queue sets can be allocated to PFs on first-come-first-serves bases, similar to VSIs.

38.25.5.8 Scheduler Configuration Schemes

The ETS-based scheme enables ETS configuration for the switching component or VSI directly connected to the physical port, and bandwidth allocation of other VSIs and switching components within each traffic type that can be either UP or TC enabled for that VSI or switching component. See Section 38.25.2.1.

Software should not attempt to configure bandwidth management attributes that are not supported by configured scheme. Firmware rejects performing invalid operations and returns an EPERM error described in Section 38.25.4.4.

38.25.5.8.1 ETS-based Scheme: Relative Bandwidth Credits Calculation

ETS-based configuration enables the 10 GbE controller's software to configure relative bandwidth allocation for each switching component and VSI within each traffic type (UP or TC). Software should use AQ commands described in Section 38.25.4.8 and Section 38.25.4.14, respectively.

The 10 GbE controller software should be able to configure ETS-based transmit scheduler based on:

- An absolute bandwidth allocation (in GB/s or Mb/s) for VSIs and switching components within each TC or UP enabled for the respective VSI or switching component.
- An absolute bandwidth allocation for VSIs and switching components and ETS configuration for each VSI and switching component.

In either case, configuration provided by system management software needs to be translated to relative bandwidth allocation of VSIs and switching components within TC or UP expressed in relative bandwidth allocation credits.

Figure 38-86 shows an example of ETS-based scheme bandwidth configuration and process of relative bandwidth credits calculation.

Figure 38-86.ETS-Based Scheme: Relative Bandwidth Calculation Example

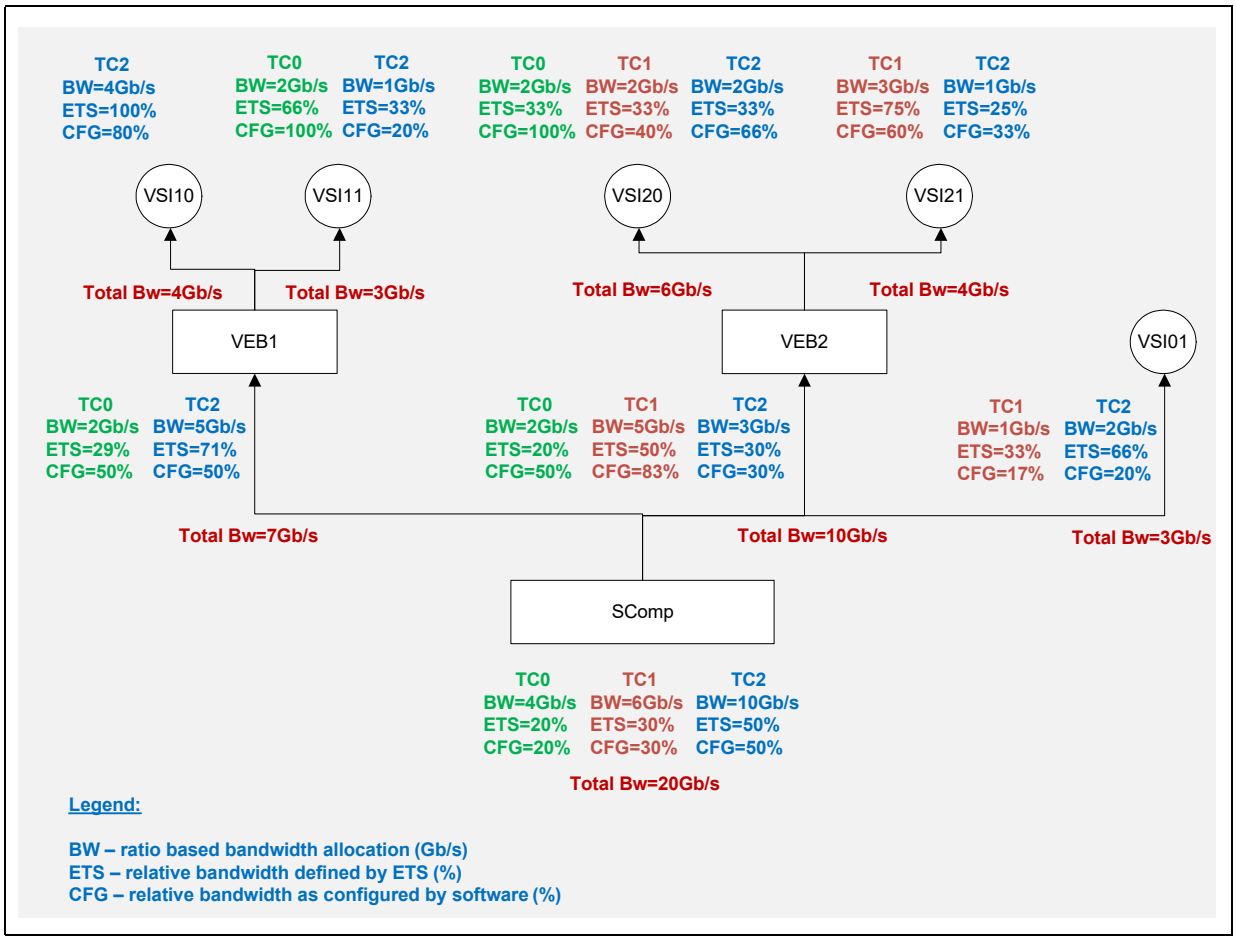


Figure 38-86 shows a single port configuration with S-comp. Two VEB switching components are connected to S-comp (VEB1 and VEB2), along with VSI (VSI01) directly attached to S-comp. Each VEB has two VSIs (VSI10 and VSI11, and VSI20 and VSI21, respectively).

Figure 38-86 for each switching component and VSI shows a per TC bandwidth allocation using three terms:

- ETS — Relative bandwidth allocation per TC within VSI/switching component in percentage with respect to other TCs. can be provided by system software along with total bandwidth allocated for the switching component and VSI.
- BW — Bandwidth allocation per TC in Gb/s, based on the physical wire speed of 20 Gb/s, and hierarchical bandwidth distribution specified by ETS. Can be either provided by system management software, or calculated based on the total bandwidth allocation for the switching component or VSI and ETS configuration.
- CFG — Calculated hierarchical relative bandwidth allocation of VSIs and switching components within each TC, in percentage with respect to other VSIs and switching components on the same hierarchy level.
- Total Bandwidth — Can be provided by system management software along with ETS configuration for the switching components and VSIs.

Note: ETS-based configuration does not guarantee bandwidth allocation for VSIs and switching components in ETS-based configuration, but it can use bandwidth allocation provided by system software to configure VSI and switching components relative bandwidth allocation within each TC or UP.

If a physical port is configured to a two-level ETS (such as bandwidth is allocated for TCs, and then distributed between UPs within same TC), then VSI and switching component bandwidth is allocated per UP.

If a physical port is configured to single-level ETS (bandwidth distribution between TCs only), the 10 GbE controller software can assume a logical one-to-one mapping of UPs to TCs for the bandwidth distribution purpose, and use AQ commands described in Section 38.25.4.8 and Section 38.25.4.14 to configure VSi and switching component bandwidth allocation within TC.

In either case, relative credits calculation should be done following this algorithm:

- If system software provided absolute bandwidth allocation per TC or UP:
  - The 10 GbE controller software should calculate a relative bandwidth allocation with respect to the parent switching entity within each traffic type (TC or UP).
  - Translate relative bandwidth allocation to credits, minimizing number of credits allocated.
- If system software provided an absolute bandwidth allocation for VSI or switching component and ETS:
  - The 10 GbE controller software should calculate an absolute bandwidth allocation per traffic type within VSI or switching component, which would be the same as an absolute bandwidth allocation of VSI or switching component within traffic type (TC or UP). Calculate a relative bandwidth allocation with respect to the parent switching entity within each traffic type (TC or UP).
  - Translate relative bandwidth allocation to credits, minimizing number of credits allocated

Table 38-331 lists an example of relative bandwidth credits calculation for the ETS-based system configuration shown on Figure 38-86. This example assumes that system management software provided a total bandwidth allocated for each VSI and switching component, and ETS.

Table 38-331.ETS-Based Scheme-relative Bandwidth Calculation Example (Sheet 1 of 2)

Switching Element Name	Parent Switching Element Name	Total BW	ETS (TCs)	ETS (%)	ETS (Gb/s)	Relative BW Within TC	Relative Credits within TC	Comments
S-comp	Physical Port	20 Gb/s	TC0	20%	4 Gb/s	100%	NA	No need to configure relative bandwidth allocation within traffic type for switching element attached directly to physical port.
			TC1	30%	6 Gb/s	100%	NA	
			TC2	50%	10 Gb/s	100%	NA	
VEB1	S-comp	7 Gb/s	TC0	29%	2 Gb/s	50%	1	VEB1 shares TC0 bandwidth with VEB2, 50% each, therefore 1 relative credit will correctly reflect a relative bandwidth allocation between VEB1 and VEB2 (1:1).
			TC2	71%	5 Gb/s	50%	5	VEB1 shared TC2 bandwidth with VEB2 and VSI01, with ratio 5:3:2, which is reflected by relative credits.

Table 38-331.ETS-Based Scheme-relative Bandwidth Calculation Example (Sheet 2 of 2)

Switching Element Name	Parent Switching Element Name	Total BW	ETS (TCs)	ETS (%)	ETS (Gb/s)	Relative BW Within TC	Relative Credits within TC	Comments
VEB2	S-comp	10 Gb/s	TC0	20%	2 Gb/s	50%	1	VEB2 shares TC1 bandwidth with VSI01, relative ratio is 5:1, which is reflected in relative credits.
			TC1	50%	5 Gb/s	83%	5	
			TC2	30%	3 Gb/s	30%	3	
VSI01	S-comp	3 Gb/s	TC1	33%	1 Gb/s	17%	1	
			TC2	66%	2 Gb/s	20%	2	
VSI10	VEB1	4 Gb/s	TC2	100%	4 Gb/s	80%	4	VSI10 shared TC2 bandwidth allocated for VEB1 with VSI11, with relative ratio 4:1, which is reflected in allocated relative credits.
VSI11	VEB1	3 Gb/s	TC0	66%	2 Gb/s	100%	1	VSI11 consumes all TC0 bandwidth allocated for VEB1, and therefore can be configured with 1 relative credit.
			TC2	33%	1 Gb/s	20%	1	
VSI20	VEB2	6 Gb/s	TC0	33%	2 Gb/s	100%	1	VSI20 consumes all TC0 bandwidth allocated for VEB2, and therefore can be configured with 1 relative credit.
			TC1	33%	2 Gb/s	40%	2	VSI20 shared TC1 bandwidth allocated to VEB2 with VSI21 with relative ratio 2:3, which is reflected in relative credits
			TC2	33%	2 Gb/s	66%	2	
VSI21	VEB2	4 Gb/s	TC1	75%	3 Gb/s	60%	3	VSI21 shared TC2 bandwidth allocated to VEB2 with VSI20 with relative ratio 1:2, which is reflected in relative credits.
			TC2	25%	1 Gb/s	33%	1	

Note: Table cells with gray background show numbers calculated by the 10 GbE controller software.

38.26 Host Memory Cache (HMC)

The 10 GbE controller uses host memory as a backing store for a number of context objects used to track queue state and iWARP objects. The HMC is the component responsible for managing the LAN and iWARP context objects stored in host memory. The HMC manages host memory on a per-PCI function basis and further breaks down each of the PCI function's HMC memory space into memory used to manage each context object that is in use for a given PCI function. Host software is responsible for allocating the host pages used by the HMC before accessing a specific object. Additionally, the amount of memory that can be used for HMC backing store for a specific function is dictated by the active resource profile that is determined by the software drivers operating environment and the number of PCI functions that are currently active. Resource profiles can be selected at software device driver initialization time.

38.26.1 Host Memory Usage

The HMC requires a backing store for numerous data structures to be resident in host memory to perform its functions. Table 38-332 lists the data structures and the amount of memory that needs to be allocated for each data structure. The HMC Object Location column indicates if the HMC object (and the associated backing store pages) is located only in the PF HMC object space or if it is located in both the PF and the VF HMC object space. In general, all LAN objects for both PFs and VFs are located in the PF HMC object space while PE HMC objects are separated into PF and VF specific HMC object spaces. The resources can be sparsely populated. For example, if a function is allotted 512 QPs and only 8 are used, then only 4 KB of memory needs to be allocated not the entire memory for all 512 QPs. Some HMC objects need to be fully populated at software device driver initialization such as PE hash table entries. Refer to the HMC resource allocation policies for LAN and the PE, respectively.

Table 38-332.HMC Objects

HMC Object	HMC Object Location	Size (Bytes)	Max Quantity	Description
LAN Transmit Queue	PFs	128	1536 Per Device	The PF owns the objects for the associated VFs. LAN absolute queue numbers assignment to PFs are determined by the programming of the PFLAN_QALLOC registers. HMC object indexes match the LAN queue indexes.
LAN Receive Queue	PFs	32	1536 Per Device	The PF owns the objects for the associated VFs. See the previous description of LAN transmit queues for details on the relationship between absolute LAN QP numbers and HMC object indexes.
Protocol Engine QP Context	PE Enabled PFs and VFs	512	256 KB	512 bytes are reserved for QP context. There is a maximum of 256 KB QP contexts per device. This number is divided among all PE enabled PCI functions. The memory for the VF objects is allocated by the PF driver and accessed using the PF Requester ID (RID). For a single function device, hardware reserves one of the 256 KB QPs for internal use.
Protocol Engine CQ Context	PE Enabled PFs and VFs	64	128 KB	64 bytes are reserved for CQ Context. There is a maximum of 128 KB CQ contexts per device. This number is divided among all PE enabled PCI functions. The memory for the VF objects is allocated by the PF driver and accessed using the PF RID.

Table 38-332.HMC Objects

HMC Object	HMC Object Location	Size (Bytes)	Max Quantity	Description
Protocol Engine Shared RQ	PE Enabled PFs and VFs	64	32 KB Per PCI Function	64 bytes are reserved for shared RQ context. There is a maximum of 32 KB S-RQ contexts per PE enabled PCI functions. The memory for the VF objects is allocated by the PF driver and accessed using the PF RID.
Protocol Engine TCP Timers	PE Enabled PFs And VFs	64	2^{28} Per PCI Function	This structure is allocated per PCI function. The number of elements allocated for this structure is dependent on the number of QPs associated with the PCI function. The equation for the number of objects is ((ROUNDUP512(number of QPs)/512) + 1) * 4096. The memory for the VF objects is allocated by the PF driver and accessed using the PF RID.
Protocol Engine Hash Table Entry	PE Enabled PFs And VF	64	(#PE QPs Multicast Groups)* 3 Per Function	This structure is allocated per PCI function. The number of elements allocated for this structure needs to be the number of QPs allocated to the function multiplied by 3. The memory for the VF objects is allocated by the PF driver and accessed using the PF RID. The actual number of hash table entries for the PF is programmed using the PFQF_CTL_2 and VFQF_CTL registers.
ARP Table Entry	PE Enabled PFs And VF	16	65536 Per PCI Function	The maximum size of the ARP table is 65536 entries per PE enabled PCI function. The memory for the VF objects is allocated by the PF driver and accessed using the PF RID.
Accelerated Port Bit Vector In-use	PE Enabled PFs And VF	8KB	1 Per PCI Function	This table is used for the 10 GbE controller to track each PCI functions usage of the accelerated port bit vector table and is only required for PE enabled PCI functions. The memory for the VF objects is allocated by the PF driver and accessed using the PF RID.
Memory Region Table Entry (MRTE)	PE Enabled PFs And VF	32	4 MB Per PCI Function	Each PE enabled PCI function can have up to 4 MB MRTE entries allocated. The memory for the VF objects is allocated by the PF driver and accessed using the PF RID.
Physical Buffer List Entry (PBLE)	PE Enabled PFs And VF	8	256 MB Per PCI Function	Each PE enabled PCI function can have up to 256 MB PBLE entries allocated. The memory for the VF objects is allocated by the VF driver and accessed using the VF RID. PF drivers are also able to (via VF) force PBLEs to be accessed with the PF RID by using the *use_pf_rid* bit in the memory registration associated with the PBLE. Any associated page descriptors are allocated by the PF driver and accessed using the PF RID.
Transmit FIFO	PE Enabled PFs And VFs	32	32 MB Per PCI Function	Each PE enabled PCI function can have up to 32 MB transmit FIFO entries per PCI function. Transmit FIFO entries are used to track un-acknowledged PE send queue work requests (WQEs). The number of transmit FIFO entries to allocate for a given PCI function can be calculated based on expected network performance and workload pattern. Allocating too few transmit FIFO entries results in head of line blocking for PE QPs running on a particular PCI function.The memory for the VF objects is allocated by the PF driver and accessed using the PF RID. Must be specified as a power of two.
Transmit FIFO Free List	PE Enabled PFs And VFs	4	16 MB Per PCI Function	Each PE enabled PCI function can have up to 16 MB transmit FIFO free list entries per PCI function.The memory for the VF objects is allocated by the PF driver and accessed using the PF RID.

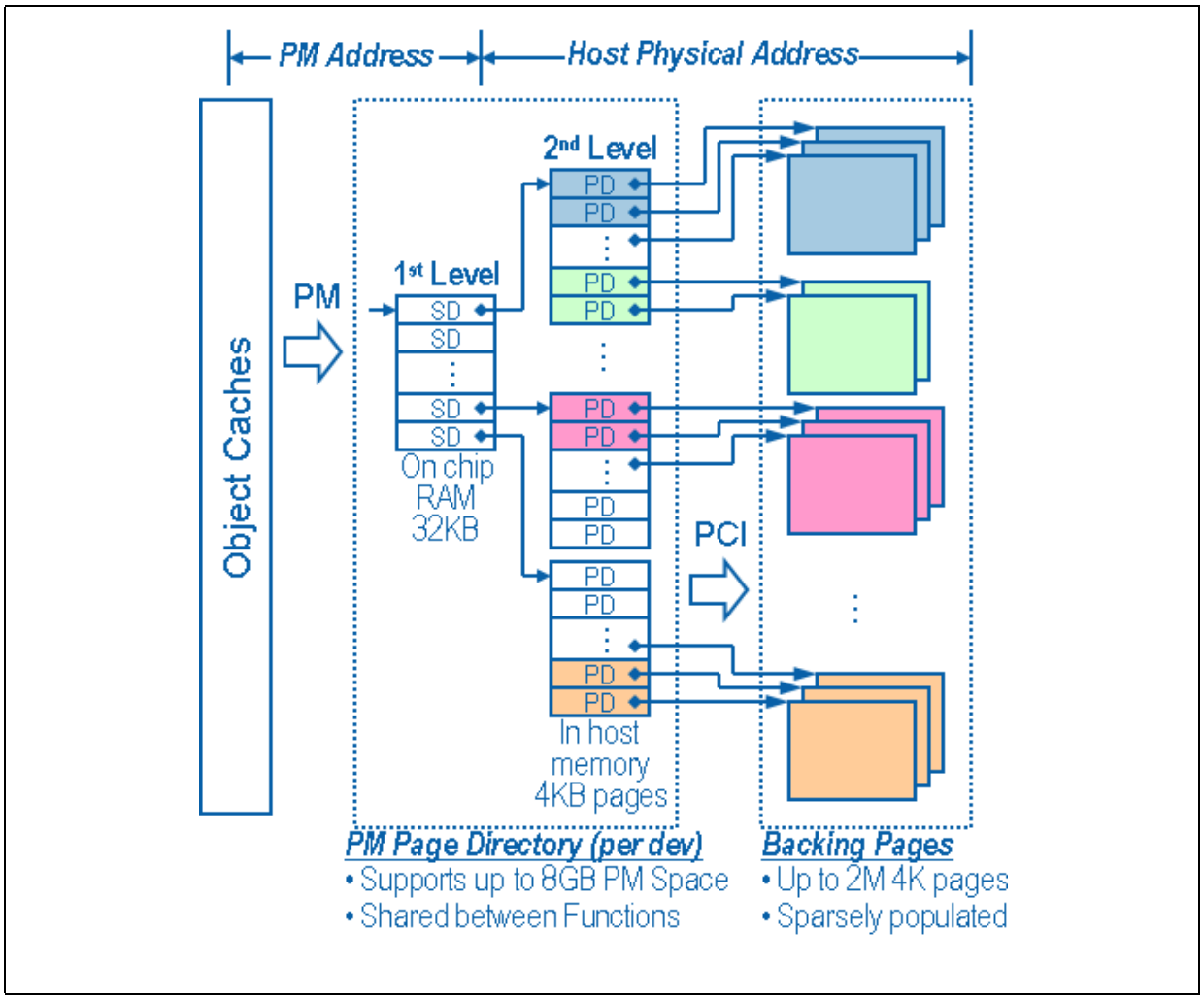
Table 38-332.HMC Objects

HMC Object	HMC Object Location	Size (Bytes)	Max Quantity	Description
Inbound RDMA Read Queue (IRRQ or Q1)	PE Enabled PFs And VF	64	32 MB Per PCI Function	Each PE enabled PCI function can have up to 32 MB IRRQ entries. IRRQ entries are allocated on a per QP basis and in quanta's of the maximum number of outstanding RDMA reads that are allowed multiplied by 2. In other words, if a PE enabled PCI function has 64 KB QPs enabled and each is allowed up to 64 outstanding RDMA reads, space for 8 MB (64K*(64*2))IRRQ entries (512 MB) must be allocated by the software device driver.The memory for the VF objects is allocated by the PF driver and accessed using the PF RID. Must be specified as a power of two.
Inbound RDMA Read Queue (IRRQ or Q1) Free List	PE Enabled PFs And VF	4	32 MB Per PCI Function	Each PE enabled PCI function can have up to 32 MB IRRQ free list entries.The memory for the VF objects is allocated by the PF driver and accessed using the PF RID.
Multicast Group	PE Enabled PFs And VF	64	8192 Per PCI Function	Each PE enabled PCI PF can have up to 8 KB multicast groups. VFs do not have multicast groups.
Address Handles	PE Enabled PFs and VFs	32	65536 per PCI Function	Each PE enabled PCI function might have up to 64 KB address handles.The memory for the VF objects is allocated by the PF driver and accessed using the PF Requestor ID (RID).

In order to access (and cache in on-chip memory) the data structures listed in Table 38-332, the HMC uses the concept of private memory address space. The 10 GbE controller has an 8 GB private memory address space that can be sparsely backed with host memory based on actual context usage. Software device drivers do not need to allocate pages for HMC objects that are not currently being used by the software device driver. The private memory address space is first broken down by PCI function, then by object or data structure type, and finally by object index. The portion of the private memory address space that is allocated to a particular PCI function is termed Function Private Memory (FPM). FPM objects for VF LAN are located in the associated PFs FPM. Also note that the VF FPMs are not programmed directly by the VF drivers. The PF driver uses the HMC function index to select the VF FPM to be programmed.

Figure 38-87 shows how the 10 GbE controller provides the address mapping between private memory and host physical addresses. Private memory address shown on the left side of the diagram indicates the 10 GbE controller private memory address from 0 to 8 GB-1. The 10 GbE controller works with private memory address space internally, which is converted to host physical addresses in order to access host memory.

Figure 38-87.Host Memory Cache Private Memory Address Space



The left portion of Figure 38-87 shows portions of the HMC that are resident on-chip. This portion includes the actual object caches that retain portions of the data from host memory to improve performance and the Segment Descriptors (SD). The SDs reside in a 32 KB on-chip RAM called the segment descriptor table. The segment descriptor table holds 4096 pointers to host memory pages. Unique ranges of sequential SDs in the segment descriptor table are allocated to a each PCI function that is active. The segment descriptor table is the first level of private memory address translation provided by the 10 GbE controller. SDs are programmed using the PFHMC_SDCMD, PFHMC_SDDATALOW, and PFHMC_SDDATAHIGH registers. PE CQP operations also can be used to program SDs. Everything to the right of the segment descriptor table in Figure 38-87 resides in host memory. Each PCI function has a set of registers (GLHMC_SDPART[n] and GLHMC_VFSDPART[n]) that define the base and number of SDs that belong to the PCI function. The GLHMC_SDPART[n] and GLHMC_VFSDPART[n] registers are programmed from the NVM. The 10 GbE controller provides range checking for each internal access to ensure that a given PCI function is never allowed to access memory outside of its valid range of SDs. The 10 GbE controller manages the SD base and number registers internally based on the resource profile that is loaded from the NVM.

The second level of private memory address translation provided by the 10 GbE controller are Page Descriptors (PDs). Each SD points to a single host page that is divided into 512 PDs that are 64-bit physical memory addresses. Each PD points to a backing page for the private memory address space. The total 8 GB private memory address space is derived using a fully populated segment descriptor table pointing to 4096, 4 KB host pages that hold the 2 M PDs. Each of the 2 M PDs point to host memory backing pages for a total of 8 GB of address space. As previously mentioned, there is no requirement to populate all SDs or PDs with memory if the portion of private memory address space is not in use by software. The format of the PD structure in host memory is listed in Table 38-333.

Table 38-333.HMC page descriptor format

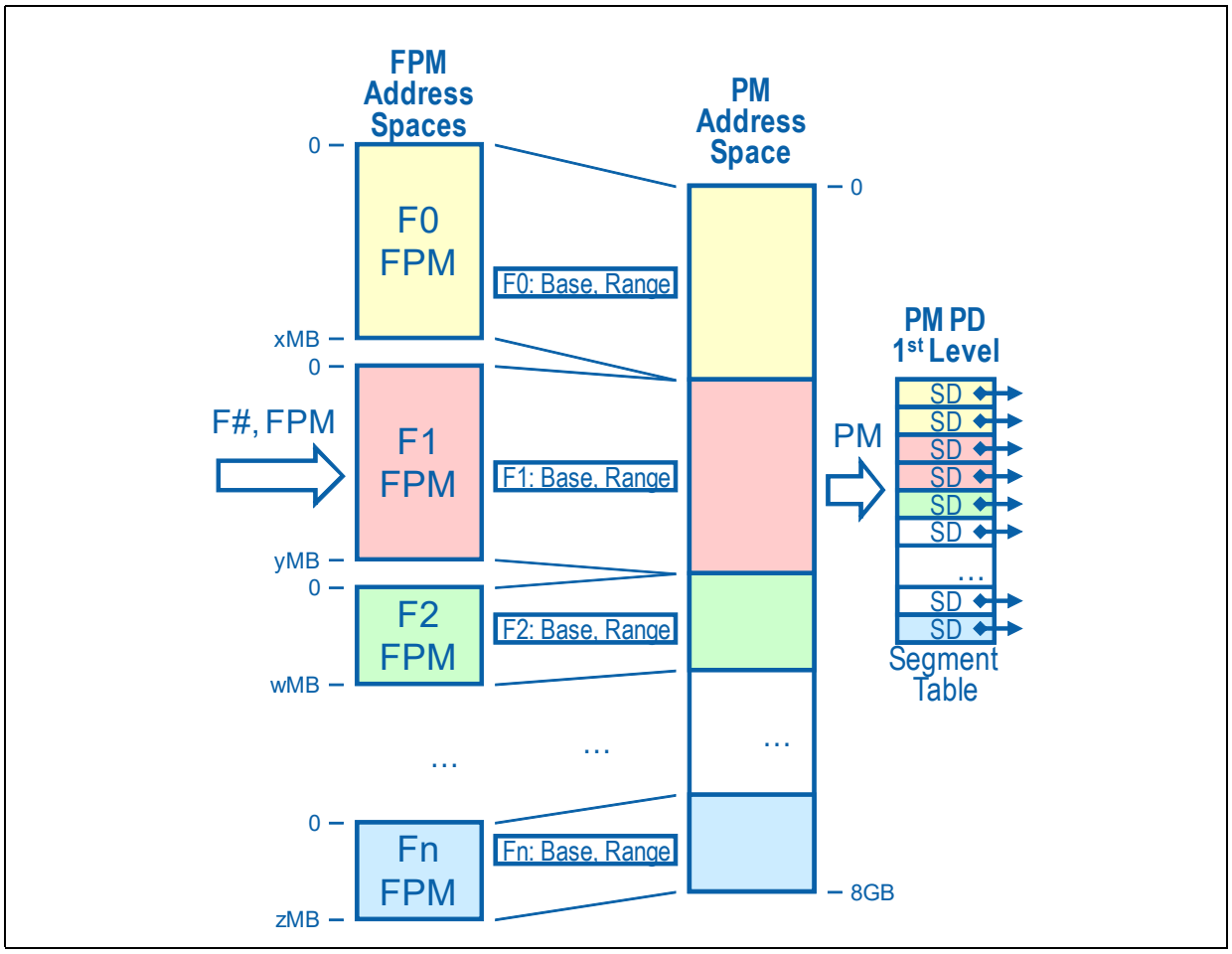
Byte Offset	[Bit Range]Field Name
0	[63:12] HMC Backing Page Physical Address [11:1] Reserved [0] PD Valid

The HMC backing page physical address is the address of a software device driver allocated page that holds HMC object context. This address must be aligned to a 4 KB address in host memory. The PD *Valid* bit enables software to sparsely populate the PD entries on an as needed basis once HMC context objects are needed. Software must allocate host memory pages that hold packed arrays of PDs as shown in Figure 38-87. The physical address of these PD pages are used to populate SD entries by using the PFHMC_SDCMD, PFHMC_SDDATALOW, and PFHMC_SDDATAHIGH registers. For more information on how to program SDs using these registers, see Section 38.26.5.

Private memory is further divided into separate PCI FPM addresses. A PCI function can be either a PF or VF. The first 16 FPM address spaces are reserved for Network Interface Cards (NICs) and PFs.The next 16 FPM address spaces are used for PFs that support PE-provided accelerations for iWARP or UDA. The final 32 FPM address spaces are used for VFs that support the PE provided accelerations for iWARP or UDA. NIC objects are located in the PF FPM.

Figure 38-88 shows how the private memory address space is divided up for each PCI function. The smallest amount of private memory that can be allocated to a function is 2 MB (1 SD). The maximum that could be allocated to a function would be the entire segment table in which case no other function can have any private memory resources. Note that the object caches address HMC objects using HMC function number to determine the correct FPM. The FPM identifies the range of private memory address space that belongs to a PCI function. Since each SD represents 2 MB of HMC PM address space, the FPM also identifies the range of SDs that belong to a PCI function.

Figure 38-88.HMC FPM Space



Each PCI function's private memory space is further divided into separate memory spaces for each object in host memory. Each PCI function has a set of registers per function that define the object's base address in FPM space and the bounds (or maximum number of entries) of a particular object. Figure 38-89 shows some of the current objects that reside in the private memory space. See Table 38-332 for a complete list of the objects. The FPM address is calculated based off of the object type, which identifies the object base register and object index (the FPM base has already been calculated). Ultimately, the FPM base address, object base address, object size, and object index are all used to determine the private memory address.

Figure 38-89.HMC FPM Object Access

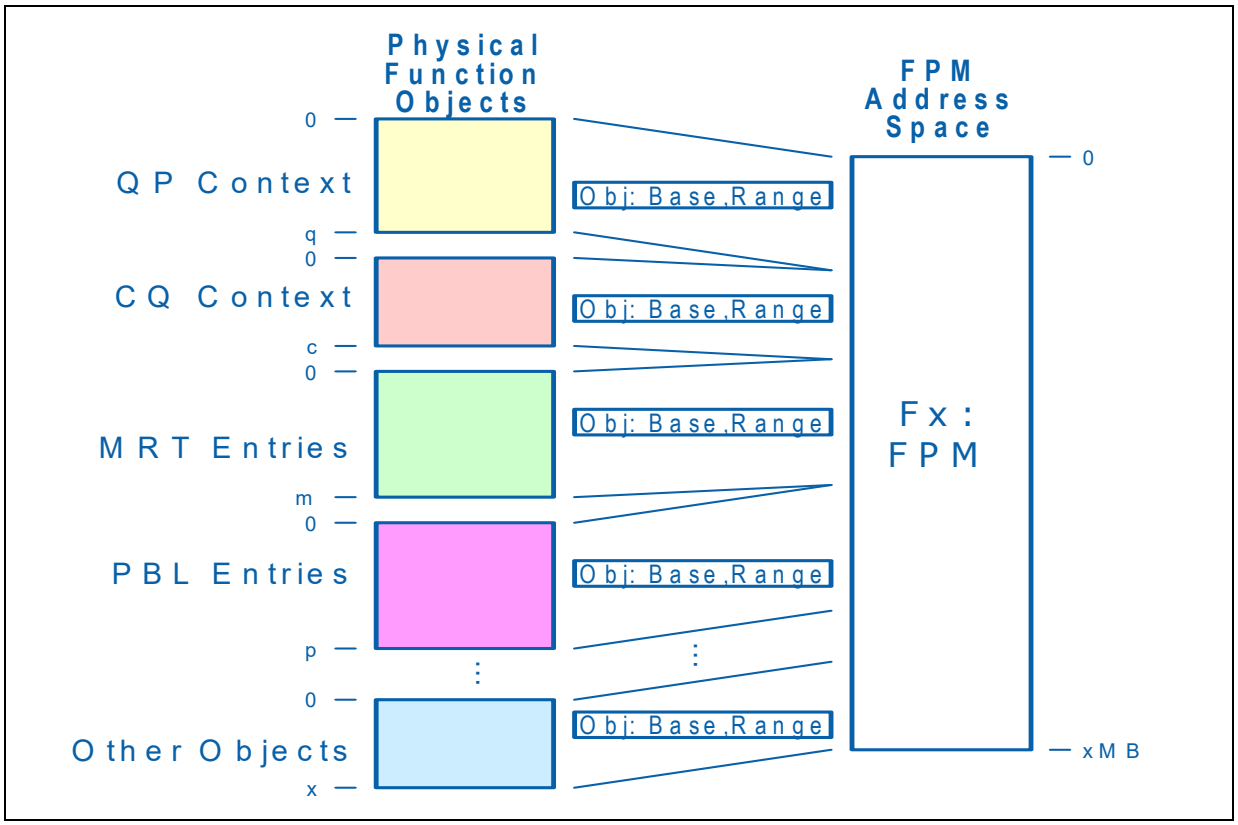
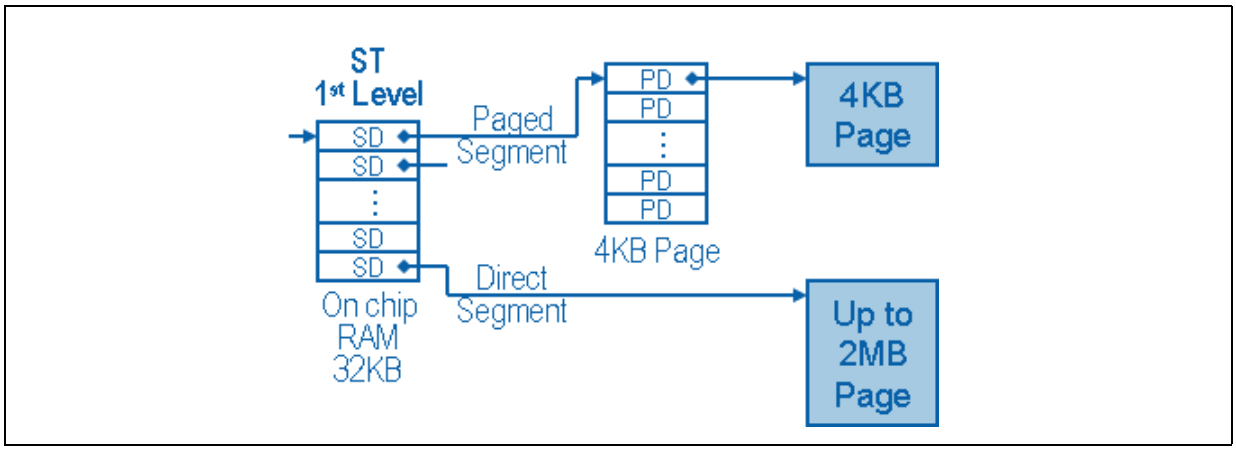


Figure 38-90 shows the decoding of the private memory address into host address. Additionally, Figure 38-90 shows an SD addressing a private memory space backing page directly instead of using the second level of indirect addressing (PD). Each PCI function can set any SD within its range of SDs to be either pointing to a PD or directly to a backing page. The segment type is specified in the PFHMC_SDDATALOW.PMSDTYPE register field. The direct segment approach can be used for PCI functions that do not have a large requirement for FPM space to reduce overhead incurred while accessing HMC objects. Additional usage of the direct segment approach is possible if the software device driver is able to allocate a physically contiguous range of pages large enough to hold the entire PD space needed to support the FPM required by the software device driver loading on a specific PCI function. This mode prevents an additional address lookup and increases the performance if the software device driver happens to allocate a block of physically contiguous memory or the operating system has support for 2 MB pages.

Figure 38-90.HMC Direct Segment



See Section 38.26.4 for more details on the specific formats of the SD entries for paged and direct addressing modes.

38.26.2 Private Memory Space Profiles

The 10 GbE controller's private memory address space configuration is a two step process. The first step is to partition the HMC related resources into per PCI function resources. The second step is to divide the resulting FPM into individual objects. In order to simplify resource allocation, The 10 GbE controller provides a resource profile concept the operating system environment into account to divide HMC private memory space and PE doorbell resources. These HMC resource profiles are used to partition the HMC segment descriptor table and PE doorbell resources. Software performs the FPM division on a per PCI function basis. The currently defined HMC resource profiles are listed in Table 38-334. PE doorbell resources place constraints on the HMC resource profiles since there is a fixed number of these resources on-die. There are enough doorbell resources for 256 KB protocol queue pairs and 128 KB PE completion queues. These resources must be partitioned between PCI functions that require PE functionality. The resource profiles drive the values found in the following registers:

- GLHMC_SDPART — Configures the HMC segment descriptor range per PF
- GLHMC_VFSDPART — Configures the HMC segment descriptor range per PE enabled VF
- GLHMC_DBQPPART — Configures the range of PE QP doorbells per PF
- GLHMC_VFDBQPPART — Configures the range of PE QP doorbells per PE enabled VF
- GLHMC_DBCQPART — Configures the range of PE CQ doorbells per PF
- GLHMC_VFDBCQPART — Configures the range of PE CQ doorbells per PE enabled VF
- GLHMC_CEQPART — Configures the range of PE CEQs per PF
- GLHMC_VFCEQPART — Configures the range of PE CEQs per PE enabled VF

These registers are loaded from the NVM to match the default profile. The admin queue can be used to change the programming of these registers to match any of the following resource profiles.

Table 38-334.10 GbE Controller HMC Resource Profiles

Profile Name	Description
Default	The default profile evenly distributes all HMC segment descriptor table entries and PE doorbell resources among all active PFs. No PE resources are allocated to VFs but LAN objects for VFs are allocated to the PFs associated with VFs as configured from NVM. See Section 38.26.2.2 for more details.
SR-IOV VF Primary	The SR-IOV VF primary HMC resource profile first distributes HMC LAN resources for PFs and VFS to PFs, then distributes a relatively small number of HMC PE resources to the active PFs, and then evenly distributes the remaining HMC PE resources among the PE enabled VFs. This resource profile assumes that the PE functionality primarily used by VFs when running a virtualized operating system. See Section 38.26.2.3 for more details.
SR-IOV Even Distribution	The SR-IOV even distribution HMC resource profile first distributes HMC LAN resources for PFs and VFs to PFs and then evenly distributes the HMC PE resources among the PFs and PE enabled VFs. This resource profile assumes that the PE functionality is used by PFs and VFs when running a virtualized operating system. See Section 38.26.2.4 for more details.

Each of these profiles are a set of equations that the 10 GbE controller uses to configure the HMC. This first phase of the HMC initialization is activated by an NVM reload (all settings revert back to the default profile) or by writing using the AQ commands specified in Section 38.27. Host software only needs to change from the default profile based upon the need for PE functionality in a VF. If this is not the case then the default profile covers all usage models. If PE functionality is required in a VF, the software device driver determines which of the two resource profiles best fit the users needs and selects that profile at software device driver initialization time. Note that all profiles cover the needs of LAN resources for both virtualized and non-virtualized environments.

38.26.2.1 Host HMC Resource Partitioning

The HMC resource profile can be selected by expansion ROM code before the operating system loads and also by the first software device driver that loads within the operating system. It is expected that expansion ROM boot code (PXE and iSCSI) does not need to change the profile, but it can change the profile if needed. Once the operating system software device driver loads, it is also allowed to change the HMC resource profile. The assumption is that all the instances of the operating system software device driver requests the same HMC resource profile. The currently active profile loaded by the 10 GbE controller can be retrieved by issuing the Get HMC Resource Profile AQ command.

38.26.2.2 Default HMC Profile Equations

The default HMC resource profile enables the maximum number of LAN queue objects for each PF and associated VFs and distributes all PE resources evenly among all active PFs (no PE resources are available for VFs). This includes partitioning the SD table, the PE QP and CQ doorbell resources, and the PE completion event queues. The number of active PFs is reported in the GLGEN_PCIFCNCNT registers.

Table 38-335.Default HMC Profile Examples

Number of Active PFs	PF Index	SD Base Index	SD Count	PE QP Doorbell Base Index	PE QP Doorbell Count	PE CQ Doorbell Base Index	PE CQ Doorbell Count	PE CEQ Base Index	PE CEQ Count
1	0	L2:0 PE:10	L2:1	0	256 KB	0	128 KB	0	128
	1-15	L2:0 PE:0	L2:0 PE:0	0	0	0	0	0	0
2	0	L2:0 PE:2	L2:1	0	128 KB	0	64 KB	0	128
	1	L2:1	L2:1	128 KB	128 KB	64 KB	64 KB	128	128
	2-15	L2:0 PE: 0	L2:0 PE:0	0	0	0	0	0	0
4	0	L2:0 PE:4	L2:1	0	64 KB	0	32 KB	0	64
	1	L2:1	L2:1	64 KB	64 KB	32 KB	32 KB	64	64
	2	L2:2	L2:1	128 KB	64 KB	64 KB	32 KB	128	64
	3	L2:3	L2:1	192 KB	64 KB	96 KB	32 KB	192	64
	4-15	L2:0 PE:0	L2:0 PE:0	0	0	0	0	0	0

The calculation used by the 10 GbE controller for the number of SDs should be SDs (4096) divided by the number of PFs, rounded down to the next integer. The calculation for the number of PE QPs is the number of PE QPs (256 KB) divided by the number of PFs, rounded down to the next full page size that would be needed for context. PE QP context size is 512 bytes per QP. The calculation for the number of PE CQs is similar except that the max number of PE CQs is 128 KB and the CQ context size is 64 bytes. The PE CEQ calculation is similar to the segment descriptor calculation except that the number of PE CEQs total is 256 and the maximum number or CEQs for any single PCI function is 128. Table 38-336 lists the resource counts for all possible PF counts in the default HMC resource profile.

Table 38-336.Initial 10 GbE Controller Host Memory Cache Partitioning per PF

PF Count	#SDs	#PE QPs	#PE CQs	#PE CEQs
1	1	262144	131072	128
2	1	131072	65536	128
3	1	87376	43648	85
4	1	65536	32768	64
5	1	52424	26176	51
6	1	43688	21824	42
7	1	37448	18688	36
8	1	32768	16384	32
9	1	29120	14528	28
10	1	26208	13056	25
11	1	23824	11904	23
12	1	21840	10880	21
13	1	20160	10048	19
14	1	18720	9344	18

Table 38-336.Initial 10 GbE Controller Host Memory Cache Partitioning per PF

PF Count	#SDs	#PE QPs	#PE CQs	#PE CEQs
15	1	17472	8704	17
16	1	16384	8192	16

38.26.2.3 SR-IOV VF Primary HMC Profile Equations

The SR-IOV VF primary HMC resource profile allocates PE resources in favor of the VFs. LAN resources are the same as the default profile since the LAN resources are always located in the PF HMC address space. In this case, the FPM spaces are broken down into two sets:

1. PFs
2. PE enabled VFs

The first 16 FPMs are always reserved for PFs even if less than 16 PFs are enabled. If less than 16 PFs are enabled, some of the FPMs will have zero resources allocated to them. The next set of FPMs (16-47) are the PE enabled VFs. The maximum number of VFs in this set is 32 but can be limited to less by setting the PE enabled VF count in the Set HMC Resource Profile AQ command.

In this resource profile, all enabled PFs are allocated the following resources:

- 10 SDs for PF PE
- 1024 PE queue Pairs
- 512 PE completion queues
- 8 PE completion event queues

After allocating the resources for the PFs and the NIC VF resources that reside in the PF, the remaining resources are evenly distributed across all PE enabled VFs. Assuming that there are 2 PFs active and 128 VFs (32 of which can be enabled for PE support), the HMC resources would be distributed as listed in Table 38-337.

Table 38-337.Example SR-IOV VF Primary Profile Resource Partitioning per FPM

Function Type	FPM Index Range	#SDs	#PE QPs	#PE CQs	#PE CEQs
PF (2)	L2: 0-1 PE: 16-17	L2: 10 PE: 10	1024	512	8
Inactive PF (14)	L2: 2-15 PE: 18-31	L2: 0 PE: 0	0	0	0
Protocol Engine Enabled VFs (32)	PE: 32-63		8128	4032	7
Total		4072	262144	130048	240

The equations for distributing PE QPs and CQs are similar to those used to distribute resources among PFs in the default HMC resource profile.

38.26.2.4 SR-IOV Even Distribution HMC Profile Equations

The SR-IOV even distribution HMC profile is used for environments where there is equal need for PE resources in the PFs and all PE enabled VFs in operating systems that support I/O virtualization. Typically the number of PE enabled VFs is reduced in this profile in order to enable sufficient PE resources for a VF to run HPC workloads.

In this resource profile, all enabled PFs are allocated the following resources:

- 1 SD for PF NIC
- Remaining SDs/(active PFs + RDMA enabled VFs) SDs for PE
- 256K/(active PFs + RDMA enabled VFs) PE queue pairs
- 128K/(active PFs + RDMA enabled VFs) PE completion queues
- 256/(active PFs + RDMA enabled VFs) PE completion Event queues

In this resource profile, all enabled VFs are allocated the following resources:

- Remaining SDs/(active PFs + RDMA enabled VFs) SDs for PE
- 256K/(active PFs + RDMA enabled VFs) PE queue pairs
- 128K/(active PFs + RDMA enabled VFs) PE completion queues
- 256/(active PFs + RDMA enabled VFs) PE completion event queues

Assuming the 2 PFs are enabled, 128 VFs are enabled, and 16 PE enabled VFs are configured, Table 38-338 lists the distribution of resources to the HMC FPMs.

Table 38-338.SR-IOV Even Distribution Profile Resource Partitioning Per FPM

Function Type	FPM Index Range	#SDs	#PE QPs	#PE CQs	#PE CEQs
PF (2)	L2: 0-1 PE: 16-17	PE: 22	14560	7232	14
Inactive PF (14)	L2: 2-15 PE: 18-31	L2: 0 PE: 0	0	0	0
PE Enabled VFs (16)	PE: 32-47	22	14560	7232	14
Total		4088	262080	130176	252

38.26.3 FPM Space Configuration

Once the NVM has set the default profile or the Set HMC Resource Profile AQ command has been used to set an appropriate HMC resource profile, the software device driver can then continue on with the second step of HMC configuration, which is to break down the FPM space into individual object regions. In order to do this, the software device driver must perform the following steps:

1. Determine the HMC function index to be configured. In the case of a PF, the HMC function number is equal to the PCI function number. In the case of a VF when the PE resources are configured, the HMC index depends on the programming of the HMC PM function table using the CQP manage HMC PM function table operation. Note that there are two arrays defined for each HMC FPM register (one array for PFs and a second array for the PE enabled VFs).
2. For LAN queue objects, read the PFLAN_QALLOC register associated with the PF to find the base queue index and number of queues associated with the PF.
3. For PE objects, read the current HMC configuration information in order to know how to calculate the numbers of each HMC object that need to be requested.

a. GLHMC_SDPART[n] or GLHMC_VFSDPART[n] — A PE enabled VF register associated with the HMC FPM reports the currently available SDs for a given PCI function, which can be used to calculate the maximum FPM space.

b. GLHMC_DBQPPART[n] or GLHMC_VFDBQPPART[n] — A PE enabled VF reports the maximum number of PE QPs that can be used. The software device driver does not have to allocate maximum FPM space based on PMDBMAXQP if less QPs are needed for the active software device driver deployment model.

c. GLHMC_DBCQPART[n] or GLHMC_VFDBCQPART[n] — A PE enabled VF used to determine the maximum number of PE completion queues that can be used by a given PCI function. Similarly to the maximum QP count, the software device driver is not required to allocate the maximum amount of FPM space for CQs if the software device driver deployment model requires less.

d. GLHMC_CEQPART[n] or GLHMC_VFCEQPART[n] — A PE enabled VF used to determine the maximum number of CEQs that can be used with a given PCI function. Note that PE CEQs do not consume FPM space but are still partitioned as part of the HMC resource profiles.

e. GLHMC_{object}MAX — Other object maximum values are found in these registers.

4. Each FPM object size register is written with the minimum of the values determined in steps 2-4 or the actual software device driver needs.

5. Write the base and count registers for each LAN object based on the maximum object index that would be used for the PCI function (HMC PM LAN objects are indexed with the absolute queue number).

For PE objects, issue the commit FPM values CQP operations to program the object base registers.The HMC resource profiles enable enough FPM space for the maximum number of LAN objects.

Table 38-339 lists all the HMC objects and the registers used to determine the object location within FPM space including the size and limit of each object.

Table 38-339.FPM Object Registers

HMC Object	Base Register Array	Object Count Register Array	Maximum Object Count Register	Object Element Size Register
LAN Transmit Queue	GLHMC_LANTXBASE	GLHMC_LANTXCNT	PFLAN_QALLOC	GLHMC_LANTXOBJSZ
LAN Receive Queue	GLHMC_LANRXBASE	GLHMC_LANRXCNT	PFLAN_QALLOC	GLHMC_LANRXOBJSZ
PE QP Context	GLHMC_PEQPBASE GLHMC_VFPEQPBASE	GLHMC_PEQPCNT GLHMC_VFPEQPCNT	GLHMC_DBQPMAX	GLHMC_PEQPOBJSZ
PE CQ Context	GLHMC_PECQBASE GLHMC_VFPECQBASE	GLHMC_PECQCNT GLHMC_VFPECQCNT	GLHMC_DBCQMAX	GLHMC_PECQOBJSZ
PE Shared Receive Queue	GLHMC_PESRQBASE GLHMC_VFPESRQBASE	GLHMC_PESRQCNT GLHMC_VFPESRQCNT	GLHMC_PESRQMAX	GLHMC_PESRQOBJSZ
PE TCP Timers	GLHMC_PETIMERBASE GLHMC_VFPETIMERBASE	GLHMC_PETIMERCNT GLHMC_VFPETIMERCNT	GLHMC_PETIMERMAX	GLHMC_PETIMEROBJSZ
PE Hash Table Entry	GLHMC_PEHTEBASE GLHMC_VFPEHTEBASE	GLHMC_PEHTCNT[1] GLHMC_VFPEHTCNT	GLHMC_PEHTMAX	GLHMC_PEHTEOBJSZ
ARP Table Entry	GLHMC_PEARPBASE GLHMC_VFPEARPBASE	GLHMC_PEARPCNT GLHMC_VFPEARPCNT	GLHMC_PEARPMAX	GLHMC_PEARPOBJSZ
Accelerated Port Bit Vector In-use	GLHMC_APBINUSEBASE GLHMC_VFAPBINUSEBASE	N/A (1 table per function)	N/A (1 table per function)	N/A (8KB fixed)
Memory Region Table Entry (MRTE)	GLHMC_PEMRBASE GLHMC_VFPEMRBASE	GLHMC_PEMRCNT GLHMC_VFPEMRCNT	GLHMC_PEMRMAX	GLHMC_PEMROBJSZ

Table 38-339.FPM Object Registers

HMC Object	Base Register Array	Object Count Register Array	Maximum Object Count Register	Object Element Size Register
Transmit FIFO	GLHMC_PEXFBASE GLHMC_VFPEXFBASE	GLHMC_PEXFCNT GLHMC_VFPEXFCNT	GLHMC_PEXFMAX	GLHMC_PEXFOBJSZ
Transmit FIFO Free List	GLHMC_PEXFFLBASE GLHMC_VFPEXFFLBASE	GLHMC_PEXFFLCNT GLHMC_VFPEXFFLCNT	GLHMC_PEXFFLMAX	N/A (4-byte fixed)
Inbound RDMA Read Queue (IRRQ or Q1)	GLHMC_PEQ1BASE GLHMC_VFPEQ1BASE	GLHMC_PEQ1CNT GLHMC_VFPEQ1CNT	GLHMC_PEQ1MAX	GLHMC_PEQ1OBJSZ
Inbound RDMA Read Queue (IRRQ or Q1) Free List	GLHMC_PEQ1FLBASE GLHMC_VFPEQ1FLBASE	GLHMC_PEQ1FLCNT GLHMC_VFPEQ1FLCNT	GLHMC_PEQ1FLMAX	N/A (4-byte fixed)
Multicast Group	GLHMC_FSIMCBASE	GLHMC_FSIMCCNT	GLHMC_FSIMCMAX	GLHMC_FSIMCOBJSZ
Address Handles	GLHMC_FSIAVBASE GLHMC_VFFSIAVBASE	GLHMC_FSIAVCNT GLHMC_VFFSIAVCNT	GLHMC_FSIAVMAX	GLHMC_FSIAVOBJSZ
Physical Buffer List Entry (PBLE)	GLHMC_PEPBLBASE GLHMC_VFPEPBLBASE	GLHMC_PEPBLCNT GLHMC_VFPEPBLCNT	GLHMC_PEPBLMAX	N/A (8-byte fixed)
Multicast Group	GLHMC_FSIMCBASE	GLHMC_FSIMCCNT	GLHMC_FSIMCMAX	GLHMC_FSIMCOBJSZ
Address Handles	GLHMC_FSIAVBASE GLHMC_VFFSIAVBASE	GLHMC_FSIAVCNT GLHMC_VFFSIAVCNT	GLHMC_FSIAVMAX	GLHMC_FSIAVOBJSZ

Notes:

1. For PFs, the sum of PFQF_CTL_0.PEHSIZE and PFQF_CTL_0.PEDSIZE must be smaller than or equal to GLHMC_PEHTCNT. For VFs, the sum of VPQF_CTL_PE.PEHSIZE and VPQF_CTL_PE.PEDSIZE must be smaller than GLHMC_VFPEHTCNT.

38.26.3.1 Programming the HMC FPM Base Registers

Host software is responsible for setting up the GLHMC_{object}CNT and GLHMC_{object}BASE registers for LAN objects. All settings of FPM registers impact only the function associated with the registers. The software device driver on a given PCI function must only program the GLHMC_{object}CNT and GLHMC_{object}BASE registers for HMC PMs that are owned by that same PCI function or HMC PMs that are associated with PE enabled VFs that the PF owns. The FPM base of the first HMC object (GLHMC_LANTXBASE) for each PCI function is always zero. The FPM base of subsequent HMC objects increment from previous HMC object base, the number of elements for the previous HMC object, and the size of the previous HMC object element. Additional rounding is necessary to get to the next FPM address that is properly aligned for the HMC object under consideration. Table 38-340 lists the FPM object order that must be maintained for proper HMC operation and the alignment requirements for each object.

Table 38-340.FPM Object Order and Alignment

HMC Object Order	HMC Object	Alignment Requirement
0	LAN Transmit Queue	512 bytes
1	LAN Receive Queue	512 bytes
2	Reserved	512 bytes
3	Reserved	512 bytes
4	PE QP Context	2 MB (must be the start of an SD)
5	PE CQ Context	512 bytes
6	PE Shared Receive Queue	512 bytes
7	PE Hash Table Entry	512 bytes
8	ARP Table Entry	512 bytes

Table 38-340.FPM Object Order and Alignment

HMC Object Order	HMC Object	Alignment Requirement
9	Accelerated Port Bit Vector In-use	512 bytes
10	Memory Region Table Entry (MRTE)	512 bytes
11	Transmit FIFO	512 bytes
12	Transmit FIFO Free List	512 bytes
13	Inbound RDMA Read Queue (IRRQ or Q1)	512 bytes
14	Inbound RDMA Read Queue (IRRQ or Q1) Free List	512 bytes
15	PE TCP Timers	512 bytes
16	Physical Buffer List Entry (PBLE)	512B (4KB for VFs, handled by CQP firmware)
17	Read Response FIFO	512 bytes
18	Read Response FIFO Free List	512 bytes
19	Header Information	512 bytes
20	Metadata	512 bytes
21	Out of Order Send Completion	512 bytes
22	Out of Order Send Completion FIFO Free List	512 bytes

As a partial example of how the 10 GbE controller must program the registers listed in Table 38-340, the following steps should be taken to program the first three HMC objects:

1. Program GLHMC_LANTXBASE = 0
2. Program GLHMC_LANRXBASE = ROUNDUP512((GLHMC_LANTXBASE*512)+(GLHMC_LANTXCNT*$2^{GLHMC\_LANTXOBJSZ}$)) / 512

The special case of PE QPs must be rounded up to the next SD boundary. The base registers for PE objects (indexes 4 and up) can be set using the commit FPM values CQP operation.

Note: The software device driver can choose to set the GLHMC_{obj}CNT register to zero if it does not need to use an object. For example, only the GLHMC_LANTXOBJSZ and GLHMC_LANRXOBJSZ register need to be non-zero if a PE resource is not needed.

38.26.4 Populating HMC Backing Pages

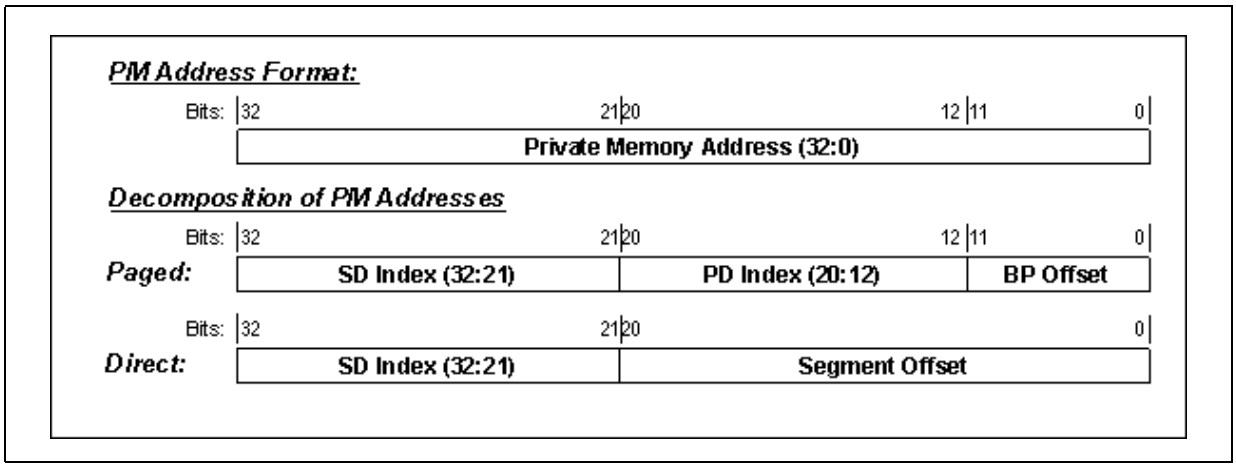
Once the HMC resource profile has been picked (Section 38.26.2) and the FPM space has been programmed (Section 38.26.3), the software device driver must populate the HMC backing pages for the PCI function that it is initializing. The first step in this phase of initialization is to allocate 4 KB pages for the PDs. Each 4 KB PD page holds 512 PDs and occupies a single SD entry. Once a 4 KB page has been allocated, initialized to zero and pinned by software, the PFHMC_SDCMD, PFHMC_SDDATALOW, and PFHMC_SDDATAHIGH registers are used to populate the SD table for the PCI function. PE enabled VFs use the GLHMC_VFSDCMD[n], GLHMC_VFSDDATALOW[n], and GLHMC_SDDATAHIGH[n] where n is the HMC VF index. PE related SDs are programmed using the update PE SDs CQP operation.

A software device driver on a given PCI function must only manipulate SD table entries that are allocated for that PCI function via the SD partitioning process. SDs are addressed on a per PCI function basis starting and zero and is limited by PMSDMAX. Note that software is not aware of the actual portion of the SD table that it is using.

Accesses outside of the SD range configured by the NVM using the PFHMC_SDCMD or GLHMC_VFSDCMD registers are ignored (operation is not performed) by the 10 GbE controller and an error is returned in the completion for the operation that is in error. The second step in this phase of software device driver initialization is to allocate additional host pages for backing HMC FPM objects for use by the 10 GbE controller before the software device driver attempts to access the object. The breakdown of the FPM address into components is listed in Table 38-340. Note that GLHMC_VFSDCMD, GLHMC_VFSDDATALOW, and GLHMC_SDDATAHIGH registers might be written only through CQP operations.

Note: Direct mapped queues in the VF object cache must be fully populated for the entire 2 MB space of the SD.

Figure 38-91.FPM Address Decomposition



The identification of which SDs to populate and which HMC FPM backing pages to populate in the PD pages can be calculated as follows in the paged scenario:

FPM_object_address = (GLHMC_{object}BASE*512) + ($2^{GLHMC\_\{object\}OBJSZ}$ * element_index)

SD_index = INT(FPM_object_address / 2 MB)

PD_index = INT(FPM_object_address / 4 KB) and 0x1FF

HMC_PM_index = PF index or the HMC VF FPM index

An example of populating the backing pages the HMC is shown assuming that a software device driver needs to allocate FPM backing pages for 512 LAN receive queues starting at index 0:

1. Allocate one PD page (capable of holding 512 backing pages of 4 KB each).
2. Identify the first SD necessary:
   a. FPM_object_address = (GLHMC_LANTXBASE[HMC_PM_index]*512) + ($2^{GLHMC\_LANTXOBJSZ}$*GLHMC_LANTXCNT[HMC_PM_index]).
   b. FPM_object_limit = FPM_object_address + ($2^{GLHMC\_LANTXOBJSZ}$*512).
   c. SD_index = FPM_object_address / 2 MB.
   d. Last_SD_index = ((FPM_object_limit - 1) / 2 MB.
3. Allocate, zero and pin a host memory page (PD page) for each SD needed from SD_index to Last_SD_index.
4. Calculate the number of PDs that need to be allocated:

a. FPM_PD_index = (FPM_object_address / 4 KB) and 0x1FF.

b. FPM_PD_limit_index = ((FPM_object_limit - 1) / 4 KB) and 0x1FF.

c. FPM_PD_count = FPM_PD_limit + 1 - FPM_PD_index.

5. Initialize the PDs:

   a. Allocate/zero/pin FPM_PD_count pages (these are the FPM object backing pages).

   b. Initialize each of the PDs with the physical address of a page allocated in step 5a and set the PD valid bit (see Table 38-333 for the format). The PDs are in the PD pages allocated in step 3.

6. Update the SD table using the PFHMC_SDCMD, PFHMC_SDDATALOW, and PFHMC_SDDATAHIGH registers for each PD page allocated in step 3 (if the PDs are associated with a PE enabled VF then the GLHMC_VFSDCMD[HMC_PM_index], GLHMC_VFSDDATALOW[HMC_PM_index] and GLHMC_VFSDDATAHIGH[HMC_PM_index] registers must be used). These registers are programmed via the update PE SDs CQP operation for backing pages related to PE objects.

   a. Write the most significant 32 bits of the physical address of the PD page to the PFHMC_SDDATAHIGH.

   b. Write the last significant 32 bits of the physical address of the PD page to the PFHMC_SDDATALOW ensuring that the lower 12 bits are 0.

   c. Write 512 to PFHMC_SDDATALOW.PMSDBPCOUNT. If this was the last SD of the FPM, the value might be lower than 512, but PE QPs are in the middle of FPM space so the value must be 512. The PMSDBPCOUNT field is used by the 10 GbE controller to calculate the end of the FPM space without having to read the valid bit for each individual PD entry.

   d. Write 0b to PFHMC_SDDATALOW.PMSDTYPE.

   e. Write 1b to PFHMC_SDDATALOW.PMSDVALID.

   f. Write PFHMC_SDCMD with PMSDIDX set to the proper SD index value and PMSDWR=1b.

When this process completes for typical configurations, the second SD is populated with the address of a single PD page, and entries 1-129 are populated with the address of the 128 FPM object backing pages that have been allocated.

If the PDs are associated with a PE enabled VF, registers GLHMC_VFSDCMD[HMC_PM_index], GLHMC_VFSDDATALOW[HMC_PM_index] and GLHMC_VFSDDATAHIGH[HMC_PM_index] must be used. These registers are programmed via the update PE SDs CQP operation for backing pages related to PE objects, but PE QPs are in the middle of FPM space so the value must be 512.

38.26.5 De-populating HMC Backing Pages

The process of de-populating and freeing HMC object backing pages is as follows:

- Make sure software and hardware are not going to access objects in the pages.
- Calculate the SD range and/or PD range that provides the address mapping to the 10 GbE controller.
- Update the associated PD entries.
- Invalidate the on-die PD cache entries using the PFHMC_PDINV (or GLHMC_VFPDINV[HMC_PM_index]) register.

- Update the SDs using the PFHMC_SDCMD, PFHMC_SDDATALOW, and PFHMC_SDDATAHIGH registers to notify the 10 GbE controller that the pages are no longer valid for use. If the PDs are associated with a PE enabled VF then the GLHMC_VFSDCMD[HMC_PM_index], GLHMC_VFSDDATALOW[HMC_PM_index] and GLHMC_VFSDDATAHIGH[HMC_PM_index] registers must be used. These registers are programmed via the update PE SDs CQP operation for backing pages related to PE objects.

38.26.5.1 Removing a Backing Page

Once software has determined that a backing page is no longer needed, software must clear the *PD_Valid* bit (see Table 38-333) in the PD entry that references the backing page. After clearing the *PD_Valid* bit in the PD in host memory, software must then write the PFHMC_PDINV register with the SD index and PD index of the newly invalidated PD entry. This is to ensure that references to the invalid PD entry have been removed from any the 10 GbE controller cache. Once this write completes, the backing page can be freed by software. Writing the PFHMC_PDINV register is not required for direct SDs since there is not a PD involved in addressing the HMC backing pages. If the backing pages are associated with a PE enabled VF, register GLHMC_VFPDINV[HMC_PM_index] must be used.

38.26.5.2 Removing a Page Descriptor Page

Once software has determined that an entire PD page is no longer needed, the PFHMC_SDDATALOW register must be written with PFHMC_SDDATALOW.PMSDVALID set to 0b and PFHMC_SDCMD must be written with PMSDIDX set to the proper SD index value and PMSDWR=1b. Once this sequence completes, software is free to de-allocate or re-use the PD page. If the PDs are associated with a PE enabled VF, registers GLHMC_VFSDCMD[HMC_PM_index], GLHMC_VFSDDATALOW[HMC_PM_index] and GLHMC_VFSDDATAHIGH[HMC_PM_index] must be used. These registers are programmed via the update PE SDs CQP operation for backing pages related to PE objects.

38.26.6 Special Cases for PE Objects

There is one instance when LAN objects and PE objects are handled differently.

- LAN objects for VFs exist in the HMC FPM of the PF but PE objects exist in the HMC FPM of up to 32 HMC VFs. Since there are less HMC VF FPM spaces than there are VFs, there is a mapping required that is described in Section 38.26.6.1.

38.26.6.1 VF Support

The HMC supports 16 HMC FPM spaces that are dedicated to PFs (index 0-15) and 32 HMC FPM spaces that are available for VFs. The 32 VF FPM spaces do not have LAN objects since these objects reside in the associated PF FPM space. Since there are less HMC FPM VF spaces available than actual VFs for the 10 GbE controller, a mapping is required. The VF mapping is performed by CQP firmware using the manage HMC PM function table operation.

38.26.7 HMC Error Reporting

HMC related errors are reported through the PFHMC_ERRORINFO and PFHMC_ERRORDATA registers. The HMC_ERR interrupt status bit in the PFINT_ICR0 register can also deliver an interrupt for HMC errors if the interrupt is enabled in the PFINT_ICR0_ENA register. When the HMC detects an error, it sets the PFHMC_ERRORINFO.ERROR_DETECTED bit along with the relevant information in the

other fields of the PFHMC_ERRORINFO and PFHMC_ERRORDATA registers. No further notification of subsequent HMC errors associated with any given PF are issued until the current error is acknowledged by writing a 0b to the PFHMC_ERRORINFO.ERROR_DETECTED bit. Table 38-341 lists the errors detected for each HMC object and the behavior associated with each error.

Table 38-341.HMC Errors

HMC Object	Error Type(s)	Error Behavior
LAN Transmit Queue	PMF Invalid, Invalid PMF Index	Not applicable to this object type.
	Invalid LAN Queue Index or HMC Object Index Too Large	Index of LAN queue is reported in the PFHMC_ERRORDATA register. The packets associated with the queue are not transmitted.
	HMC Private Memory Address Too Large, Segment Descriptor Invalid, Segment Descriptor Too Small, Page Descriptor Invalid	The private memory address of the LAN queue object is reported in the PFHMC_ERRORDATA register. Packets associated with the queue are not transmitted.
LAN Receive Queue	PMF Invalid, Invalid PMF Index	Not applicable to this object type.
	Invalid LAN Queue Index or HMC Object Index Too Large	Index of the LAN queue is reported in the PFHMC_ERRORDATA register. Packets associated with the queue are dropped.
	HMC Private Memory Address Too Large, Segment Descriptor Invalid, Segment Descriptor Too Small, Page Descriptor Invalid	The private memory address of the LAN queue object is reported in the PFHMC_ERRORDATA register. Packets associated with the queue are dropped.
Protocol Engine QP Context	PMF Invalid, Invalid PMF Index, HMC Object Index Too Large	The index of the PE QP is reported in the PFHMC_ERRORDATA register. Packets associated with the QP are dropped. PE AEs are not reported.
	Invalid LAN Queue Index	Not applicable to this object type.
	HMC Private Memory Address Too Large, Segment Descriptor Invalid, Segment Descriptor Too Small, Page Descriptor Invalid	The private memory address of the PE QP object is reported in the PFHMC_ERRORDATA register. Packets associated with the QP are dropped. PE AEs are not reported.
Protocol Engine CQ Context	PMF Invalid, Invalid PMF Index, HMC Object Index Too Large	The index of the PE CQ is reported in the PFHMC_ERRORDATA register. Completions for all QPs associated with the CQ are dropped. PE AEs are not reported.
	Invalid LAN Queue Index	Not applicable to this object type.
	HMC Private Memory Address Too Large, Segment Descriptor Invalid, Segment Descriptor Too Small, Page Descriptor Invalid	The private memory address of the PE CQ object is reported in the PFHMC_ERRORDATA register. Completions for all QPs associated with the CQ are dropped. PE AEs are not reported.
Protocol Engine Shared Receive Queue	PMF Invalid, Invalid PMF Index, HMC Object Index Too Large	The index of the PE S-CQ is reported in the PFHMC_ERRORDATA register. The receive data is dropped and the QP is put into the terminate state. A PE AE for the QP is reported.
	Invalid LAN Queue Index	Not applicable to this object type.
	HMC Private Memory Address Too Large, Segment Descriptor Invalid, Segment Descriptor Too Small, Page Descriptor Invalid	The private memory address of the PE S-RQ object is reported in the PFHMC_ERRORDATA register. The receive data is dropped and the QP is put into the terminate state. A PE AE for the QP is reported.

Table 38-341.HMC Errors

HMC Object	Error Type(s)	Error Behavior
Protocol Engine TCP Timers	PMF Invalid, Invalid PMF Index, HMC Object Index Too Large	The index of the PE timer object is reported in the PFHMC_ERRORDATA register. Timer operation associated with the function goes idle. Host software is responsible for cleaning up all PE resources associated with the PCI function and reporting a fatal adapter error to the RDMA protocol stack.
	Invalid LAN Queue Index	Not applicable to this object type.
	HMC Private Memory Address Too Large, Segment Descriptor Invalid, Segment Descriptor Too Small, Page Descriptor Invalid	The index of the PE timer object is reported in the PFHMC_ERRORDATA register. Timer operation associated with the function goes idle. Host software is responsible for cleaning up all PE resources associated with the PCI function and reporting a fatal adapter error to the RDMA protocol stack.
Protocol Engine Hash Table Entry	PMF Invalid, Invalid PMF Index, HMC Object Index Too Large	The index of the PE hash table entry is reported in the PFHMC_ERRORDATA register. Packets associated with the hash table entry are dropped. PE AEs are not reported.
	Invalid LAN Queue Index	Not applicable to this object type.
	HMC Private Memory Address Too Large, Segment Descriptor Invalid, Segment Descriptor Too Small, Page Descriptor Invalid	The private memory address of the PE hash table entry object is reported in the PFHMC_ERRORDATA register. Packets associated with the hash table entry are dropped. PE AEs are not reported.
ARP Table Entry	PMF Invalid, Invalid PMF Index, HMC Object Index Too Large	The index of the ARP table entry is reported in the PFHMC_ERRORDATA register. Packets associated with the ARP table entry are dropped. PE AEs are not reported.
	Invalid LAN Queue Index	Not applicable to this object type.
	HMC Private Memory Address Too Large, Segment Descriptor Invalid, Segment Descriptor Too Small, Page Descriptor Invalid	The private memory address of the ARP table entry is reported in the PFHMC_ERRORDATA register. Packets associated with the ARP table entry are dropped. PE AEs are not reported.
Accelerated Port Bit Vector In-use	PMF Invalid, Invalid PMF Index, HMC Object Index Too Large	The index of the APBVT entry is reported in the PFHMC_ERRORDATA register. Packets associated with the APBVT entry are dropped. PE AEs are not reported.
	Invalid LAN Queue Index	Not applicable to this object type.
	HMC Private Memory Address Too Large, Segment Descriptor Invalid, Segment Descriptor Too Small, Page Descriptor Invalid	The private memory address of the APBVT entry is reported in the PFHMC_ERRORDATA register. Packets associated with the APBVT entry are dropped. PE AEs are not reported.
Memory Region Table Entry (MRTE)	PMF Invalid, Invalid PMF Index, HMC Object Index Too Large	The index of the MRTE is reported in the PFHMC_ERRORDATA register. The receive data is dropped and the QP is put into the terminate state. A PE AE for the QP is reported.
	Invalid LAN Queue Index	Not applicable to this object type.
	HMC Private Memory Address Too Large, Segment Descriptor Invalid, Segment Descriptor Too Small, Page Descriptor Invalid	The private memory address of the MRTE object is reported in the PFHMC_ERRORDATA register. The receive data is dropped and the QP is put into the terminate state. A PE AE for the QP is reported.
Transmit FIFO	PMF Invalid, Invalid PMF Index, HMC Object Index Too Large	The index of the transmit FIFO entry is reported in the PFHMC_ERRORDATA register. The receive data is dropped and the QP is put into the terminate state. A PE AE for the QP is reported.
	Invalid LAN Queue Index	Not applicable to this object type.
	HMC Private Memory Address Too Large, Segment Descriptor Invalid, Segment Descriptor Too Small, Page Descriptor Invalid	The private memory address of the transmit FIFO entry is reported in the PFHMC_ERRORDATA register. The receive data is dropped and the QP is put into the terminate state. A PE AE for the QP is reported.

Table 38-341.HMC Errors

HMC Object	Error Type(s)	Error Behavior
Transmit FIFO Free List	PMF Invalid, Invalid PMF Index, HMC Object Index Too Large	The index of the transmit FIFO free list entry is reported in the PFHMC_ERRORDATA register. The receive data is dropped and the QP is put into the terminate state. A PE AE for the QP is reported.
	Invalid LAN Queue Index	Not applicable to this object type.
	HMC Private Memory Address Too Large, Segment Descriptor Invalid, Segment Descriptor Too Small, Page Descriptor Invalid	The private memory address of the transmit FIFO free list entry is reported in the PFHMC_ERRORDATA register. The receive data is dropped and the QP is put into the terminate state. A PE AE for the QP is reported.
Inbound RDMA Read Queue (IRRQ or Q1)	PMF Invalid, Invalid PMF Index, HMC Object Index Too Large	The index of the IRRQ entry is reported in the PFHMC_ERRORDATA register. The receive data is dropped and the QP is put into the terminate state. A PE AE for the QP is reported.
	Invalid LAN Queue Index	Not applicable to this object type.
	HMC Private Memory Address Too Large, Segment Descriptor Invalid, Segment Descriptor Too Small, Page Descriptor Invalid	The private memory address of the IRRQ entry is reported in the PFHMC_ERRORDATA register. The receive data is dropped and the QP is put into the terminate state. A PE AE for the QP is reported.
Inbound RDMA Read Queue (IRRQ or Q1) Free List	PMF Invalid, Invalid PMF Index, HMC Object Index Too Large	The index of the IRRQ free list entry is reported in the PFHMC_ERRORDATA register. The receive data is dropped and the QP is put into the terminate state. A PE AE for the QP is reported.
	Invalid LAN Queue Index	Not applicable to this object type.
	HMC Private Memory Address Too Large, Segment Descriptor Invalid, Segment Descriptor Too Small, Page Descriptor Invalid	The private memory address of the IRRQ free list entry object is reported in the PFHMC_ERRORDATA register. The receive data is dropped and the QP is put into the terminate state. A PE AE for the QP is reported.
Physical Buffer List Entry (PBLE)	PMF Invalid, Invalid PMF Index, HMC Object Index Too Large	The index of the PBLE entry is reported in the PFHMC_ERRORDATA register. The receive data is dropped and the QP is put into the terminate state. A PE AE for the QP is reported.
	Invalid LAN Queue Index	Not applicable to this object type.
	HMC Private Memory Address Too Large, Segment Descriptor Invalid, Segment Descriptor Too Small, Page Descriptor Invalid	The private memory address of the PBLE entry object is reported in the PFHMC_ERRORDATA register. The receive data is dropped and the QP is put into the terminate state. A PE AE for the QP is reported.

38.27 AQ

38.27.1 Preface

The AQ is designed with the following goals:

- Abstract firmware interface so that firmware can be changed, whether for added functionality or for firmware updates, without changing the software device driver. Further extension of the firmware enables changing parts of the hardware without modifying the software device driver.
- Remove MMIO access from all non-essential software device driver paths.
- Incorporate the VF-to-PF and function-to-function mailboxes into a single, extensible interface. Shared resources should be accessed through the AQ.
- It is possible to write one software device driver that would work both on a primary function and on a virtual function.
- A low-resource software device driver such as a pre-boot driver or an out-of-box software device driver can use the AQ for limited transmit and receive.

The 10 GbE controller provides one type of admin queue, which is used for communication between the software driver and the 10 GbE controller's firmware engine. When implemented by firmware, this queue is capable of supporting PF driver-to-VF driver communication.

The 10 GbE controller provides the following sets of AQs:

- One firmware AQ per PF, used for software-to-firmware communication, exposed in the PF memory BAR
- One firmware AQ per VF in the VF memory BAR

Assumptions:

- Currently, firmware does not maintain context for any software device driver operation, except for the state of the queue itself.
- Firmware deals with one command at a time and does not start working on a new command before finishing its current task. This might change in a future version of the firmware, so the queue mechanism must allow for it today in order to avoid the need to modify the software device driver if this happens. Firmware does however pipeline descriptor and data fetches to optimize execution latency.

38.27.2 Queue Structure

The AQ is comprised of an Admin Transmit Queue (ATQ) and an Admin Receive Queue (ARQ). Software device river commands are posted on the ATQ. Firmware completes software device driver commands by writing back onto the command descriptor. Events that are not an immediate result of a command are written to the ARQ. The software device driver posts empty buffers to the ARQ and then firmware fills them with events.

Both queues support direct and in-direct commands.An AQ direct command is one that fits entirely in the queue descriptor and while extended or an indirect command is one that uses an additional buffer, which is specified in the descriptor. When a command needs an additional external buffer it marks the BUF flag. If the buffer contains data that firmware needs to read, the RD flag is used, a buffer bigger than 512 bytes (AQ_LARGE_BUF) must have the LB flag set. The maximum buffer size supported in this version of the queues is 4096 bytes.

Both queues use the same descriptor structure. All descriptors and commands are defined using Little endian notation with 32-bit words. Software device drivers using other conventions should take care to do the proper conversions.

Table 38-342.Admin Queue Descriptor Structure (in LE 32 Order)

+3								+2								+1								+0							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
Opcode																FE	IE	SI	BUF	VFC	RD	LB						VFE	ERR	CMP	DD
Return Value / VFID																Data Len															
Cookie High																															
Cookie Low																															
Param0																															
Param1																															
Data Address High																															
Data Address Low																															

Table 38-343.Admin Descriptor Field descriptions

Name	Bytes.Bits	Description
Flags.DD	0.0	Set by firmware to mark entry done.
Flags.CMP	0.1	Set by firmware to mark entry as completion.
Flags.ERR	0.2	Set by firmware to mark entry as an error indication.
Flags.VFE	0.3	Set by firmware to mark entry as an event forwarded from a VF driver.
Flags.Reserved	0.4-1.0	Reserved, must be zeroed by sender and ignored by receiver.
Flags.LB	1.1	Set by software device driver to indicate that indirect buffer is longer than AQ_LARGE_BUF.
Flags.RD	1.2	Set by software device driver to indicate that firmware needs to read indirect buffer.
Flags.VFC	1.3	Set by software device driver to indicate command on behalf of a VF.
Flags.BUF	1.4	This command uses additional data.
Flags.SI	1.5	Do not interrupt when this command completes.
Flags.EI	1.6	Interrupt on error (supersedes Flags.SI in case of an error).
Flags.FE	1.7	If previous command completed in error, flush this one.
Opcode	2-3	Command opcode (see Table 38-354).
Datalen	4-5	Indirect data length (can be used for other uses if Flags.RD is not set).
Return Value/ VFID	6-7	Return value / VF ID for command or event, only sent by firmware or PF driver. See Section 38.27.8 and Table 38-353.
Cookie High	8-11	Opaque data, echoed by receiver, high half.
Cookie Low	12-15	Opaque data, echoed by receiver, low half.
Param0	16-19	First general use parameter.
Param1	20-23	Second general use parameter.
Data Address High	24-27	Indirect data pointer (can be used for other uses if Flags.RD is unset).
Data Address Low	28-31	Indirect data pointer (can be used for other uses if Flags.RD is unset).

See Section 38.27.5 for details on the different command types

38.27.2.1 AQ CSRs

AQs have 32 byte descriptors. They are serviced by the following registers (Table 38-344).

{PFX} denotes the register scope prefix:

- for VFs it is VF_
- for PFs it is PF_

Except for the name prefix the PF and VF registers are exactly the same.

Table 38-344.AQ Registers

Name	Width	Comment
{PFX}ATQBAH	32 bits	High bytes of ATQ base address.
{PFX}ATQBAL	32 bits	Low bytes of ATQ base address, address must be 64-byte aligned.
{PFX}ATQLEN	10+4 bits	ATQ length in descriptors, MSB is set for queue enable, three error bits (critical error, overflow error and VF error) are set by FW to indicate error conditions (see further Section 38.27.9.1).
{PFX}ATQH	10 bits	ATQ head pointer (firmware updates).
{PFX}ATQT	10 bits	ATQ tail pointer (software device driver updates).
{PFX}ARQBAH	32 bits	High bytes of ARQ base address.

Table 38-344.AQ Registers

Name	Width	Comment
{PFX}ARQBAL	32 bits	Low bytes of ARQ base address, address must be 64-byte aligned.
{PFX}ARQLEN	10+4 bits	ATQ length in descriptors, MSB is set for queue enable, three error bits (critical error, overflow error and VF error) are set by firmware to indicate error conditions (see further Section 38.27.9.1).
{PFX}ARQH	10 bits	ARQ head pointer (firmware updates).
{PFX}ARQT	10 bits	ARQ tail pointer (software device driver updates).

38.27.2.2 AQ Interrupts

Firmware triggers an interrupt when it completes descriptors on the transmit queue or when it posts events to the receive queue. The interrupt triggered is the other interrupt and the *ADMINQ* cause bit is set to signals that this was the reason.

38.27.3 Initialization

when initializing the queue, the software device driver must do the following:

- The software device driver allocates and set ups appropriately sized host memory for the queues.
- The software device driver must post initialized buffers to the receive queue before it can use the transmit queue (see Section 38.27.3.1 for receive queue element initialization).
- The software device driver clears the head and tail registers for each queue ({PFX}ATQH and {PFX}ARQH. Tail registers are {PFX}ATQT and {PFX}AROT.
- The software device driver then programs the base, and length registers for each queue ({PFX}ATQBAL, {PFX}ATQBAH, {PFX}ATQLEN, {PFX}ARQBAL, {PFX}ARQBAH and {PFX}ARQLEN) and sets *length.enable* to 1b to signal firmware that the queue is now enabled.
- For a firmware AQ, the software device driver must then issue a Get Version admin command and check the queue and firmware major version numbers before it can use the queue for anything else. If the major versions does not match what the software device driver expects, it reports the mismatch and fails to load. For details and current queue version number (see Section 38.27.11.1).
- For a firmware AQ, after the software device driver has verified the queue version, it sends a Driver Version command to the 10 GbE controller. The 10 GbE controller then sends an indication to the BMC that the PF driver is present.
- A PF driver must have at least one buffer posted to the receive queue for each VF that is currently running. Note that this can never be greater than the number of logical cores in the system. This must be done before enabling VFs.
- A PF might disable a VF transmit and receive admin queues by clearing the matching GL_VF_CTRL_TX[n].AQ_TX_EN bit or GL_VF_CTRL_RX[n].AQ_RX_EN bit for Tx or Rx admin queue commands.

38.27.3.1 Receive Queue Element Initialization by Software Device Driver

The software device driver must clear any unused fields (including unused flags) and set data pointers and data length to a mapped DMA pointer.

The software device driver might set the SI and the EI flags in the receive queue element. The software device driver must not set the FE flag on receive queue elements.

Table 38-345.Receive Queue Element - Initial Values

Name	Bytes.Bits	Value	Remarks
Flags.DD	0.0	0	Software device driver must clear.
Flags.CMP	0.1	0	Software device driver must clear.
Flags.ERR	0.2	0	Software device driver must clear.
Flags.VFE	0.3	0	Software device driver must clear.
Flags.Reserved	0.4 1.0	0	Reserved, must be zeroed by sender and ignored by receiver.
Flags.LB	1.1	0 or 1	Set by the software device driver if buffer is longer than AQ_LARGE_BUF (512 bytes).
Flags.RD	1.2	0	Not applicable to receive queue.
Flags.VFC	1.3	0	Software device driver must clear.
Flags.BUF	1.4	1	Receive queue elements always have an additional buffer.
Flags.SI	1.5	Driver Might Set	Do not interrupt when this command completes.
Flags.EI	1.6	Driver Might Set	Interrupt on error - supersedes Flags.SI in case of error.
Flags.FE	1.7	0	Software device driver must clear.
Opcode	2-3		Software device driver must clear.
Datalen	4-5	Buffer Len	Additional data length.
Return Value/ VFID	6-7		Software device driver must clear.
Cookie High	8-11		Software device driver must clear.
Cookie Low	12-15		Software device driver must clear.
Param0	16-19		Software device driver must clear.
Param1	20-23		Software device driver must clear.
Data Address High	24-27	Buffer ADDR	Indirect data pointer (can be used for other uses if Flags.RD is not set).
Data Address Low	28-31	Buffer ADDR	Indirect data pointer (can be used for other uses if Flags.RD is not set).

The values written by firmware when it uses the EAQ element are discussed in the following paragraphs.

38.27.4 Driver Unload and Queue Shutdown

When shutting down the firmware AQ the software device driver (both for PF and VF):

- Posts a Queue Shutdown admin command (0x0003). See Section 38.27.11.3 for more detail. In this command, the software device driver sets the Driver Unloading flag if it intends to unload.
- Software must not send any additional commands on the queue until the flow is completes.

Firmware:

- Waits for any pending DMA transactions from firmware to be acknowledged by hardware.
- Closes the Rx queue by clearing its *Enable* bit.
- Sends a completion to the software device driver as usual, honoring all the interrupt control bits in the descriptor.
- Software then closes the Tx queue by clearing its *Enable* bit.

If the software device driver is unloading, it issues a function reset (PFR or VFR) to the 10 GbE controller.

If the software device driver is unloading, firmware signals the MC.

- Only for a PF driver.

Note that before the firmware AQs are re-enabled, software should clear the head and tail registers of the transmit and receive firmware AQ.

38.27.5 Commands Description

38.27.5.1 Direct Command

38.27.5.1.1 Direct Admin Command

The template for a command that is fully contained in the descriptor and does not need an additional data buffer.

Table 38-346.Direct Admin Command Structure

Name	Bytes.Bits	Value	Remarks
Flags.DD	0.0	0	Software device driver must clear.
Flags.CMP	0.1	0	Software device driver must clear.
Flags.ERR	0.2	0	Software device driver must clear.
Flags.VFE	0.3	0	Software device driver must clear.
Flags.Reserved	0.4-1.0	0	Reserved, must be zeroed by sender and ignored by receiver.
Flags.LB	1.1	0	A direct command has no additional buffer.
Flags.RD	1.2	0	A direct command has no additional buffer.
Flags.VFC	1.3	0	Software device driver must clear.
Flags.BUF	1.4	0	A direct command does not have an additional write buffer.
Flags.SI	1.5	Driver Might Set	Do not interrupt when this command completes.
Flags.EI	1.6	Driver Might Set	Interrupt on error (supersedes Flags.SI in case of error).
Flags.FE	1.7	Driver Might Set	If set, the command is flushed if the preceding command resulted in an error.
Opcode	2-3	Opcode	Command opcode (see Table 38-354).
Datalen	4-5	0	
Return Value/ VFID	6-7		
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Param0	16-19		First command parameter.
Param1	20-23		Second command parameter.
Data Address High	24-27		Can be used for an additional command parameter.
Data Address low	28-31		Can be used for an additional command parameter.

38.27.5.1.2 Direct Command Completion

Table 38-347.Direct Command Completion Event Template

Name	Bytes.Bits	Value	Remarks
Flags.DD	0.0	1	Firmware must set.
Flags.CMP	0.1	1	Firmware must set.
Flags.ERR	0.2	0 or 1	Firmware must set only if it reporting an error.
Flags.VFE	0.3	0	Firmware must set.
Flags.Reserved	0.4 -1.0	0	Reserved, must be zeroed by sender and ignored by receiver.
Flags.LB	1.1	Echo	Firmware copies the value from the command.
Flags.RD	1.2	Echo	Firmware copies the value from the command.
Flags.VFC	1.3	Echo	Firmware copies the value from the command.
Flags.BUF	1.4	Echo	Firmware copies the value from the command.
Flags.SI	1.5	Echo	Firmware copies the value from the command.
Flags.EI	1.6	Echo	Firmware copies the value from the command.
Flags.FE	1.7	Echo	Firmware copies the value from the command.
Opcode	2-3	Opcode	Command opcode (see Table 38-354).
Datalen	4-5		Can be used for an additional command parameter.
Return Value/ VFID	6-7		Firmware returns value 0 = no error (for error codes see Table 38-353).
Cookie High	8-11	Echo	Opaque value is copied by firmware from the command.
Cookie Low	12-15	Echo	Opaque value is copied by firmware from the command.
Param0	16-19		First command parameter.
Param1	20-23		Second command parameter.
Data Address High	24-27		Can be used for an additional command parameter.
Data Address Low	28-31		Can be used for an additional command parameter.

38.27.5.2 Indirect Command

38.27.5.2.1 Indirect Admin Command

An indirect write command uses an additional DMA buffer specified in the descriptor.

The *BUF* flag must be set by the software device driver. If the buffer is larger than 512 bytes the *LB* flag must be set.

This version of the queue is limited to buffers up to 4096 bytes. If the command uses the buffer to pass data, the *RD* flag needs to be set.

Table 38-348.Indirect Command Template

Name	Bytes.Bits	Value	Remarks
Flags.DD	0.0	0	Software device driver must clear.
Flags.CMP	0.1	0	Software device driver must clear.
Flags.ERR	0.2	0	Software device driver must clear.
Flags.VFE	0.3	0	Software device driver must clear.
Flags.Reserved	0.4-1.0	0	Reserved, must be zeroed by sender and ignored by receiver.
Flags.LB	1.1	0 or 1	Set by the software device driver if the buffer is longer than AQ_LARGE_BUF (512).
Flags.RD	1.2	0 or 1	Set if additional buffer has command parameters.

Table 38-348.Indirect Command Template (Continued)

Name	Bytes.Bits	Value	Remarks
Flags.VFC	1.3	0	Software device driver must clear.
Flags.BUF	1.4	0 or 1	Software device driver must set this flag on an indirect command.
Flags.SI	1.5	Driver May set	Do not interrupt when this command completes.
Flags.EI	1.6	Driver May set	Interrupt on error (supersedes Flags.SI in case of error).
Flags.FE	1.7	Driver May set	If set, the command is flushed if the preceding command resulted in an error.
Opcode	2-3	Opcode	Command opcode (see Table 38-354).
Datalen	4-5	Buffer len	Usable length of additional buffer.
Return value/VFID	6-7		
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Param0	16-19		First command parameter.
Param1	20-23		Second command parameter.
Data Address high	24-27	Buff Addr	High bits of buffer address.
Data Address low	28-31	Buff Addr	Low bits of buffer address.

38.27.5.2.2 Indirect Command Completion

When completing an indirect command, firmware overwrites the data length with the actual length of data returned by the command.

Table 38-349.Indirect Command Completion

Name	Bytes.Bits	Value	Remarks
Flags.DD	0.0	1	Firmware must set.
Flags.CMP	0.1	1	Firmware must set.
Flags.ERR	0.2	0 or 1	Firmware must set only if it reporting an error.
Flags.VFE	0.3	0	Firmware must clear.
Flags.Reserved	0.4 -1.0	0	Reserved, must be zeroed by sender and ignored by receiver.
Flags.LB	1.1	Echo	Firmware copies value from command.
Flags.RD	1.2	Echo	Firmware copies value from command.
Flags.VFC	1.3	Echo	Firmware copies value from command.
Flags.BUF	1.4	Echo	Firmware copies value from command.
Flags.SI	1.5	Echo	Firmware copies value from command.
Flags.EI	1.6	Echo	Firmware copies value from command.
Flags.FE	1.7	Echo	Firmware copies value from command.
Opcode	2-3	Opcode	Command opcode (see Table 38-354).
Datalen	4-5		Can be used for an additional command parameter.
Return Value/ VFID	6-7		Firmware returns a value of 0 = no error (for error codes see Table 38-353).
Cookie High	8-11	Echo	Opaque value is copied by firmware from the command.
Cookie Low	12-15	Echo	Opaque value is copied by firmware from the command.
Param0	16-19		First command parameter.
Param1	20-23		Second command parameter.
Data Address High	24-27		Can be used for an additional command parameter.

Table 38-349.Indirect Command Completion

Name	Bytes.Bits	Value	Remarks
Data Address Low	28-31		Can be used for an additional command parameter.

38.27.6 Command Fetch and Verification

When a command posts, firmware looks it up in an internal permission table to decide if the request should be honored. Possible actions are:

- Allow — Firmware acts upon the command.
- Forward — Firmware halts the queue and forward the command to the PF driver, a completion for this command is initiated by the PF driver when it finishes it, only then further processing of commands from this queue is allowed. The PF driver needs to re-enable the queue after it deals with it.
- Error — Firmware completes the action by returning the error specified in the table.
- Drop — Firmware behaves as if the command succeeded but does nothing.

A VF driver is only allowed to post the commands listed in Table 38-350. If any other command is posted by a VF, firmware returns the EPERM (operation not permitted) error code.

Table 38-350.Commands Allowed for VF Firmware Admin Queues

Opcode	Command
0x0001	Get version.
0x0801	Send to PF.
0x0B02	Set RSS key.
0x0B03	Set RSS LUT.
0x0B04	Get RSS key.
0x0B05	Get RSS LUT.

38.27.7 Non-Completion Events

Events that are not an immediate result of a command completion, are posted by the FW onto the receive queue.

Table 38-351 lists the currently defined events, note that whenever possible the same number is used for the opcode that generates the event and for the event ID.

Table 38-351.Non-Completion Events

Name	Opcode	Reference	Type
Send to PF	0x0801	Section 38.27.12.1	Indirect/Direct
Send to VF	0x0802	Section 38.27.12.2	Indirect/Direct
Send to Peer PF	0x0803	Section 38.28	Indirect/Direct
Send to Peer Driver	0x804		Indirect/Direct

38.27.8 Error Handling

When firmware encounters an error it uses the return value field to indicate the type of error. The error code is comprised of two bytes: the lower byte is a user-visible code from Table 38-353, the higher byte can be used by firmware to report internal state or debug information. Note that the software device driver must log this information;

however, firmware might change this value from release to release. It is not to be reported to the user and no other action is to be taken upon this data. when the return value is zero (no error) firmware must not set the high byte. In the event that the queue itself is inaccessible, firmware overwrites the queue base addresses with an error code and clears the enable bit of both queues.

Table 38-352.Admin Queue Return Value Fields

Name	Bytes	comment
Code	0:7	Return value from Table 38-353.
Firmware Internal Code	8:15	Firmware internal code, must be zero if the *Code* field is zero.

38.27.9 Error dodes

When firmware completes a command, it must use the following error codes.

Table 38-353.Admin Queue Return Values and Error Codes

Error Code	Value	Meaning
(No Error)	0	No error (success).
EPERM	1	Operation not permitted.
ENOENT	2	No such element.
ESRCH	3	Bad opcode.
EINTR	4	Operation interrupted.
EIO	5	I/O error.
ENXIO	6	No such resource.
E2BIG	7	Arg too long.
EAGAIN	8	Try again.
ENOMEM	9	Out of memory.
EACCES	10	Permission denied.
EFAULT	11	Bad address.
EBUSY	12	Device or resource busy.
EEXIST	13	Attempt to crate something that exists.
EINVAL	14	Invalid argument.
ENOTTY	15	Not a typewriter.
ENOSPC	16	No space left or allocation failure.
ENOSYS	17	Function not implemented.
ERANGE	18	Parameter out of range.
EFLUSHED	19	Command flushed because a previous command completed in error.
BAD_ADDR	20	Internal error, descriptor contains a bad pointer.
EMODE	21	Operation not allowed in current device mode.
EFBIG	22	File too big.

38.27.9.1 Critical Error Indication

- On any error that prevents data placement to a queue ATQLEN.ATQCRIT (or ARQLEN.ARQCRIT) bit are set by firmware and the queue is stopped (by clearing its enable bit), then an interrupt is sent by firmware to the software device driver.
- Software reads and reports the error code and then resets the queue.

- If an overflow occurs and a message to the queue is dropped because the queue is full, firmware sets ARQLEN.ARQOVFL and interrupts the software device driver. Firmware does not stop the queue since, depending on the software device driver mode, this might be a recoverable error.

Note: This error can currently only happen on the receive queue, but to simplify the hardware design, the bit is present on both queues.

- When a VF has an event that causes firmware to set an error bit in its queue, firmware sets ATQLEN.ATQVFE in the corresponding PFs queue and interrupts it. Firmware does not stop the PF queue in this case.

Note: Events and completions that have already been posted before the error are still readable and can be handled by software.

38.27.10 Command Opcodes

Opcodes are 16 bits of which the upper 8 bits designate the group of the opcode and the lower 8 bits are the command in the group. Each group is described in the relevant sections that follow.

Table 38-354.opcode groups

Name	Opcodes
Generic	0x00XX
Mac Address	0x0100
PXE Mode	0x0110
WoL	0x012X
Switch	0x02XX
DCB	0x03XX
Scheduler	0x04XX
HMC	0x05XX
Link	0x06XX
NVM	0x07XX
Virtualization	0x08XX
Alternate Structure	0x09XX
LLDP	0x0AXX
Receive Filters	0x0BXX
Profiles Handling	0x0C40:0x0C5F

38.27.11 Generic Firmware Admin Commands

Table 38-355.Generic Commands

Name	Opcode	Reference	Type
Get Ver	0x0001	Section 38.27.11.1	Direct
Driver Version	0x0002	Section 38.27.11.2	Direct
Queue Shutdown	0x0003	Section 38.27.11.3	Direct
Set PF Context	0x0004	Section 38.27.11.4	Direct
Get Expanded AQ Error Reason	0x0005		Direct
Request Resource Ownership	0x0008	Section 38.27.11.5	Direct

Table 38-355.Generic Commands

Name	Opcode	Reference	Type
Release Resource Ownership	0x0009	Section 38.27.11.6	Direct
Discover Function Capabilities	0x000A	Section 38.27.11.7	Indirect
Discover Device Capabilities	0x000B	Section 38.27.11.7	Indirect
CSR Access	0x0206		Direct

38.27.11.1 Get Version Admin Command

This must be the first command that the software device driver issues before it can use the queue for other purposes. The software device driver must inspect the reply to ensure that the firmware version is compatible. If firmware is still initializing it might delay response until it completes. Both firmware and the API have two unassigned 16-bit values as a minor and major version. The software device driver must not continue loading if the major version is a mismatch. Minor versions are for tracking changes that do not need software device driver modifications.

Table 38-356.Get Version Command (Opcode: 0x0001)

Name	Bytes.Bits	Value	Remarks
Flags	1-0		See AQ section for details.
Opcode	2-3	0x0001	Command opcode.
Datalen	4-5	0	no external response buffer.
Return Value/ VFID	6-7		Return value. Zeroed by the software device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
ROM Build ID	16-19		Device ROM build version.
Firmware Build ID	20-23		Device Firmware build version.
AQ API Patch Version	31		AQ API patch version (unsigned 8-bit integer).
Firmware Major Version	24-25		Major firmware version (unsigned 16-bit integer).
Firmware minor Version	26-27		Minor firmware version (unsigned 16-bit integer).
API Major Version	28-29	1	Major queue version (unsigned 16-bit integer). One for this version of the queues.
API Minor Version	31-32	1	Minor queue version (unsigned 16-bit integer). One for this version of the queues.

38.27.11.2 Driver Version Indirect Admin Command

Table 38-357.Driver Version Command (Opcode: 0x0002) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	1-0		See AQ section for details.
Opcode	2-3	0x0002	Command opcode
Datalen	4-5		Buffer length. Can be up to 32.
Return Value/ VFID	6-7		Return value. Zeroed by the software device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.

Table 38-357.Driver Version Command (Opcode: 0x0002) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Driver Version	16-19	version	byte 16 = major version. byte 17 = minor version. byte 18 = build version. byte 19 = sub-build version.
Reserved	20-23	0x0	Reserved.
Data Address High	24-27		Address of response buffer.
Data Address Low	28-31		Address of response buffer.

Table 38-358.Driver Version Buffer

Name	Length	Description
Driver Version	Datalen	Driver string (not null terminated) as reported by the software device driver.

38.27.11.3 Queue Shutdown Command

This is the final command posted to the queue, closing the queue as described in Section 38.27.4. When this command completes the software device driver is allowed to free any host resources associated with the firmware AQ.

If the software device driver is going to unload it must set the *Driver Unloading* flag to signal firmware.

Once this command is posted, the software device driver is not allowed to issue any more commands on the queue before a reset is done.

Note: Interrupt generation and the interrupt control flags in this command are handled as usual by firmware. This means that if an interrupt was not inhibited by setting the *SI* flag it happens. If the software device driver is in polling mode and cannot handle an interrupt, it needs to either inhibit the interrupt or have interrupts disabled through the interrupt control registers.

Table 38-359.Queue Shutdown Command (Opcode: 0x0003)

Name	Bytes.Bits	Value	Remarks
Flags	1-0		See AQ section for details.
Opcode	2-3	0x0003	Command opcode.
Datalen	4-5	0	No external data.
Return Value/VFID	6-7		Return value. Zeroed by the software device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Driver Unloading	16.0		Set to one if the software device driver intends to unload. Zero otherwise.
Reserved	17-31	0x0	Reserved.

38.27.11.4 Set PF Context

This admin command is used to set an explicit PF ID number initiated from the Tools firmware AQ. It is needed for some admin commands that requires control on a specific PF context regardless of the PF from which the admin command is initiated.

Table 38-360.Set PF Context Command (Opcode: 0x0004)

Name	Bytes.Bits	Value	Remarks
Flags	1-0		See AQ section for details.
Opcode	2-3	0x0003	Command opcode.
Datalen	4-5	0	No external data.
Return Value/VFID	6-7		Return value. Zeroed by the software device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
PF ID	16		PF ID.
Reserved	17-31	0x0	Reserved.

38.27.11.5 Request Resource Ownership Admin Command

This command is used by the software device driver to request ownership of a shared resource. It specifies the resource and the type of access it requests (see Table 38-361). On success, the command returns a value of zero. The completion specifies the maximum time in ms that the software device driver might hold the resource in the *Timeout* field. The software device driver must free the resource before that time. It must also free a resource before asking for it again. A request for a resource that is already held by th software device driver fails with an EACCESS error code. A timeout value of zero means no timeout.

If the resource is held by another resource, the command completes with a busy return value and the *Timeout* field indicates the maximum time the current owner of the resource has to free it.

Firmware implements a timeout mechanism taking back the ownership if the software device driver hangs. Any further commands by this software device driver that attempt to access this resource fails with an EPERM error code until the software device driver frees the resource and requests it again.

Table 38-361.Shared Resources

Resource	ID	Supported Modes	Default Timeout
NVM	0x0001	1 = Read. 2 = Write.	3000ms for read. 180000ms for write.
SDP	0x0002	1 = Read. 2 = Write.	Zero (no timeout).

Table 38-362.Request Resource Ownership Admin Command (Opcode: 0x0008) (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ section for details.
Opcode	2-3	0x0008	Command opcode.
Datalen	4-5	0	No external buffer for this command.
Return Value/ VFID	6-7		Return value. Zeroed by the software device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.

Table 38-362.Request Resource Ownership Admin Command (Opcode: 0x0008) (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Resource ID	16-17	ID	See Table 38-361 for list of IDs.
Access Type	18-19	Type	See Table 38-361 for list.
Timeout	20-23	Timeout	Timeout in ms (written by firmware).
Resource Number	24-27	Number	This is the ID pin for an SDP.
Reserved	28-31		Reserved.

38.27.11.6 Release Shared Resource Admin Command

This command is used to return ownership of a shared resource to firmware. The software device driver specifies the ID of the resource it is releasing in the ID field.

Table 38-363.Release Resource Ownership Admin Command (Opcode: 0x0009)

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ section for details.
Opcode	2-3	0x0009	Command opcode.
Datalen	4-5	0	No external buffer for this command.
Return Value/ VFID	6-7		Return value. Zeroed by the software device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Resource ID	16-17	ID	See Table 38-361 for list of IDs.
Reserved	18-23	Reserved	Reserved.
Resource Number	24-27	Number	If a resource number was specified in the request, it needs to be specified here too.
Reserved	28-31	Reserved	Reserved.

38.27.11.7 Discover Device/function Capabilities

This command is used to request the list of capabilities of the device or the function. Firmware fills in the capabilities structure and returns the length to the software device driver. If the buffer size is not big enough for the entire structure, firmware returns ENOMEM and sets the length to the size of the structure (firmware does not fill in a partial structure). This can be used to discover the structure size. Capabilities are described using the structure listed in Table 38-365. This list of resources recognized by this version of the command are listed in Table 38-366. Additional capabilities can be retrieved using the Get PHY Abilities (0x0600) command and the Get Switch Resource Allocation (0x0202) command.

Table 38-364.Discover Device/function Capabilities (Opcode: 0x000A, 0x000B)

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ section for details.
Opcode	2-3		Command opcode.
Datalen	4-5	0x0	Must be zero. Value is ignored.

Table 38-364.Discover Device/function Capabilities (Opcode: 0x000A, 0x000B)

Name	Bytes.Bits	Value	Remarks
Return Value/VFID	6-7		Return value. Zeroed by software device driver. Written by firmware. 0x0 = Command success. EPERM is returned if the command was called while not in PXE mode.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Operation Mode	16.0		Defines the operation mode of firmware 0b = Normal operation mode. 1b = NVM recovery mode.
Reserved	16.1-19	Reserved	Reserved.
Cap Count	20-23		Number of capability records returned. Zeroed by the software device driver and written by firmware).
Data Address High	24-27	0x0	Address of response buffer.
Data Address Low	28-31	0x0	Address of response buffer.

Table 38-365.Capability Structure

Name	Length	Remarks.
Capability	16 bits	See Table 38-366 for a list of capabilities.
Major Version	8 bits	
Minor Version	8 bits	
Number	32 bits	Number of resources described by this capability.
Logical ID	32 bits	Only meaningful for some types of resources.
Physical ID	32 bits	Only meaningful for some types of resources.
Reserved1	64 bits	Reserved.
Reserved2	64 bits	Reserved.

Table 38-366.Resources Recognized by This Version of the Command

Name	Capability	Ver	Number	Logical ID	Physical ID
Switching Mode	0x0001	1.0	Returns the switching mode supported: 0 = EVB switching (including cloud). Other = Reserved.		
Manageability Mode	0x0002	1.0	Returns the value of the *Manageability Mode* field in the NVM.		
OS2BMC Capable	0x0004	1.0	Returns the value of the *OS2BMC Capable* field in the NVM.		
Functions Valid	0x0005	1.0	Bits 15:0 each corresponds to a PFPCI_STATUS1.FUNC_VALID bit for PFs 0,…,15.		
Alternate RAM Structure	0x0006	1.0	One.		

Table 38-366.Resources Recognized by This Version of the Command

Name	Capability	Ver	Number	Logical ID	Physical ID
WoL and Proxy Support	0x0008	1.0	Number of ACPI supported filters = 8 if supported, 0 otherwise. Same value for all ports and for the device command. For the 10 GbE controller, return the total number of filters across all supporting functions.	SEID of WoL and proxy VSI of the port. N/A if programming method is hardware. N/A for the 10 GbE controller.	Bit0 = APM WoL is supported (based on PFPM_APM.APME bit for device command or of all the PFs bits. Bit 1 = ACPI programming method (relevant only if number of ACPI supported filters is not zero). 1 = AQC. Bit 2: Proxy support. 0 = disabled 1 = enabled for device command or of all the PFs bits. Other bits = reserved
SR-IOV	0x0012	1.0	Set to one if enabled in config space. For the 10 GbE controller, should be set if set in any of the functions.	SR-IOV version (1.1)	
Virtual Function	0x0013	1.0	Function: Number of allocated VFs. Device: Total number of VFs exposed to all functions.	Logical ID of first VF	
VMDq	0x0014	1.0	Set to one.		
802.1Qbg	0x0015	1.0	One if enabled.		
VSI	0x0017	1.0	Function: Number of guaranteed VSI as read from PF allocations structure for PFs. Device: Number of VSIs allocated to the host (not including EMP VSIs).		
DCB	0x0018	1.0	One if enabled.	Device = 0: Bitmap of active TCs.	Max number of TCs 8.
Reserved	0x0021	1.0	Reserved.		
iSCSI	0x0022	1.0	0x1 if iSCSI is enabled. 0x0 if not enabled. For device capability it is always set, and for a function capability it is a reflection of the PFGEN_STATE.PFSCEN flag.		
RSS	0x0040	1.0	Table size 512 for PFs. 64 for VFs.	Entry width in bits: 6 for PFs. 4 for VFs.	
Rx Queues	0x0041	1.0	Function: Number of queues allocated to the PF. Device: Total number of queues available to the device.		ID of first queue.
Tx Queues	0x0042	1.0	Function: Number of queues. Device: Total number of queues available.		ID of first queue.

Table 38-366.Resources Recognized by This Version of the Command

Name	Capability	Ver	Number	Logical ID	Physical ID
MSI-X	0x0043	1.0	Number of vectors.		
VF-MSIX	0x0044	1.0	Number of MSIX Vectors available to VFs of this PF.		
Flow Director	0x0045	1.0	Function: Number of filters guaranteed to this PF. Device: Number of filters available in device (8192).	Function: Number of best effort filters. Device: Number of filters available in device (8192).	
1588	0x0046	1.1	Function: TimeSync is enabled on the port on which this function runs. Device: TimeSync-enabled on any of the ports (IPRTTSYN_CTL.TSYNENA).	Function: N/A. Device: A bitmap of the enabled ports.	
MaxMTU	0x47	1.0	Function: Max MTU of the function. Device: Max MTU of hardware (9728).		
NVM Versions[1]	0x48	1.0	1st word - NVM address. 2nd word - NVM value.	1st word - NVM address. 2nd word - NVM value.	1st word - NVM address. 2nd word - NVM value.
iWARP	0x0051	1.0	One if enabled.		
LED[2]	0x0061	1.0	Always one.	Action.	Pin number (GPIO index).
SDP[3]	0x0062	1.0	Always one.	Action.	Pin number (GPIO index).
MDIO[4]	0x0063	1.0	One if enabled.	Mode: 0x0 = MDIO shared. 0x1 = MDIO dedicated. 0x2 = I^2C.	The interface used to control this port.
WR_CSR_PROT	0x0064	1.0	Bytes 0 … 3 of the WR_CSR_PROT parameter. LS byte is byte 0 of the field.	Bytes 4 … 7 of the WR_CSR_PROT parameter. MS byte is byte 7 of the field.	
Drop/No-drop Policy	0x0065	1.0	Returns the value of the PXE Mode No-Drop Policy Supported NVM bit.	• 0x1, if no-drop policy is enabled on the port used by the current PF. • 0x0, if no-drop policy is not enabled on the port used by the current PF.	
Number of Ports in Switch Mode	0x0072	1.0	Returns the number of ports in switch mode.		

Notes:

1. The NVM versions capability is used to return NVM version data to software in a centralized location. It appears multiple times, reporting three NVM versions per entry. The following NVM words are returned as versions: 0x18-0x2E (inclusive), 0x34 and 0x35.
2. Repeat this entry for each assigned LED. If device = 1, returns an entry for each LED in the device.
3. Repeat this entry for each assigned pin. If device = 1, returns an entry for each SDP in the device.
4. This entry is not relevant for device = 1 and is not returned.

Note: Fields that are not applicable (empty in the table) are set to zero by firmware.

Note: When device capabilities are requested, the total numbers for the entire device should be returned.

Table 38-367.Configure No-drop Policy (Opcode: 0x0112)

Name	Bytes.Bits	Value	Remarks
Flags	1:0	0x0	See Section 38.27.5.2.1 for details.
Opcode	2-3	Opcode	Command opcode.
Datalen	4-5	0x0	Must be 0x0, value is ignored.
Return Value/ VFID	6-7		Return value. Zeroed by device driver. Written by firmware. 0x0 = Command success. EPERM returned if the command was called while not in PXE mode. EPERM returned if software control of the drop/no-drop policy is locked by NVM setting.
Cookie High	8-11	Cookie	Opaque value. Copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value. Copied by firmware into the completion of this command.
Force No-drop	16.0	0x0	Sets the No-drop/Drop Policy. 1b = Force no-drop. 0b = Remove the force and return to normal firmware behavior.
Reserved	16.1-31	0x0	Reserved.

38.27.12 Virtualization Admin Commands

Table 38-368.Virtualization Admin Commands

Name	Opcode	Ref	Type
Send to PF	0x0801	Section 38.27.12.1	Indirect /Direct
Send to VF	0x0802	Section 38.27.12.2	Indirect /Direct
Send to peer	0x0803	Section 38.28	Indirect /Direct

38.27.12.1 Send Message to PF Admin Command

This command, together with the next one, implements a communication channel between PFs and their VFs. The data in the external buffer is copied into an event on the PF receive queue. The command completes once the data is copied. The contents of the messages is defined by software.

Since the value of the cookie is copied to the event, if 8 bytes are enough for needed message, the software device driver might specify a length of zero and not use an external buffer. In this case, it should not set the *BUF* flag.

Note: This version of the queues supports messages of up to 4096 bytes.

Table 38-369.Send to PF Command (Opcode: 0x0801)

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ section for details.
Opcode	2-3	0x0801	Command opcode.
Length	4-5	buffer length	Length of message.
Return Value/ VFID	6-7		Return value. Zeroed by the software device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.

Table 38-369.Send to PF Command (Opcode: 0x0801) (Continued)

Name	Bytes.Bits	Value	Remarks
Reserved	16-19		
Reserved	20-23		
Data Address High	24-27	0x0	Address of data buffer.
Data Address Low	28-31	0x0	Address of data buffer.

After posting the event to the PF admin receive queue, firmware completes this command by updating the flags and returning a value of zero for success.

Table 38-370.Message From VF Event (Opcode: 0x0801)

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See Section 38.27.5.1.1 for details.
Opcode	2-3	0x0801	Event code.
Length	4-5	Buffer Length	Length of message.
Return Value/ VFID	6-7	VFID	ID of sending VF.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Reserved	16-19		Reserved.
Reserved	20-23		Reserved.
Data Address High	24-27	0x0	Address of data buffer.
Data Address Low	28-31	0x0	Address of data buffer.

38.27.12.2 Send Message To Vf admin command

This command, together with the previous one, implements a communication channel between PFs and their FVs. The data in the external buffer is copied into an event on the VF receive queue. The command completes once the data is copied. The contents of the messages are defined by software. A PF can only send messages to one of it's VFs.

Since the value of the cookie is copied to the event, if 8 bytes are enough for needed message, the software device driver might specify a length of zero and not use an external buffer. in this case, it should not set the *BUF* flag.

Note: This version of the queues supports messages of up to 4096 bytes.

Table 38-371.Send to VF Command (Opcode: 0x0802)

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ section for details.
Opcode	2-3	0x0802	Command opcode.
Length	4-5	buffer length	Length of message.
Return Value	6-7		Return value. Zeroed by the software device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
VFID	16-19	VFID	ID of VF.
Reserved	20-23		
Data Address High	24-27	0x0	Address of data buffer.

Table 38-371.Send to VF Command (Opcode: 0x0802) (Continued)

Name	Bytes.Bits	Value	Remarks
Data Address Low	28-31	0x0	Address of data buffer.

After posting the event to the VF admin receive queue, firmware completes this command by updating the flags and returning a value of zero for success.

Table 38-372.Message From PF Event (Opcode: 0x0802)

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ section for details.
Opcode	2-3	0x0802	Event code.
Length	4-5	Buffer Length	Length of message.
Return Value/ VFID	6-7		Reserved.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Reserved	16-19		Reserved.
Reserved	20-23		Reserved.
Data Address High	24-27	0x0	Address of data buffer.
Data Address Low	28-31	0x0	Address of data buffer.

38.28 Statistics

The 10 GbE controller provides statistics for the following interfaces:

- Physical ports
- Virtual switch elements VEBs, PVs, and VSIs
- PE statistics, some of which are global and some are per VSI
- Flow director statistics
- Host interface statistics

A minimal set of Ethernet interface group statistics (RFC 2863) is provided by the 10 GbE controller at the switch sampling points, a fuller set of statistics is provided for physical ports.

The prefixes for the statistics counter names are listed in Table 38-373.

Table 38-373.Counter Name Prefixes

Element	Register Prefix	Instances	Remark
Port	GLPRT	4	
VEPA or VEB	GLSW	16	
VEB per VLAN	GLVEBVL	128	
VEB per TC	GLVEBTC	16 x 8	Two dimensional array of 8 TCs for 16 VEBs.
VSI	GLV	384	

38.28.1 Counter Implementation

Most of the counters in the 10 GbE controller are used by more than one entity. For example, VSI counters are both a virtual switch port (hence used by a switch monitor agent) and a software device driver interface. The 10 GbE controller does not provide clear-on-read statistics counters, since these need to be duplicated per client.

As a general rule, the 10 GbE controller statistics counters should not be reset by software, because they are shared between more than one owner. Software needs to maintain a delta from the first value seen. If software needs the ability to reset counters, this should be implemented by updating the delta reference value.

To enable firmware to reset counters when a switching element is allocated, or for debug, the counters are implemented as clear-on-write, that means that writing any value clears the counter.

The 10 GbE controller provides 48-bit counters for byte and packet counters, and 32 bit counters for errors and discard events. Software should maintain counters that are appropriate in size for the operating system and MIB requirement.

48-bit counters are implemented as a two registers who’s names end with high and low containing the upper 16 bits and lower 32 bits, respectively.

When accessed using a 64-bit operation from PCI, these accesses are guaranteed by design to be atomic. They are internally converted to two 32-bit accesses that are always consecutive with no interleaving between reads from different software device drivers. A special hardware indication signals the statistics block that this is the case. The statistics block preforms one read of the counter memory to satisfy both requests, thus providing atomicity.

When 32-bit accesses are used to read the counters, they each cause a separate read from the statistics block and therefore care must be taken to properly handle the possibility of one half around between the reads, since atomicity is not guaranteed.

VFs have no access to statistics registers, they should query the PF through the VF to PF virtual channel for their statistics.

38.28.2 Statistics Consistency Rules

- At each switch sample point, a packet is either good or error. That means that when a packet is discarded it is not counted in any of the none-error counters. It is also never counted by any of the next processing stages.
- All drops are counted. A packet with an error is only counted in one counter.
- Ethernet octet counters count the packet as seen on the wire, from the Ethernet header up to and including the CRC (for both Rx and Tx). This means that even though an Rx packet might be stripped of tags before placement the byte counters record the original size. Tx packets are counted after all offloads and insertions.
- Flow control packets are only counted in the flow control counters.
- Statistics specific to DDP/iWARP count packets as they are delivered to the offload engine on Rx, and as they leave the engine on Tx. This applies to both packet lengths and packets that were merged or split. For example packets merged by LRO are counted before the aggregation.
- Maximum lengths are adjusted to accommodate added tags. For example, the highest histogram bin MIB counter GLPRT_PTC9522 also count any packet that would have fit in it without the added STags and VLANs.

38.28.3 Supported MIBs

The 10 GbE controller supports different statistic counters as described in this section. The statistic counters can be used to create statistic reports as required by different standards. The 10 GbE controller statistic counters enable support for the following standards:

- IEEE 802.3 clause 30 management — DTE section.
- NDIS 6.0 OID_GEN_STATISTICS.
- RFC 2819 — RMON Ethernet statistics group for ports and VSIs.
- RFC 2863 — SMON Ethernet statistics group for various switch elements.
- Linux Kernel
- net_device_stats

The following section describes the match between the 10 GbE controller's internal statistic counters and the counters requested by the different standards.

38.28.4 Interface Statistics at VSIs and Logical Interfaces

The 10 GbE controller provides Ethernet interface group statistics per RFC 2863 on each of the VSIs, port aggregators and virtual bridges. Table 38-374 lists the interface counters in the 10 GbE controller and their sizes.

VF and PF VSIs have the same counter set. The only difference between the counter set for VSIs and the one for other switch element is that the RUPP (unknown protocol) counter is replaced by a missed packet counter RMPC.

Note that the counter names repeat themselves with a different prefix for the element type. See Table 38-373 for the list of prefixes. The width of the counters that are provided, is calculated to enable ample time for software to get the statistics before they wrap around. The 48-bit counters are a pair of registers, one ending with an L that holds with the lower 32-bits of the counter, the higher 12 bits rest of the bits are in a register ending with and H (listed in Table 38-374 as {H,L}).

Table 38-374.Ethernet Interface Group Statistics Counters

Register Name	Width	Description
[PFX]GORC{H,L}	48	Incoming packet octets.
[PFX]UPRC{H,L}	48	Incoming number of unicast packets.
[PFX]MPRC{H,L}	48	Incoming number of multicast packets.
[PFX]BPRC{H,L}	48	Incoming number of broadcast packets.
[PFX]RDPC	32	Incoming packet discards.
[PFX]RUPP	32	Incoming unknown protocol packets.
[PFX]GOTC{H,L}	48	Outgoing packet octets.
[PFX]UPTC{H,L}	48	Outgoing number of unicast packets.
[PFX]MPTC{H,L}	48	Outgoing number of multicast packets.
[PFX]BPTC{H,L}	48	Outgoing number of non unicast packets.
[PFX]TDPC	32	Outgoing number of discards.
[PFX]TEPC	32	Outgoing packet errors.

Some error counters are not implemented at specific switch elements because they cannot happen at those elements in the current switch design. If software is queried about the value of these counters it should return zero (see Table 38-375).

Table 38-375.Error Counters Not Implemented in Current Design

Counter Name	Switch element	Description
GLPRT_XEC	Port	Port XSUM error count.
GLPRT_TEPC	Port	Port transmit error packet count.
GLV_TDPC	VSI	VSI transmit packets discarded count.
GLSW_RDPC	PA or VEB	Switch receive packets discarded count.
GLSW_TEPC	PA or VEB	Switch transmit error packet count.
GLPRT_TDPC	Port	Packets that were discarded on transmit while link was down.

38.28.4.1 MAC or Physical Uplink Interface Statistics

The MAC or physical uplink interface statistics counters are accessible to the device management/control entity in VMM or IOVM through the PF. The MAC or physical link interface statistics use the counters listed in Table 38-376. The only exception is the Unknown Protocol Packet (UPP) counter, which has no meaning in the case of the physical port and therefor is absent.

In addition, the 10 GbE controller provides per MAC port some of the statistics out of the IEEE 802.3 clause 30 management counters as well as part of the RMON Ethernet statistics group as defined by IETF RFC 2819. See Table 38-376.

Table 38-376.Additional Per Port Counters

Register Names	Width	Description
GLPRT_PRC64{H,L}	48	Packets Received [64 Bytes] Count.
GLPRT_PRC127{H,L}	48	Packets Received [65–127 Bytes] Count.
GLPRT_PRC255{H,L}	48	Packets Received [128–255 Bytes] Count.
GLPRT_PRC511{H,L}	48	Packets Received [256–511 Bytes] Count.
GLPRT_PRC1023{H,L}	48	Packets Received [512–1023 Bytes] Count.
GLPRT_PRC1522{H,L}	48	Packets Received [1024 to 1522] Count.
GLPRT_PRC9522{H,L}	48	Packets Received [1523 to Max Bytes] Count.
GLPRT_PTC64{H,L}	48	Packets Transmitted (64 Bytes) Count.
GLPRT_PTC127{H,L}	48	Packets Transmitted [65–127 Bytes] Count.
GLPRT_PTC255{H,L}	48	Packets Transmitted [128–255 Bytes] Count.
GLPRT_PTC511{H,L}	48	Packets Transmitted [256–511 Bytes] Count.
GLPRT_PTC1023{H,L}	48	Packets Transmitted [512–1023 Bytes] Count.
GLPRT_PTC1522{H,L}	48	Packets Transmitted [1024 to 1522] Count.
GLPRT_PTC9522{H,L}	48	Packets Transmitted [1523 to Max Bytes] Count.
GLPRT_LXONRXCNT	32	Link XON Received Count.
GLPRT_LXOFFRXCNT	32	Link XOFF Received Count.
GLPRT_LXONTXCNT	32	Link XON Transmitted Count.
GLPRT_LXOFFTXCNT	32	Link XOFF Transmitted Count.
GLPRT_PXONRXCNT[n]	32	Priority XON Received Count.
GLPRT_PXOFFRXCNT[n]	32	Priority XOFF Received Count.
GLPRT_PXONTXCNT[n]	32	Priority XON Transmitted Count.
GLPRT_PXOFFTXCNT[n]	32	Priority XOFF Transmitted Count.
GLPRT_PRRXON2OFFCNT[n]	32	Priority XON to XOFF Count.
GLPRT_CRCERRS	32	CRC Error Count.
GLPRT_ILLERRC	32	Illegal Byte Error Count.
GLPRT_MLFC	32	MAC Local Fault Count.
GLPRT_MRFC	32	MAC Remote Fault Count.
GLPRT_RLEC	32	Receive Length Error Count.
GLPRT_RUC	32	Receive Undersize Count.
GLPRT_RFC	32	Receive Fragment Count.
GLPRT_ROC	32	Receive Oversize Count.
GLPRT_RJC	32	Receive Jabber Count.
GLPRT_MSPDC	32	MAC short Packet Discard Count. Only supported when in 10G mode. note that these packets are also counted as undersized packets
GLPRT_RDPC	32	Received packets from the network that are dropped in the receive packet buffer. The packets are dropped due to possible lack of bandwidth on PCIe or total bandwidth of the internal data path.
GLPRT_LDPC	32	Loopback Dropped Packet Count.
GLPRT_TDOLD	32	Transmit Packets Dropped On Link Down.
GLPRT_RUPP	32	Packets received on this port that were dropped because they did not match any S-tag. Relevant only if a PV is used on this port, should be ignored otherwise.

38.28.4.2 VEB Statistics

The 10 GbE controller supports SMON statistics per RFC 2613 for each instance of the VEB. The 10 GbE controller supports the following set of smonVlanStats counters per 802.1Q VLAN and smonPrioStats counters per priority. See Table 38-377 and Table 38-378.

Table 38-377.VEB Per VLAN Counters

Register Name	Width	Description
GLVEBVL_GORC{H,L}	48	Incoming packet octets.
GLVEBVL_GOTC{H,L}	48	Outgoing packet octets.
GLVEBVL_UPC{H,L}	48	number of unicast packets.
GLVEBVL_MPC{H,L}	48	number of multicast packets.
GLVEBVL_BPC{H,L}	48	number of broadcast packets.

Table 38-378.VEB Per UP Counters

Register Name	Width	Description
GLVEBTC_RPC{H,L}	48	Total number of packets received per priority.
GLVEBTC_RBC{H,L}	48	Total number of octets received per priority.
GLVEBTC_TPC{H,L}	48	Total number of packets transmitted per priority.
GLVEBTC_TBC{H,L}	48	Total number of octets transmitted per priority.

Table 38-379.VEB Per VSI Counters

Register Name	Width	Description
GLV_TEPC	32	Transmit error packet count.
GLV_GOTC{H,L}	48	Transmit octet count. Counts number of bytes transmitted by this VSI.
GLV_GORC{H,L}	48	Receive octet count. Counts number of bytes received by this VSI.
GLV_UPRC{H,L}	48	Receive unicast packet count. Counts number of unicast packets received by this VSI.
GLV_RUPP{_H,}	48	Receive packets dropped because of an unknown protocol or no forward destination.
GLV_BPTC[H,L}	48	Transmit broadcast packet count. Counts number of broadcast packets transmitted by this VSI.
GLV_MPTC{H,L}	48	Transmit multicast packet count. Counts number of multicast packets transmitted by this VSI.
GLV_MPRC{H,L}	48	Receive multicast packet count. Counts number of multicast packets received by this VSI.
GLV_BPRC{H,L}	48	Receive broadcast packet count. Counts number of broadcast packets received by this VSI.
GLV_UPTC{H,L}	48	Transmit unicast packet count. Counts number of unicast packets transmitted by this VSI.
GLV_RDPC	48	Counts (per VSI) packets that were drop due to no descriptors in host queue.
GLV_GORCH	48	Receive octet count. Counts number of bytes received by this VSI.

The 10 GbE controller provides virtual bridge port or interface statistics counters for VSIs and uplinks associated with a VEB instance. See Section 38.28.4 for additional details on VSI and uplink interface statistics.

38.28.4.3 Statistics Resources

Most all statistic counters are statically allocated. This means that for each element type there is a 1:1 ratio between the number of elements and the number of counter sets. The only two exceptions to this are VEB per VLAN statistics. There can be up to 256 VEBs (switch IDs), but only 32 of these might be allocated a statistic block. The total number of VLAN x VEB combinations in the system can exceed the number of counter sets.

For statically allocated counters, the admin command allocating the object returns the offset in the counter array of the counters for this entity. For dynamically allocated counters, the allocator must request statistics counters for the object. If statistical counters are not available, the allocation fails. The allocating software device driver might retry the operation without requesting statistics in that case.

38.28.5 Other Statistics

This section includes statistic counters that are not included in the previous sections.

Table 38-380.Other Statistics

Name	Description	Size	Comment
GL_RXERR1 {H,L}	Receive Error Counter 1	64 Bits	Count dropped packet due to one of the following exceptions: • Packet size is larger than RXMAX of the queue. • Internal receive queue context error (could be a result of a reset in progress). • Receive descriptor unsupported request on the PCI or internal dummy completion (the dummy completion is a result of a reset in progress).
GL_RXERR2 {H,L}	Receive Error Counter 2	64 Bits	Count dropped packet due to one of the following exceptions: • Packets directed to invalid receive queues. • Packets directed to disabled receive queues. • Packets dropped by the switch filters (these packets are counted also by the switch statistics). • Packets dropped due to MAC errors or FC CRC errors. • Packets dropped by the FD filter. • Packets dropped due to VM reset, VF reset or PF reset.

38.28.6 RDMA Statistics

Refer to the PE statistics section for more details. Note that switch VSI statistics are counted for PE VSIs at PE ingress, that is before any PE processing.

38.29 LLDP Protocol

38.29.1 Introduction

The 10 GbE controller supports the IEEE DCB standards such as IEEE 802.1Qaz ETS, IEEE 802.1Qbb PFC, and IEEE 802.1Qbg ECB. Devices that support these standards use IEEE 802.1AB LLDP to exchange configuration information with their network link partner.

LLDP is a link layer protocol that enables a LAN station to advertise capabilities and status of the system. An LLDP agent transmits and receives information to and from the LLDP agents of other stations attached to the same LAN. The information distributed and received in each LLDPDU (LLDP data unit) is stored in two MIBs per physical LAN port, one for nearest bridge and the other for non-TPMR. This information is used to configure the 10 GbE controller DCB features.

38.29.2 Scope

The 10 GbE controller supports an embedded LLDP agent that runs on the EMP.

This section describes implementation of the embedded LLDP agent. It also describes the agents operational modes, supported TLVs, configuration and run time operation of LLDP.

38.29.3 LLDP Agent

The following IEEE standards use LLDP to exchange configuration parameters with the link partner.

- EVB
- DCB

The IEEE 802.Qbg specification enables a service provider to support multi-point LAN services to customers over a single physical link by using multiple virtual channels, known as S-channels.

The DCB features (PFC, ETS, DCBX, App TLV) are defined as requirements for a VLAN aware bridge component. Both C-components and S-components are VLAN-aware bridge components. The 10 GbE controller supports a single instance of DCB logic per physical LAN port. Therefore, the 10 GbE controller DCB logic is associated with the component connected to the physical LAN port, either C or S depending on the use case/configuration.

The 10 GbE controller supports an LLDP agent that exchanges LLDP MIB with its peer.

The LLDP agent is active under the following conditions:

1. During pre-boot operations including S5 (D0u and D0a).
2. During operating system present mode unless software explicitly turns off the agent using the Stop LLDP Agent AQ command.

During DCBX exchange, the LLDP agent sets the willingness bit and accepts recommended setting from its link partner.

The LLDP agent is enabled during power-on by setting the NVM *Factory LLDP Admin Status* field, (enabled separately per Ethernet port). It is disabled when the Stop LLDP Agent AQ command is executed (per Ethernet port). Software might transfer ownership of LLDP processing back to the device by issuing a Start LLDP Agent AQ command.

38.29.4 LLDP Processing

38.29.4.1 LLDPDU Addressing and Forwarding

38.29.4.1.1 Egress Rules

On the egress side, the LLDP agent uses the appropriate group MAC address as the destination MAC along with Ether type 0x88cc:

- DCB — Nearest bridge address
- CDCP — Non-TPMR nearest bridge address

A control port that needs to send control packets overriding these rules should use the *SWTCH* field in the transmit context descriptor of the control packet. Tagged (802.1Qbg or 802.1BR) LLDP packets are forwarded like regular packets. The software device driver might define different rules (forward to control VSI or drop) using the Add Control Packet Filter admin command.

38.29.4.1.2 Ingress Rules

When an LLDP agent is enabled in the device, the following forwarding rules apply for Rx LLDP packets:

Table 38-381.Forwarding Rules for Rx LLDP Packets

Destination MAC address	S-Tagged LLDP Packets	Untagged LLDP Packets
Nearest Bridge	Forward to PF if programmed so by the PF. Drop otherwise.	Forward to EMP.
Non-TPMR	Forward to PF if programmed so by the PF. Drop otherwise.	Forward to EMP.
Nearest Customer Bridge	Forward to PF if programmed so by the PF. Drop otherwise.	SFP mode. Forward to PF if programmed so by the PF. Drop otherwise.

38.29.4.2 Supported TLV

The LLDP agent should include following TLV as part of the LLDPDU.

Chassis ID TLV — Is one of four mandatory TLVs. Uses TLV type value of one and subtype value of four (MAC address). This TLV contains the permanent/factory assigned MAC address of the LAN port.

Port ID TLV — Is one of four mandatory TLVs. Uses TLV type value of two and subtype value of three (MAC address). This TLV contains MAC address of any physical function. If there is none assigned to a physical function, then this TLV contains the permanent/factory assigned MAC address of the LAN port.

TTL TLV — Is one of four mandatory TLVs. Uses TLV type value three. This TLV contains time to live value and is the lower between [(msgTxHold * msgTxInterval) +1] and 65535. Default msgTxHold and msgTxInterval are defined in NVM and are loaded by the agent during initialization. Refer to LLDP configuration section for default values.

End of LLDPDU TLV — Is one of four mandatory TLVs. Uses TLV type value zero. This TLV does not contain any information and sets the TLV information string length to zero.

Note: Although this is a mandatory TLV, there might some implementations that do not send it. The 10 GbE controller therefore does not require that an LLDPDU ends with this TLV.

OEM Device type TLV — Is one of the three OEM required TLVs. This TLV uses TLV type value of 127 with subtype of one. Refer to OEM specific LLDP specifications for OUI and contents of the TLV.

OEM Firmware Version TLV — Is one of the three OEM required TLVs. This TLV uses TLV type value of 127 with subtype of three. Refer to OEM specific LLDP specifications for OUI and contents of the TLV.

OEM Port Capabilities TLV — Is one of the three OEM required TLVs. This TLV uses TLV type value of 127 with subtype of four. Refer to the OEM specific LLDP specifications for OUI and contents of the TLV.

DCBX ETS Configuration TLV — This TLV uses type value of 127 with OUI of IEEE 802.1, which is 0x0080C2. Subtype is 0x09.

This TLV information string uses the following default values:

- *Willing* bit set to 1b indicating that this station is willing to accept a configuration from a remote station.
- CBS bit is set to 0b indicating that this station does not support a credit-based shaper.
- Maximum TCs.
- Priority assignment table is a four octet string with four bits per entry.

- Bandwidth table is an eight octet string with each octet defining the bandwidth percentage for TCs. Octet zero specifies a bandwidth percentage of TC zero, octet one for TC one and so on.
- TSA assignment table is a string of 8 octets with an 8-bit value specifying a transmission selection algorithm per TC.

DCBX PFC Configuration TLV — This TLV uses type value of 127 with OUI of IEEE 802.1, which is 0x0080C2 along with a subtype of 0x0B.

This TLV information string uses the following default values:

- *Willing* bit set to 1b indicating that this station is willing to accept a configuration from a remote station.
- *MBC* bit is set to 0b.
- PFC capability is a 4-bit unsigned integer indicating the number of TCs that simultaneously support PFC. Default value is eight.
- PFC enable is an 8-bit vector that indicates which TCs have PFC enabled. Default value is zero. No traffic call uses have PFC enabled by default.

DCBX Application Priority Configuration TLV — This TLV uses type value of 127 with OUI of IEEE 802.1, which is 0x0080C2 along with a subtype of 0x0C.

By default this optional TLV is not transmitted by the LLDP agent but the agent reflects the table received from a peer network port if the agent receives LLDPDU with a remote link partner.

This TLV uses a type value of 127 with OUI of IEEE 802.3, which is 00-12-0F along with a subtype of five. TLV information string carries timing information following Low Power Idle (LPI) exit.

The first six octets contain TransmitTw, ReceiveTw and FallbackTw. The default values for TransmitTw and Receive Tw values are loaded from the NVM. The value of FallbackTw is the same as ReceiveTw (FallbackTw = ReceiveTw). TransmitTw and ReceiveTw can be changed at runtime by using admin queue commands from a PF.

Note: There is a single LLDP agent for all LAN ports and each LAN port has multiple PFs. Therefore, it is likely that these variables can overwritten by multiple admin queue commands. The last admin command takes precedent.

Echo Transmit Tw and Echo Receive Tw values are of the remote link partner. If the link partner has not transmitted an LLDPDU then hardware is the default value for the resolved PHY type.

CDCP TLV — This TLV is over Ethernet frames with the nearest non-TPMR bridge group address.

38.29.4.3 LLDPDU Transmission and Reception

The first LLDP transmission is immediate after the LLDP agent reaches a ready state and the link is up. This first transmission populates the LLDPDU with default TLV values.

The LLDP agent enters a fast transmission state/period each time the LLDP agent detects a new neighbor. The LLDP agent enters a new neighbor detected state (fast transmission period) when an IEEE 802.1az link layer LLDP frame with a new chassis and port ID is received. The fast transmission state exists after transmitting a txFastInit number of LLDPDUs. The default value of txFastInit is four.

While not in fast transmission mode, the LLDP agent transmits every msgTxInterval unless there is change in local LLDP MIB variables. LLDPDU is transmitted immediately, without waiting for msgTxInterval when there is a change in local MIB variables.

The 10 GbE controller supports a single LLDP neighbor (such as a single remote MSAP). Therefore, if the received LLDPDU is from a new neighbor, the existing information within the current remote systems MIB is discarded in order to make space for the new information received in the LLDPDU.

The transmission and reception flow is interrupted by several events, each causing a disruption in the normal flow. The behavior in such cases is as follows:

- LLDP events are visible to both ends:
  - Aging of the LLDP timer or LLDPDU is received with a TTL value of zero (like a shutdown LLDPDU).
    - Delete all information in the LLDP remote systems MIB associated with the respective MSAP identifier.
    - EMP reconfigures the device based on the local defaults.
  - Missing TLV in a received LLDPDU:
    - Delete all information for the missing TLV in the LLDP remote systems MIB associated with the respective MSAP identifier.
    - EMP reconfigures the device based on the local defaults of this TLV.
- Events that cause a link down:
  - Link down event (only if the LLDP timer has not expired).
    - When the link is back up, delete all information in the remote systems MIB associated with the LLDP agent(s) for this link.
    - EMP reconfigures the device based on the local defaults.
  - Device resets that cause link down (such as EMP reset, global reset, PCI resets that cause link down).
    - Device performs an internal GLOBR that brings the link down momentarily.
    - Continue as in the link down case.
  - LLDP ownership transfer from software to firmware:
    - Software issues a global reset to the device.
    - Software sends a Start LLDP Agent AQ command to start the LLDP agent in firmware.
    - Continue as in the link down case.
- Events not visible to the other end:
  - LLDP ownership transfer from firmware to software.
    - LLDP agent terminated in the device.
    - No automatic change in device configuration.
    - EMP waits for software commands.
  - Device resets that do not cause a link down (like a core reset, PCI resets that do not cause link down or function-level resets):
    - Device performs an internal CORER.
    - EMP reconfigures the core based on the LLDP MIBs.

Note: No change in MAC/PHY state including flow control configuration and operation.

LLDP reception and transmission continues once the previous configuration completes. Configuration is estimated to take milliseconds, which is below the LLDP timeout values (usually in seconds).

38.29.4.4 LLDP Protocol Variables

Unless otherwise specified, the LLDP agent operational state variables are set to values recommended in Section 9.2.2 of the IEEE 802. AB - 2009 standard. However, certain values can be configured via the NVM and are loaded from the NVM when the LLDP agent reaches an operational state; this occurs after a power up or a device reset. See Section 38.29.5.2 for LLDP protocol variables that can be configured via the NVM.

38.29.4.5 LLDP Data Store

The LLDP agent maintains sufficient information to enable a host software based management agent to support basic LLDP MIB and organizationally specific LLDP MIB extensions.

The following basic variables are maintained by the LLDP agent:

msgTxInterval — This variable defines the time interval in timer ticks between transmissions during normal transmission periods.

msgTxHold — This variable is used as a multiplier of msgTxInterval to determine the value of txTTL that is carried in LLDP frames transmitted by the LLDP agent.

reinitDelay — This parameter indicates the amount of delay from when adminStatus becomes disabled until re-initialization is attempted.

txCreditMax — Maximum number of consecutive LLDPDUs that can be transmitted at any time.

msgFastTx — This variable defines the time interval in timer ticks between transmissions during fast transmission periods.

txFastInit — This value determines the number of LLDPDUs that are transmitted during a fast transmission period.

adminStatus — This variable indicates whether or not the LLDP agent is enabled. The defined values for this variable are as follows:

- Integer value of three means the LLDP agent is enabled for reception and transmission of LLDPDUs.
- Integer value of two means the LLDP agent is enabled for transmission of LLDPDUs only.
- Integer value of one means the LLDP agent is enabled for reception of LLDPDUs only.
- Integer value of zero means the LLDP agent is disabled for both reception and transmission.

The following statistics are maintained by LLDP agent per port:

RemTableLastChangeTime, RemTableInserts, RemTableDeletes, RemTbleDrops, RemTableAgeouts, TxPortFramesTotal, RxPortFramesDiscarded, RxPortFrameErrors, RxPortFramesTotal, RxPortTLVsDiscardedTotal, RxPortTLVsUnrecognizedTotal, RemTooManyNeighbors.

The LLDP agent maintains the last received LLDPDU per port. The upper bound for the size of LLDPDU is 1500 bytes. Details about the data structure used for information store is implementation specific and beyond scope of this document.

The 10 GbE controller provides LSAP services to the operating system provided MSAP services that are used by LLDP agents hosted by system processors.

38.29.5 Initialization and Configuration

38.29.5.1 Initialization

As part of EMP initialization, an the LLDP agent is loaded and initialized by the EMP. The LLDP agent loads default values for TLVs from the NVM. LLDP agent transitions to a ready state and waits for LAN link up before transmitting the first LLDPDU.

The DCBX agent operates in slave mode and sets the willingness bit and accepts recommendations from a link partner.

38.29.5.2 LLDP Configuration

38.29.5.2.1 LLDP Protocol Variables

The following LLDP timers can be configured via the NVM:

msgFastTx — This variable defines the time interval in timer ticks between transmissions during fast transmission periods. The default value of msgFastTx is one. This value can be changed by management to any value in the range 1 through 3600.

msgTxInterval — This variable defines the time interval in timer ticks between transmissions during normal transmission. The default value for msgTxInterval is 30 seconds. This value can be changed by management to any value in the range 1 through 3600.

msgTxHold — This variable is used as a multiplier of msgTxInterval to determine the value of txTTL that is carried in LLDP frames transmitted by the LLDP agent. The recommended default value of msgTxHold is four. This value can be changed by management to any value in the range 1 through 100.

txCreditMax — This value determines the maximum number of LLDPDUs that can be sent per second. This value defaults to five. This value can be changed by management to any value in the range 1 through 10.

txFastInit — This value determines the number of LLDPDUs that are transmitted during a fast transmission period. The default value of txFastInit is four. This value can be changed by management to any value in the range one through eight.

reinitDelay — This parameter indicates the amount of delay from when adminStatus becomes disabled until re-initialization is attempted. The default value is two.

38.29.5.2.2 LLDP Admin Queue Commands

The following commands are supported by the 10 GbE controller to manage the LLDP agent and provide information to the software device drivers.

Table 38-382.LLDP Admin Queue Commands (Sheet 1 of 2)

Command	OpCode	Description
Get LLDP MIB	0x0A00	Fetch latest LLDP MIB.
Configure LLDP MIB Change Event	0x0A01	Request and deliver a notification that the peer has sent an updated LLDP MIB.
Add LLDP TLV	0x0A02	Add a new TLV to the local LLDP MIB.
Update LLDP TLV	0x0A03	Update an existing TLV in the local LLDP MIB.
Delete LLDP TLV	0x0A04	Delete a TLV from the local LLDP MIB.
Stop LLDP Agent	0x0A05	Used to stop or shutdown the LLDP agent.
Start LLDP Agent	0x0A06	Start an LLDP agent running.
Get CEE DCBX CFG	0x0A07	Retrieves the CEE configuration.

Table 38-382.LLDP Admin Queue Commands (Sheet 2 of 2)

Command	OpCode	Description
Set Local LLDP MIB	0x0A08	Load the DCBX configuration.
Stop/Start a Specific LLDP Agent	0x0A09	Stop and restart the firmware DCBX agent.

Get LLDP MIB

This command, posted on the ATQ, is an indirect AQ command. The software device driver requests the complete LLDP MIB, providing a response buffer (address/length pair) and the type of LLDP MIB requested. The LLDP MIB is associated with a physical LAN port. Instead of formatting the LLDP MIB in any particular way, firmware should write the entire packet (including headers) that was sent/received on the wire for the particular MIB type specified. The MIB is guaranteed to fit in a 1.5 KB packet, so there is no need to use a large buffer. For a particular bridge type, software might request the local MIB, the remote MIB, or both MIBs.

Firmware writes back the complete LLDP MIB to the response buffer. It writes the length of the LLDP MIB into the *Datalen* field in the descriptor on the ATQ. It writes the status of the request in the *Return Value* field. If both MIBs are returned, firmware always writes the local MIB first and the remote MIB immediately following the Local MIB.

Table 38-383.Get LLDP MIB Command

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ section for details.
Opcode	2-3	0x0A00	Command opcode.
Datalen	4-5		Length of response buffer.
Return Value/ VFID	6-7		Return value. Zeroed by the software device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Type	16	MIB Type	Bit 0-1 = Direction. 0 = Local MIB (sent by device). 1 = Remote MIB (received by device). 2 = Both local and remote MIBs. 3 = Reserved. Bits 2-3 = Bridge type. 0 = Nearest bridge. 1 = Non-TPMR bridge. 2 = Reserved. 3 = Reserved. Bits 4-7: Reserved.
Reserved	17-23	Reserved	Reserved.
Data Address High	24-27		Address of response buffer.
Data Address Low	28-31		Address of response buffer.

Table 38-384.Get LLDP MIB Response

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ
Opcode	2-3	0x0A00	Command opcode.
Datalen	4-5		Length of LLDP MIB if status = success. In bytes.
Return Value/ VFID	6-7		Return value. Zeroed by software device driver. Written by firmware. Status of request. A value of success means that the command was performed successfully. Error Codes: ENOENT - Firmware returns this value if any requested LLDP MIB doesn't exist. EPERM - Firmware returns this value if software has taken control of LLDP processing. EFBIG - Firmware returns this value when the size of LLDPDU is larger than size of the response buffer. EINVAL - Firmware returns this value when software requests for a bad request (like an invalid bridge type or direction).
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Type	16	MIB Type	Bit 0-1 = Direction. 0 = Local MIB (sent by device). 1 = Remote MIB (received by device). 2 = Both local and remote MIBs. 3 = Reserved. Bits 2-3 = Bridge yype. 0 = Nearest bridge. 1 = Non-TPMR bridge. 2 = Reserved. 3 = Reserved. Bits 4-7 = Reserved.
Reserved	17	Reserved	Reserved.
Local MIB Length	18-19	Len	If the response buffer contains a local MIB, this field reports its length. If no local MIB is present, this field is written with a value of zero.
Remote MIB Length	20-21	Len	If the response buffer contains a remote MIB, this field reports its length. If no remote MIB is present, this field is written with a value of zero.
Reserved	22-23	Reserved	Reserved.
Data Address High	24-27		Address of response buffer.
Data Address Low	28-31		Address of response buffer.

Note: If only one MIB is present, it is written to the response buffer beginning at the first byte of the response buffer (offset=0). If two MIBs are present, the local MIB is always written first (offset=0), and the remote MIB is written immediately following the local MIB (offset=LocalMIBLength).

Configure LLDP MIB Change Event

This command, posted on the ATQ, is a direct AQ command. The software device driver uses this command to request that firmware post an event on the ARQ when the LLDP MIB associated with this interface changes. The software device driver can also use this command to request that firmware stop posting this event to the ARQ.

Firmwasre writes back the status of the request to the *Return Value* field.

Table 38-385.Configure LLDP MIB Change Event Command

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ
Opcode	2-3	0x0A01	Command opcode.
Datalen	4-5	0	Direct command; no response buffer.
Return Value/ VFID	6-7		Return value. Zeroed by the software device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Command	16	Cmd	Bit 0 = Command. 0 = Enable event. 1 = Disable event. Bits 1-7 = Reserved.
Reserved	17-31	Reserved	Reserved.

Table 38-386.Configure LLDP MIB Change Event Response

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ section for details.
Opcode	2-3	0x0A01	Command opcode.
Datalen	4-5	0	Direct command; no response buffer.
Return Value/ VFID	6-7		Return value. Zeroed by the software device driver. Written by firmware. Status of request. A value of success means that the command was performed successfully. Error code: EPERM - Firmware returns this value if software has taken control of LLDP processing.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Reserved	16-31	Reserved	Reserved

LLDP MIB Change Event

This event is posted on the ARQ to indicate to software that any LLDP MIB associated with the physical interface has changed. The LLDP MIB change event must also be posted if a multiple peers condition is detected or if a TLV is aged out.

Firmware provides the status of the event, the length of the LLDP MIB in bytes, the type of LLDP MIB which has changed, and copies the cookie value from the associated configure LLDP MIB change event.

The response buffer includes the entire MIB that changed. The formatting is identical to the response for the Get LLDP MIB command.

Note that the opcode for this event is the same as the opcode for the related command on the ATQ that enabled this event to be sent to software.

Table 38-387.LLDP MIB Change Event

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ section for details.
Opcode	2-3	0x0A01	Event code.
Datalen	4-5		Length of LLDP MIB if status = success. In bytes.
Return Value/ VFID	6-7		Return value. Zeroed by the software device driver. Written by firmware. Status of request.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Type	16	MIB Type	Bit 0-1 = Direction. 0 = Local MIB (sent by device). 1 = Remote MIB (received by device). 2 = Reserved. 3 = Reserved. Bits 2-3 = Bridge type. 0 = Nearest bridge. 1 = Non-TPMR bridge. 2 = Reserved. 3 = Reserved. Bits 4-5 = Miscellaneous. 0 = Port's Tx active. 1 = Port's Tx suspended and drained. 2 = Reserved. 3 = Port's Tx suspended and drained. Blocked TC pipe flushed. Bits 6-7 = Reserved.
Reserved	17-23	Reserved	Reserved.
Data Address High	24-27		Address of response buffer.
Data Address Low	28-31		Address of response buffer.

Add LLDP TLV

This command, posted on the ATQ, is an indirect AQ command. The software device driver provides the type of MIB to be updated, the TLV to be added, and the address/ length of the buffer containing the TLV. Software is responsible for guaranteeing that the TLV to be added does not already exist in the MIB or follows the TLV usage rules for TLVs, which allow multiple instances. Software can only add TLVs to the Local LLDP MIB.

Firmware adds the a new TLV to the Local LLDP MIB just before the End of LLDPDU TLV command. Firmware writes back the complete LLDP MIB to the response buffer. It writes the length of the LLDP MIB into the *Datalen* field in the descriptor on the ATQ.

Table 38-388.Add LLDP TLV Command (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ section for details.
Opcode	2-3	0x0A02	Command opcode.
Datalen	4-5	Len	Length of indirect buffer.
Return Value/ VFID	6-7		Return value. Zeroed by the software device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.

Table 38-388.Add LLDP TLV Command (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Type	16	MIB Type	Bit 0-1 = Reserved. Bits 2-3 = Bridge type. 0 = Nearest bridge. 1= Non-TPMR bridge. 2 = Reserved. 3 = Reserved. Bits 4-7 = Reserved.
Reserved	17	Reserved	Reserved.
Length	18-19	Len	Length of TLV placed in the indirect buffer by the software device driver.
Reserved	20-23	Reserved	Reserved.
Data Address High	24-27		Address of indirect buffer.
Data Address Low	28-31		Address of indirect buffer.

Table 38-389.Add LLDP TLV Response

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ section for details.
Opcode	2-3	0x0A02	Command opcode.
Datalen	4-5		Length of LLDP MIB if status = success. In bytes.
Return Value/ VFID	6-7		Return value. Zeroed by the software device driver. Written by firmware. Status of request. Error codes: ENOMEM – Firmware returns this value if there is not enough space available to add the new TLV. EINVAL – Firmware returns this value if the MIB Type associated with this command does not exist. EPERM - Firmware returns this value if software has taken control of LLDP processing.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Type	16	MIB Type	Bit 0-1 = Reserved. Bits 2-3 = Bridge type. 0 = Nearest bridge. 1 = Non-TPMR bridge. 2 = Reserved. 3 = Reserved. Bits 4-7 = Reserved.
Reserved	17-23	Reserved	Reserved.
Data Address High	24-27		Address of response buffer.
Data Address Low	28-31		Address of response buffer.

Update LLDP TLV

This command, posted on the ATQ, is an indirect AQ command. The software device driver provides the bridge type of the MIB to be updated, the TLV to be updated (both original and updated versions), the offset/length of both TLVs in the indirect buffer, and

the address/length of the indirect buffer. Software is responsible for guaranteeing that the TLV to be updated does already exist in the MIB. Only a local MIB can be updated by the software device driver.

Firmware must match the entire original TLV in the MIB before performing an update. There are some TLV types that might appear more than once, and so matching based only on type/subtype is not sufficient.

Firmware writes back the complete LLDP MIB to the response buffer. It writes the length of the LLDP MIB into the *Datalen* field in the descriptor on the ATQ.

Table 38-390.Update LLDP TLV Command

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ section for details.
Opcode	2-3	0x0A03	Command opcode.
Datalen	4-5	Len	Length of indirect buffer.
Return Value/ VFID	6-7		Return value. Zeroed by the software driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Type	16	MIB Type	Bit 0-1 =Reserved. Bits 2-3 = Bridge type. 0 = Nearest bridge. 1 = Non-TPMR bridge. 2 = Reserved. 3 = Reserved. Bits 4-7 = Reserved.
Reserved	17	Reserved	Reserved.
Length1	18-19	Len	Length of original TLV in the indirect buffer. Offset is assumed to be zero.
Offset2	20-21	Offset	Offset of updated TLV in the indirect buffer.
Length2	22-23	Len	Length of updated TLV in the indirect buffer.
Data Address High	24-27		Address of indirect buffer.
Data Address Low	28-31		Address of indirect buffer.

Table 38-391.Update LLDP TLV Response (Sheet 1 of 2)

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ section for details.
Opcode	2-3	0x0A03	Command opcode.
Datalen	4-5		Length of LLDP MIB if status = success. In bytes.
Return Value/ VFID	6-7		Return value. Zeroed by the software device driver. Written by firmware. Status of request. Error codes: EINVAL – Firmware returns this value if the requested MIB does not exist. ENOXIO – Firmware returns this value if the original TLV does not exist in the requested MIB. ENOMEM – Firmware returns this value if the updated TLV is larger than the original TLV and there is not enough space to increase the size of the TLV. EPERM - Firmware returns this value if software has taken control of LLDP processing.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.

Table 38-391.Update LLDP TLV Response (Sheet 2 of 2)

Name	Bytes.Bits	Value	Remarks
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Type	16	MIB Type	Bit 0-1 = Reserved. Bits 2-3 = Bridge type. 0 = Nearest bridge. 1 = Non-TPMR bridge. 2 = Reserved. 3 = Reserved. Bits 4-7 = Reserved.
Reserved	17-23	Reserved	Reserved.
Data Address High	24-27		Address of response buffer.
Data Address Low	28-31		Address of response buffer.

Delete LLDP TLV

This command, posted on the ATQ, is an indirect AQ command. The software device driver provides the type of MIB to be updated and a copy of the TLV to be deleted. Software is responsible for guaranteeing that the TLV to be deleted does already exist in the MIB.

Firmware writes back the complete LLDP MIB to the response buffer. It writes the length of the LLDP MIB into the *Datalen* field in the descriptor on the ATQ.

Table 38-392.Delete LLDP TLV Command

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ section for details.
Opcode	2-3	0x0A04	Command opcode.
Datalen	4-5	Len	Length of indirect buffer.
Return Value/ VFID	6-7		Return value. Zeroed by the software device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Type	16	MIB Type	Bit 0-1 = Reserved. Bits 2-3 = Bridge type. 0 = Nearest bridge. 1 = Non-TPMR bridge. 2 = Reserved. 3 = Reserved. Bits 4-7 = Reserved.
Reserved	17	Reserved	Reserved
Length	18-19	Len	Length of TLV to be deleted. The TLV itself is copied into the indirect buffer by the software device driver.
Reserved	20-23	Reserved	Reserved.
Data Address High	24-27		Address of indirect buffer.
Data Address Low	28-31		Address of indirect buffer.

Table 38-393.Delete LLDP TLV Response

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ section for details.
Opcode	2-3	0x0A04	Command opcode
Datalen	4-5		Length of LLDP MIB if status = success. In bytes.
Return Value/ VFID	6-7		Return value. Zeroed by the software device driver. Written by firmware. Status of request. Error codes: EINVAL – Firmware returns this value if the requested MIB does not exist. ENOXIO – Firmware returns this value if the TLV to be deleted does not exist in the requested MIB. EPERM - Firmware returns this value if software has taken control of LLDP processing.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Type	16	MIB Type	Bit 0-1 = Reserved. Bits 2-3 = Bridge type. 0 = Nearest bridge. 1 = Non-TPMR bridge. 2 = Reserved. 3 = Reserved. Bits 4-7 = Reserved.
Reserved	17-23	Reserved	Reserved.
Data Address High	24-27		Address of response buffer.
Data Address Low	28-31		Address of response buffer.

Stop LLDP Agent

This command, posted on the ATQ, is a direct AQ command. The software device driver uses this command to request that firmware stop or shutdown the LLDP agent on the port.

If stop is specified, the 10 GbE controller stops the LLDP agent on the port and directs all untagged ingress LLDP frames received on the port to the default queue of the control VSI associated with the port aggregator or port extender.

If shutdown is specified, the 10 GbE controller stops the LLDP agent on the port and sends a last LLDP PDU on the wire with TTL = 0 and with nearest bridge and non-TPMR destination address. Firmware then directs all untagged ingress LLDP frames received on the port to the default queue of the S-component control VSI.

Firmware writes back the status of the request.

Any preceding registration to events on the port (via the Configure LLDP MIB Change Event command) is discarded. Software should register for events again once the LLDP agent in the device is active again.

After the response, the software device driver should route the LLDP flows to a control VSI using the Add Control Packet Filter command.

Table 38-394.Stop LLDP Agent Command

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ section for details.
Opcode	2-3	0x0A05	Command opcode.
Datalen	4-5	0	Direct command; no response buffer.
Return Value/ VFID	6-7		Return value. Zeroed by the software device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Command	16	Cmd	Bit 0: Command 0b = Stop LLDP Agent 1b = Shutdown LLDP Agent Bit 1: Command 0b = No effect 1b = Persistent disablement of LLDP agent Bits 2-7: Reserved
Reserved	17-31	Reserved	Reserved.

Table 38-395.Stop LLDP Agent Response

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ section for details.
Opcode	2-3	0x0A05	Command opcode
Datalen	4-5	0	Direct command; no response buffer.
Return Value/ VFID	6-7		Return value. Zeroed by the software device driver. Written by firmware. Status of request. A value of success means that command was performed successfully. Error code: EPERM - Firmware this value if software has taken control of LLDP processing or if the firmware LLDP agent is disabled.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Reserved	16-31	Reserved	Reserved.

Start LLDP Agent

It is expected that CORER has occurred before this command is issued. CORER causes the EMP to reload LLDP forwarding rules from the NVM and re-initialize per NVM settings.

This command, posted on the ATQ, is a direct AQ command. In response to this command EMP re-enables LLDP agent over all ports enabled by the NVM Factory LLDP Admin Status word. It is expected that only pre-boot software issues this command to start the LLDP agent.

If the software device driver defined LLDP forwarding rules previous to sending this command, these rules should be removed using the Remove Control Packet Filter admin command before sending this command.

Table 38-396.Start LLDP Agent Command

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ section for details.
Opcode	2-3	0x0A06	Command opcode.
Datalen	4-5	0	Direct command; no response buffer.
Return Value/ VFID	6-7		Return value. Zeroed by the software device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Command	16	Cmd	Bit 0: Command 0b = Do not start the LLDP agent (Note: this value should not be used) 1b = Start LLDP agent Bit 1: Command 0b = No effect. 1b = Persistent enablement of LLDP agent. Bits 2-7: Reserved
Reserved	17-31	Reserved	Reserved.

Table 38-397.Start LLDP Agent Response

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ section for details.
Opcode	2-3	0x0A06	Command opcode.
Datalen	4-5	0	Direct command; no response buffer.
Return Value/ VFID	6-7		Return value. The following error value can be returned: EEXIST - LLDP agent is already running in firmware. EPERM - Firmware returns this value if the firmware LLDP agent is disabled and command bit 1 was not set.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Reserved	16-31	Reserved	Reserved.

Restore LLDP Agent Factory Settings

This command, posted on the ATQ, is a direct AQ command. The software device driver uses this command to request that firmware uses the factory default setting for firmware's LLDP agent on the port.

Table 38-398.Restore LLDP Agent Factory Settings Command

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See Admin Queue section for details.
Opcode	2-3	0x0A0A	Command opcode
Datalen	4-5	0	Direct command; no response buffer.
Return Value	6-7		Return value. Zeroed by driver. Written by Firmware.
Cookie High	8-11	Cookie	Opaque value, will be copied by the FW into the completion of this command

Table 38-398.Restore LLDP Agent Factory Settings Command

Name	Bytes.Bits	Value	Remarks
Cookie Low	12-15	Cookie	Opaque value, will be copied by the FW into the completion of this command
Command	16	Cmd	Bit 0: Command 0b= Do not restore factory settings. 1b = Restore factory settings. Bits 1-7: Reserved
Reserved	17-31	Reserved	Reserved

Table 38-399.Restore LLDP Agent Factory Settings Response

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See Admin Queue section for details.
Opcode	2-3	0x0A06	Command opcode
Datalen	4-5	0	Direct command; no response buffer.
Return Value	6-7		Return value.
Cookie High	8-11	Cookie	Opaque value, will be copied by the FW into the completion of this command
Cookie Low	12-15	Cookie	Opaque value, will be copied by the FW into the completion of this command
Settings	16		Bit 0: Factory Setting. 0b = Firmware LLDP agent disabled. 1b = Firmware LLDP agent enabled.
Reserved	17-31	Reserved	Reserved

Get CEE DCBX OPER CFG

This command, posted on the ATQ, is an indirect AQ command. The software device driver requests the operational configuration of CEE/DCBX. The software device driver then provides a response buffer (address/length pair).

The EMP writes back the CEE/DCBX configuration to the response buffer.

Table 38-400.Get CEE DCBX OPER CFG Command

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ section for details.
Opcode	2-3	0x0A07	Command opcode.
Datalen	4-5		Length of response buffer.
Return Value/ VFID	6-7		Return value. Zeroed by the software device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Reserved	17-23	Reserved	Reserved.
Data Address High	24-27		Address of response buffer.
Data Address Low	28-31		Address of response buffer.

Table 38-401.Get CEE DCBX OPER CFG Response

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ section for details.
Opcode	2-3	0x0A07	Command opcode.
Datalen	4-5		Length of LLDP MIB if status = success. In bytes.
Return Value/ VFID	6-7		Return value. Zeroed by the software device driver. Written by firmware. Status of request. A value of success means that command was performed successfully. Error codes: ENOENT - Firmware returns this value if any requested LLDP MIB doesn't exist. EPERM - Firmware returns this value if software has taken control of LLDP processing.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Reserved	16-23	Reserved	Reserved.
Data Address High	24-27		Address of response buffer.
Data Address Low	28-31		Address of response buffer.

Table 38-402.Get CEE DCBX OPER CFG Response Buffer Format (Sheet 1 of 2)

Offset (Bytes)	Description
0	Reserved, return zero.
1	Local Oper Num Traffic Class Supported. Returns a value between 1-8 for the number of TCs supported locally.
2-5	Local Oper Priority Assignment. For each UP, 4-bits for the TCID. Available value is 0-7. Upper bit = 0. Bits 0-3 = Assigned for up 0. Bits 4-7 = Assigned for up 1. : Bits 27-31 = Assigned for up 7.
6	Reserved, return zero.
7-14	Local Oper TCB Bandwidth. An 8-byte field. Each byte represents the relative bandwidth allocation of one enabled TC. Byte 7 assigned for TC 0. Relative bandwidth allocation is a value 0-100 and represents the percentage of the available bandwidth this TC is allocated with. The sum of the bandwidth allocated for all TCs equal to 100%.
15	Local Oper PFC Enable. A bitmap containing a PFC-enable flag for each UP. Bit 0 relative to UP 0.
16	Reserved, return zero.
17-18	Local Oper Application Priority. Indicates the application TLV negotiated for iSCSI and FIP (encoded in three bits). 17.0-2 = Reserved. 17.3-5 = Reserved for iSCSI application priority. 17.6-7 = Reserved. 18.0-2 = Reserved for FIP application priority. 18.3-7 = Reserved.
19	Reserved, return zero.

Table 38-402.Get CEE DCBX OPER CFG Response Buffer Format (Sheet 2 of 2)

Offset (Bytes)	Description
20-21	Status flags for DCBX TLVs. For each TLV indicates the status of the TLV: operational mode, synced, error (1b = true, 0b = false). 20.0-2 = Status bits for PG. 20.3-5 = Status bits for PFC. 20.6-7 = Reserved. 21.0-2 = Status bits for application priority.

Set Local LLDP MIB

This command, posted on the ATQ, is an indirect AQ command. The software device driver configures the complete DCBX MIB, providing a response buffer (address/length pair). The DCBX MIB is associated with a physical LAN port and must be expressed in IEEE format even though firmware sends it as CEE TLVs on the wire if needed. Instead of formatting the DCBX TLVs in any particular way, firmware should send on the wire the entire DCBX MIB (including headers) that was posted by the host. The MIB is guaranteed to fit in a 1.5 KB packet, so there is no need to use a large buffer. Once DCBX negotiation with the peer completes, the local DCBX MIB returned by the Get LLDP MIB command reflects the final resolved values. In the future, the command could be extended to support additional LLDP MIBs.

This command is useful to configure the local DCBX agent into the non-willing mode (master mode) and to set the DCB configuration to be pushed to the peer.

Firmware writes back the status of the request in the *Return Value* field.

If the willing bit is set in a DCBX TLV, the DCBX agent should take it in account when resolving DCBX with the peer, as per the rules defined in the IEEE/CEE standard.

Firmware might change the local configuration twice, once after receiving the AQ command to align the default configuration to what is published in the TLVs sent to the peer, and after receiving the peer's TLV when resolving DCBX. These two steps might be collapsed into one single configuration change in case the peer's TLV is received within short delays. The new default configuration is maintained until the next GLOBR event or until a Stop DCBX Agent AQ command is received.

If the command is received while the firmware DCBX agent is disabled or stopped, the MIB is parsed by firmware and used to configure the local DCB settings of the port, with no DCB TLV exchange with the peer performed by firmware. Firmware drains the Tx pipe if TC or PFC changes were pushed, as if was a result from a regular DCBX negotiation flow.

Table 38-403.Set local LLDP MIB Command

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ section for details.
Opcode	2-3	0x0A08	Command opcode.
Datalen	4-5		Length of response buffer.
Return Value/VFID	6-7		Return value. Zeroed by the software device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Type	16	MIB Type	Bit 0: 0 = Local DCBX MIB (sent by device). Bits 1-7 = Reserved.
Reserved	17	Reserved	Reserved.
Local MIB Length	18-19	Len	Length of the command buffer.
Reserved	20-23	Reserved	Reserved.
Data Address High	24-27		Address of command buffer.
Data Address Low	28-31		Address of command buffer.

Table 38-404.Set Local LLDP MIB Response

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ section for details.
Opcode	2-3	0x0A08	Command opcode.
Datalen	4-5		Length of LLDP MIB if status = success. In bytes.
Return Value/ VFID	6-7		Return value. Zeroed by the software device driver. Written by firmware. Status of request. A value of success means that command was performed successfully. Error codes: EPERM - If any local LLDP MIB set in the command doesn't exist (such as if the specific agent was stopped). EINVAL - If a DCBx TLV is missing in the pushed MIB or if the TLVs pushed are not consistent or illegal.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Type	16	MIB Type	Bit 0: 0 = MIB is silently changed. Event to software is not required. 1 = MIB is changed, MIB change event is triggered. Bits 1-7 = Reserved.
Reserved	17-31	Reserved	Reserved.

Stop/start a Specific LLDP Agent

This command, posted on the ATQ, is a direct AQ command. The software device driver uses this command to request that firmware stops or restarts the DCBX agent on the port. In the future, the command could be extended to support more specific agents.

Stopping the DCBX agent on the port means the following:

1. When parsing the LLDP TLVs received from the peer, the LLDP agent skips over all DCBX TLVs.
2. LLDP agent does not send DCBX TLVs to the peer.
3. Local DCB configuration is returned to the hardware default (single TC, no PFC).
4. Get LLDP MIB will still return the remote DCBX MIBs if such are received from the peer.

Restarting the DCBX agent on the port means the following:

1. LLDP agent parses and processes the DCBX TLVs received from the peer.
2. LLDP agent sends DCBX TLVs to the peer.
3. Local DCB configuration is modified according to the DCBX negotiation with peer.

Firmware writes back the status of the request.

Table 38-405.Stop/Start a Specific LLDP Agent Command

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ section for details.
Opcode	2-3	0x0A09	Command opcode
Datalen	4-5	0	Direct command; no response buffer.
Return Value/ VFID	6-7		Return value. Zeroed by the software device driver. Written by firmware.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Command	16	Cmd	Bit 0 = Command. 0 = Stop DCBX agent. 1 = Restart DCBX agent. Bits 1-7 = Reserved.
Reserved	17-31	Reserved	Reserved

Table 38-406.Stop/Start a Specific LLDP Agent Command Response

Name	Bytes.Bits	Value	Remarks
Flags	1:0		See AQ section for details.
Opcode	2-3	0x0A09	Command opcode.
Datalen	4-5	0	Direct command; no response buffer.
Return value/ VFID	6-7		Return value. Zeroed by the software device driver. Written by firmware. Status of request. A value of success means that the command was performed successfully.
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.

Table 38-406.Stop/Start a Specific LLDP Agent Command Response

Name	Bytes.Bits	Value	Remarks
Status	16	Cmd	Bit 0: Status. 0 = DCBX agent stopped. 1 = DCBX agent active. Bits 1-7 = Reserved.
Reserved	17-31	Reserved	Reserved.

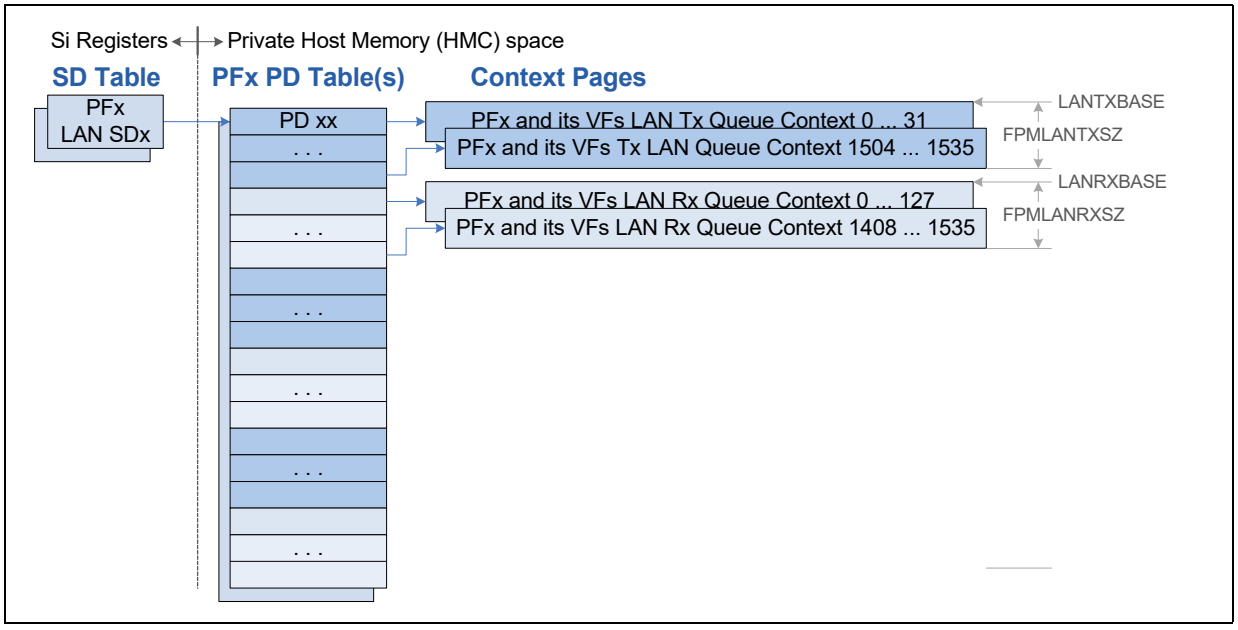
38.30 LAN Engine

38.30.1 LAN Cache and Private Host Memory

The LAN transmit and receive queue contexts are stored in memory pages in the FPM space detailed in Section 38.26. The FPM for the Tx and Rx queue contexts is defined by the GLHMC_LANTXBASE, GLHMC_LANRXBASE, GLHMC_LANTXOBJSZ and GLHMC_LANRXOBJSZ registers. The context of the PF and its VFs reside in the private memory spaces of the PF. A conceptual description of these structures (assuming each object starts at a new PD) is shown in Figure 38-92.

The FPM is considered a private memory of the hardware so software is not expected to access it other than in LAN queue contexts for initial programming (as described in this section). During run time, the hardware fetches the queue context from the FPM to its cache according to internal needs.

Figure 38-92.PF LAN FPM



38.30.2 Shared LAN Data Path

38.30.2.1 LAN Queue Pairs Allocation

38.30.2.1.1 LAN Queue Pairs Allocation to PFs

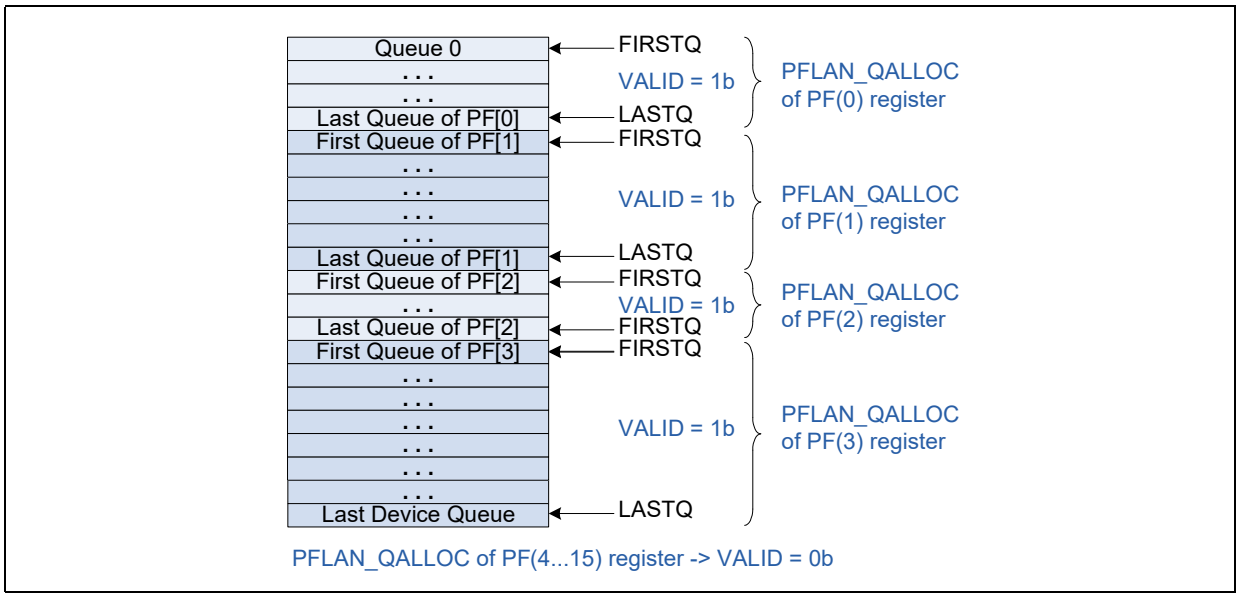
Queues are allocated in pairs of transmit and receive queues, called LAN Queue Pairs (LQP) in this section. LQPs are statically allocated to PFs in a flexible manner according to PFLAN_QALLOC registers (loaded from the NVM):

- Active PFs must have LQPs enabled by the VALID flag.
- LQPs of a PF start at the absolute LQP index defined by the FIRSTQ field and end at the absolute LQP index defined by the LASTQ field.

The allocated LQPs for the PFs must be defined monotonically. This means that LQPs of PF(n) must be allocated after the LQPs of PF(n-1) in the absolute physical device space. Figure 38-93 provides an example of 4 PFs.

Note: Based on the PFLAN_QALLOC values, the PF software allocates the LQPs for its usage and for its VFs as detailed in the following sub-sections.

Figure 38-93.LAN Queue Pairs Allocation Example of 4 PFs



38.30.2.1.2 LAN Queue Pairs Allocation Within a PF

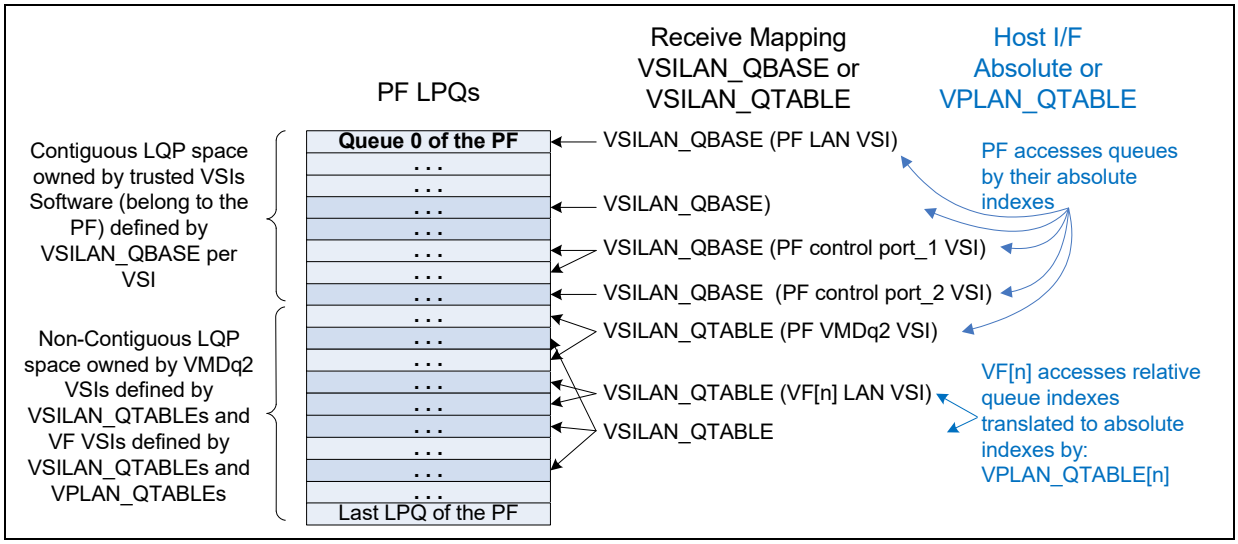
The PF software device driver is expected to read the values of the PFLAN_QALLOC register to determine the LQPs owned. The PF allocates these LQPs by programming the VPLAN_QTABLEs and VPLAN_QBASE registers and by the add VSI AQ command that programs the VSILAN_QBASE and VSILAN_QTABLEs. An example of LQP allocation within a PF to its VSIs is shown in Figure 38-94. Note the following:

- Receive queue zero of the PF is reserved for the flow director filter programming status descriptor. Note that LPQ zero must be a member of a VSI. If it is needed to avoid mixed traffic and flow director status indications on the same receive queue, this VSI should not be the default VSI of the port and should not be a target of any switch filters.
- VFs used by the VMM are expected to be statically allocated and therefore could be contiguous. Contiguous space is defined by the VFFIRSTQ and VFNUMQ field in the

VPLAN_QBASE register. VFs that are allocated contiguous LQPs in the PF space can get up to 256 LQPs.

- Non-statically allocated VFs and VMDq2 VSIs are allocated up to 16 scattered LQPs in the PF space by the VPLAN_QTABLE, and VSILAN_QTABLE registers. The VSILAN_QTABLE registers enable scattered LQPs allocation useful for dynamic VM motion.
  - For VFs, the *VSILAN_QTABLE* registers of all its VSIs must be set to the same indexes assigned to the VF by the *VPLAN_QTABLE* registers.
  - These tables are not readable for the VFs. Instead, the VF is notified by the PF about the number of its allocated LQPs (using a software API or the mailbox). The PF on its side, must allocate the LQPs starting at LQP zero and setting contiguous entries in these tables.
  - Mapping of a *VSILAN_QTABLE*[m] to a VF is the same for all other VSI registers defined by the VSI_VSI2F registers.
- VSIs allocated to nominal PF traffic or PF control ports are expected to be statically allocated and therefore could be contiguous. Contiguous space is defined by the *VSIBASE* field in the VSILAN_QBASE register. VSIs that are allocated contiguous LQPs in the PF space can get any number of LQPs.
  - The size of the contiguous VSI is not enforced by hardware setting so it is software's responsibility to keep within the VSI boundaries.
  - During reception, hardware checks that the queue index generated by the receive classification filters does not exceed the PF queue range. However, it does not check if the queue index exceeds the VSI range. It is PF software's responsibility to define the receive classification filters so that the queues do not exceed the VSI space.
- Contiguous vs. scattered VSI is controlled by the VSIQTABLE_ENA flag in the VSILAN_QBASE register.
- It is PF software's responsibility to define free LQPs that are within the space of the PF.

Figure 38-94.PF Queues (Example)



Software Access to the Queues of the Functions

PF software accesses its LQPs by their index within the PF space. Queue index 'm' of PF 'n' equals to 'm + PFLAN_QALLOC[n].FIRSTQ' in the device space. Accessing LQPs outside the PF space does not impact device functionality. Write accesses are terminated successfully on the PCIe bus with no impact on the hardware while read accesses provide meaningless data.

VF software accesses its LQPs using an index within the VF space. LQPs indexes are translated to indexes in the PF space by the VPLAN_QTABLE and VPLAN_QBASE registers. The VPLAN_QTABLE supports up to 16 LQPs. When using a contiguous allocation via the VPLAN_QBASE, up to 256 queues can be accessed. A VF might get less than the maximum number of LQPs. A VF software attempt to access LQPs above the allocated ones does not impact device functionality (the same as previously described for the PF).

Note that a function (PF or VF) might have multiple VSIs. Still, software accesses all LQPs by their index in the function space rather than index in the VSI spaces.

Associating Received Packets to VSI Queues

The 10 GbE controller associates received packets to VSIs by the embedded switch. It then associates packets to queues using the classification filters.

- As opposed to the software interface, the classification filters assign a queue index in the VSI space. The queue index 'n' is mapped to the PF space as follows:
  - Queue index 'n' of a contiguous VSI is mapped to queue index 'n + VSILAN_QBASE.VSIBASE'
  - Queue index 'n' of a "scattered" VSI is mapped as follows:
    - For an even 'n' it is mapped to VSILAN_QTABLE[n/2].QINDEX_0
    - For an odd 'n' it is mapped to VSILAN_QTABLE[(n-1)/2].QINDEX_1
- Invalid receive queues
  - An invalid queue for contiguous VSI is identified if n + VSILAN_QBASE.VSIBASE exceeds the PF queue space.
  - An invalid queue 'n' for scattered VSI is identified if its QINDEX in the VSILAN_QTABLE equals to 0x3FF or if 'n' is greater or equal than 16.
- Packets received to invalid queues are dropped and counted by the GLV_REPC counter of the VSI.

Dynamic Queue Allocation

The 10 GbE controller supports dynamic queue allocation between VSIs of each PF. Dynamic queue allocation is mainly aimed to support dynamic load balance of VFs according to needs due to VM motion. Queues can be de-allocated from an active VSI during run time. These queues can then be allocated to another VSI of the same VF or to another VF. The entire flow is managed by the PF as described in the text that follows.

The following steps relate to LQPs of VFs while it is similar to LQPs of VMs:

- PF software communicates to the VF that it needs to give up a specific LQP(s).
- The PF removes the queue(s) from the VPLAN_QTABLE, *VPLAN_QBASE* and the VSILAN_QTABLE. As a result, the VF can no longer access the queue(s).
- PF software disables the relevant queues, following the queue disable flow described in Section 38.30.3.3.3 and Section 38.31.3.1.2. As part of the flow, the PF waits for the hardware indication that the queues are disabled with no further activity to host memory.

- The PF notifies the VF that it can release the host memory structures of the removed LQP(s).
- The PF remaps these LQP(s) to another VF as follows:
  - It programs the new queue context parameters in the FPM and then enables these queues (see Section 38.30.3.3.2 and Section 38.31.3.1.2 for receive and transmit queue enablement flows).
  - It maps the LQP(s) to the new VF in the VPLAN_QTABLE, VPLAN_QBASE and VSILAN_QTABLE and informs the VF about this action.
- The queues are ready to be used by the VF.

38.30.2.1.3 LAN Queue Pair Allocation Example

Allocate the LQPs to the VSIs of the PF by setting the VSILAN_QBASE registers, VSILAN_TABLE and VPLAN_TABLE. Table 38-407 lists 5 VSIs while the PF owns 128 LQPs (as configured by PFLAN_QALLOC.FIRSTQ = 128 and PFLAN_QALLOC.LASTQ = 255).

Table 38-407.5 VSIs While the PF Owns 128 LQPs Example

VSI #	VSI Usage	Requirement	Setting
None	N/A	1 LQP	Allocating dedicated receive queue for the flow director filter programming status descriptors. Queue index 0 in the PF space is absolute queue index 128.

0	PF Control Port	1 LQP	VSILAN_QBASE[0].VSIQTABLE_ENA = 0 // contiguous range.
			VSILAN_QBASE[0].VSIBASE = 1.
			Queue index 1 in the PF space is absolute queue index 129.

1	PF LAN and its VMDq1 VMs	64 LQPs	VSILAN_QBASE[1].VSIQTABLE_ENA = 0 // contiguous range.
			VSILAN_QBASE[1].VSIBASE = 2.
			Queue indexes 2...65 in the PF space are absolute queue index 130...193.

2	PF Traffic	16 LQPs	VSILAN_QBASE[2].VSIQTABLE_ENA = 0 // contiguous range
			VSILAN_QBASE[2].VSIBASE = 66.
			Queue indexes 66...81 in the PF space are absolute queue index 194...209.

10	VMDq2	2 LQPs	VSILAN_QBASE[10].VSIQTABLE_ENA = 1 // scattered range.
			VSILAN_QTABLE[10,0].QINDEX_0 and QINDEX_1 = 100, 102 and QINDEX_0 and QINDEX_1 in VSILAN_QTABLE[10,1...7] = 0x7FF.
			Queues 100,102 in the PF space are absolute queues 228,230.

100	VF[20]	4 LQPs	VSILAN_QBASE[100].VSIQTABLE_ENA = 1 // scattered range.
			VSILAN_QTABLE[100,0].QINDEX_0 and QINDEX_1 = 90, 98.
			VSILAN_QTABLE[100,1].QINDEX_0 and QINDEX_1 = 110, 112.
			QINDEX_0 and QINDEX_1 in VSILAN_QTABLE[100,2...7] = 0x7FF.
			VPLAN_QTABLE[20; 0,1,2,3,4:15].QINDEX = 90, 98, 110, 112, 0x7FF.
			Queues 90,98,110,112 in the PF space are absolute queues 218,226,238,240.

38.30.2.2 LAN Initialization Flow

This section describes the LAN engine initialization flow executed by the PF software driver. It is assumed that a PF reset (PFR) was initiated by the software prior to this flow:

- LQPs are allocated to the PF by an NVM setting loaded to the PFLAN_QALLOC registers following a core reset (CORER).
- The statistic counters are cleared only at POR. As part of PF driver initialization, the software should read all the PF and its VF statistic counters. The values of these counters is the baseline for any statistics collected later.

- Operating system driver only step — Transition the device to non-PXE mode by initiating the Clear PXE Mode admin command. See the command description in Section 38.30.2.2.1 and Section 38.30.2.2.2 for more detail.
- Most of the LAN engine logic is cleared by the hardware by core reset signal (CORER). If it is not guaranteed that a CORER was initiated, software should clear the following control registers (listed in the LAN Transmit Receive Registers section) of the PF and its VFs:
  - LQPs mapping:
    - Clear the QTX_ENA, QRX_ENA, QTX_TAIL and QRX_TAIL registers of all the PF and its VFs LAN queue pairs.
    - Initialize LAN port parameters using the Set Port Parameters admin command (setting save bad packet, default VSI and more). In a single VSI per port, the VSI parameters are loaded from the NVM even though software could execute this step only if it requires another setting other than the NVM image.
- Allocate the LQPs to VFs and VSIs of the PF:
  - For each assigned VSI do the following as part of the create VSI procedure:
    - Allocate the LQPs to the VSI (see the programming example in Section 38.30.2.1.3).
    - Program the VSILAN_QBASE and VSILAN_QTABLE (use QBASE option for contiguous LQPs or the QTABLE option for scattered LQPs).
  - Program the VPLAN_QTABLE and VPLAN_QBASE of each assigned VF.
- Enable individual receive and transmit queues of the PF and its VFs following the flow described in Section 38.30.3.3.2 and Section 38.31.3.1.1, respectively.

38.30.2.2.1 Clear PXE Mode Admin Command

The Clear PXE Mode admin command transitions the device from PXE to non-PXE mode. The structure of the admin command and its completion are listed in Table 38-408. The complete software and device response is followed in Section 38.30.2.2.2.

Table 38-408.Clear PXE Mode Admin Command (Opcode: 0x0110)

Name	Bytes.Bits	Value	Remarks
Flags	0-1		See AQ section for details.
Opcode	2-3	0x0110	Command opcodes.
Datalen	4-5	0x00	N/A (reserved zero).
Return Value/ VFID	6-7	0x00	N/A (reserved zero).
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Reserved	16	0x2	Reserved 0x2.
Reserved	17-31		Reserved.

Table 38-409.Clear PXE Mode Admin Command Completion (Opcode: 0x0110)

Name	Bytes.Bits	Value	Remarks
Flags	0-1		See AQ section for details.
Opcode	2-3	0x0110	Command opcode.
Datalen	4-5	0x00	N/A.
Return Value/ VFID	6-7		Some comments on specific errors: 0x0 = No Error. 0xD = EEXIST (no action, the device is already in non-PXE mode).
Cookie High	8-11	Cookie	Opaque value is copied by firmware into the completion of this command.
Cookie Low	12-15	Cookie	Opaque value is copied by firmware into the completion of this command.
Reserved	16-31		Reserved.

38.30.2.2.2 Transitioning Flow to Non-PXE Mode

Device Response to Clear the PXE Mode Admin Command

1. When the Clear PXE Mode admin command is initiated, the device checks the value of the PXE_MODE flag in the GLLAN_RCTL_0 register.
2. If the PXE_MODE flag is found cleared:
   a. Return a command completion with an EEXIST indication.
3. Else, the PXE_MODE flag is active:
   a. Disable Rx queue 0 and Rx queue 1 of all enabled PFs by clearing the QENA_REQ flag in the QRX_ENA[0] and QRX_ENA[1] registers of the PFs.
   b. Wait until all receive queues are disabled by polling the QENA_STAT flag in the QRX_ENA registers of all previous Rx queues. It is expected that all Rx queues are disabled within a few microseconds.
   c. Clear the PXE_MODE flag in the GLLAN_RCTL_0 register.
   d. Flow is completed by posting a command completion with No Error.

Software Steps Transitioning to Non-PXE Mode

1. The admin queue must be active before the following steps.
2. Software initiates the Clear PXE Mode admin command and waits for its completion.
3. Proceeds with the software initialization flow.

38.30.2.3 Cloud Computing Support

The cloud provides software, storage and compute infrastructure services over the Internet. Cloud platforms should provide isolation among tenants by creating private virtual networks over the physical network infrastructure and should be able to provision services on any physical machine within the data center. The 10 GbE controller provides filtering and pruning functionality that supports these goals.

Cloud providers build virtual network overlays over existing network infrastructure that provide tenant isolation and scaling. Tunneling layers added to the packets carry the virtual networking frames over existing Layer 2 and IP networks. Conceptually, this is similar to creating virtual private networks over the Internet. Processing these tunneling layers by the hardware is a critical element of this solution.

The tunneling packet formats supported by the 10 GbE controller are shown in Figure 38-95. Refer to the transmit and receive descriptor sections for requested offloads on transmission and reported status on reception.

The 10 GbE controller should be set for tunneling mode or non-tunneling mode by an NVM image. NVM images are used to set the VSI filters and receive classification filters. There are three planned NVM images:

1. Non-tunneling.
2. MAC in UDP tunneling.
3. IP in IP, IP in GRE and MAC in GRE tunneling.

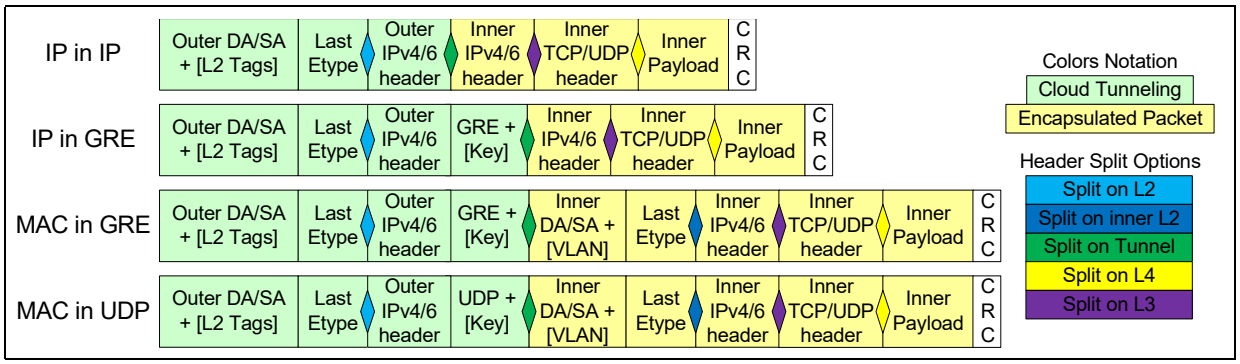
An overview of the packet processing is listed in Table 38-410.

Table 38-410.Tunneled Packet Processing

Packet Type	Accept / Reject Packet	Forward Packet To Specific VSI (Based On...)	Identify, Insert / Strip L2 Tags	L4/IPv4 XSUM	TSO	Tunneling Header Split (On Top Of Non-tunneling Options)	RSS (Hash) And FD Filters
IP in IP	Based on outer L2 MAC and optional L2 Tags (including VLAN, QinQ or S Tag with/without VLAN)	Outer IP	L2 tags in the outer L2 headers Only single VLAN in the inner L2 header	Both inner and outer IP headers, Inner L4 header	Yes	Outer IP header	Yes (same as non-tunneled packets)
IP in GRE		Outer IP with/ without GRE key				GRE header	
MAC in GRE		Inner L2 MAC/VLAN with/without one of the following: GRE key; Outer IP; Outer MAC				GRE header	
MAC in UDP		Inner L2 MAC/VLAN with/without VN Key words				Outer UDP header and its extension header	

The Figure 38-95 shows the supported tunneled packet formats:

Figure 38-95.Tunneled Packet Formats



38.30.2.4 Steering Tag and Processing Hint Support for LAN Engine Traffic (TPH)

Refer to the TLP processing hint support section for more detail. The Table 38-411 lists how a steering tag and processing hints are generated and how TPH operation is enabled for types of DMA traffic associated with the LAN queues.

Table 38-411.Steering Tag and Processing Hint Programming by the LAN Engine

Traffic Access	Steering Tag Value And TPH Enablement	PH Value
Read Receive Descriptor	CPUID and TPHRDesc in the Rx Queue Context	Desc_PH in GLTPH_CTRL register
Write Back Receive Descriptor	CPUID and TPHWDesc in the Rx Queue Context	Desc_PH in GLTPH_CTRL register
Write Receive Packet Payload	CPUID and TPHData in the Rx Queue Context	DATA_PH in GLTPH_CTRL register
Write Receive Packet Header	CPUID and TPHHead in the Rx Queue Context	DATA_PH in GLTPH_CTRL register
Read Transmit Descriptor	CPUID and TPHRDesc in the Tx Queue Context	Desc_PH in GLTPH_CTRL register
Write Back Transmit Descriptor	CPUID and TPHWDesc in the Tx Queue Context	Desc_PH in GLTPH_CTRL register
Transmit Head Write back	CPUID and TPHWDesc in the Tx Queue Context	Desc_PH in GLTPH_CTRL register
Read Transmit Packet	CPUID and TPHRPacket in the Tx Queue Context	DATA_PH in GLTPH_CTRL register

38.30.3 LAN Receive Data Path

The LAN receive data path includes the following major topics:

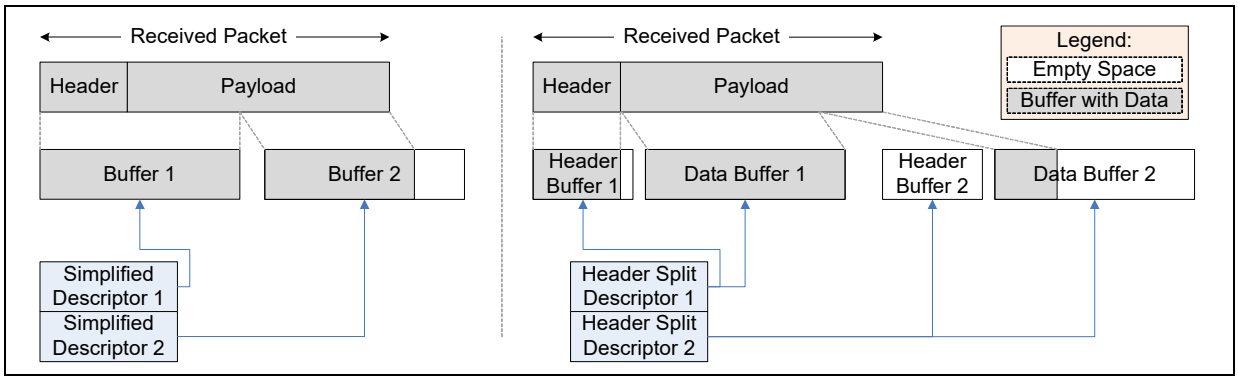
- Receive packets stored in system memory.
- Indicating the free memory structures to hardware and indicating the completed structure back to software.
- Receive descriptor queues, which are also called descriptor ring.
- Receive arbitration.
- Stateless receive offloads.

38.30.3.1 Receive Packet in System Memory

Receive packets are posted to system (host) memory buffers indicated to the hardware by descriptors. There are several types of descriptors that include pointers to the data buffers and status indications of the received packets. Figure 38-96 shows two examples of receive packets in host memory composed of two buffers (indicated by two matched descriptors). The 10 GbE controller fetches the receive descriptors (on demand) to an internal cache. A few rules relating to receive packet posting to host memory are:

- Receive packets might span from one to five buffers (descriptors).
- Receive packets shorter than 64 bytes are never posted to host memory (even in save bad frame mode enabled by the *SBP* flag in the PRT_SBPVSI register). These packets are counted by the GLPRT_MSPDC counter per LAN port.

Figure 38-96.Receive Packet in System Memory



38.30.3.1.1 Receive Descriptor Cache

Descriptor Fetch Policy

The 10 GbE controller fetches multiple receive descriptors at a time in order to minimize PCIe and memory bandwidth overhead (eight or four descriptors when using 16- or 32-byte descriptors, respectively. New descriptors are fetched to the cache, when there are fewer descriptors than incoming packets require or the last free descriptor is used for a received packet.

Note the following rules relating descriptor fetch policy:

- Following a CORE reset, hardware wakes up in PXE mode (the PXE_MODE flag in the GLLAN_RCTL_0 is set). PXE mode functionality and limitations are:
  - The receive queue should not be larger than 16 x 16 byte descriptors.
  - Software can bump the tail at descriptor granularity. Hardware fetches and writes back these descriptor at descriptor granularity as well.
  - Each packet might span only on a single buffer (in a single descriptor). A receive packet that is larger than a single buffer is reported as OVERSIZE in the receive descriptor.
- During normal operating mode, the PXE_MODE flag must be cleared by software. This step is expected to occur during the PF software initialization procedure. In case of multiple active PFs, only the first PF affects the device setting while the others do nothing.
- When the PXE_MODE flag is cleared, software should bump the tail at the entire 8 x descriptors granularity. In this mode, hardware fetches descriptors in the entire cache lines (4 x 32 byte descriptors or 8 x 16 byte descriptors).

38.30.3.2 LAN Receive Descriptors

38.30.3.2.1 Receive Descriptor - Read Format

16-Byte Receive Descriptors Read Format

Table 38-412 lists the 16-byte receive descriptor read format prepared by software.

Table 38-412.16-Byte Receive Descriptors Read Format

Quad Word	63 ... 0
0	Packet Buffer Address
1	Header Buffer Address

Packet buffer address (64) — The physical address of the packet buffer defined in byte units. The packet buffer size is defined by the DBUFF parameter in the receive queue context.

Header buffer address (64) — The physical address of the header buffer defined in byte units. The header address should be set by software to an even number (word-aligned address). The header buffer address is meaningful only for header split queues and split always queues as defined by the DTYPE field in the receive queue context. If a received packet spans across multiple buffers, only the first descriptor's header buffer is used. The header buffer size is defined by the HBUFF parameter in the receive queue context.

Note that the LS bit should be set to zero regardless of header split enablement since it is used for Descriptor Done (DD) indication to software (as described in the descriptor write back format).

32-Byte Receive Descriptors Read Format

Following is the 32-byte receive descriptor read format prepared by software.

Table 38-413.32-Byte Receive Descriptors Read Format

Quad Word	63 ... 0
0	Packet Buffer Address
1	Header Buffer Address
2	Reserved (0x0)
3	Reserved (0x0)

The fields in first 16 bytes — Identical to the 16-byte descriptors previously described.

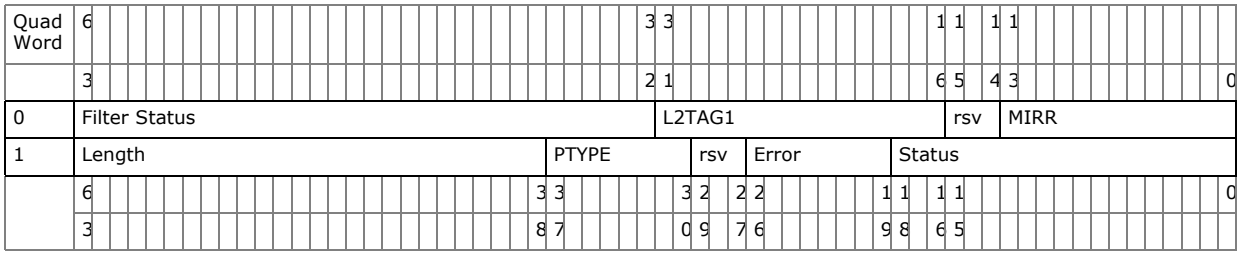
38.30.3.2.2 Receive Descriptor – Write-Back Format

The following sub-sections describe the fields of receive descriptor write back when using 16-byte and 32-byte descriptors. In both cases, a single packet might span on a single buffer or multiple buffers reported by their matched descriptors. If a packet is described by a single descriptor then all the fields are valid. Following are some rules that apply for a packet that is described by multiple descriptors:

- These fields are valid in all descriptors of a packet: Done (DD) flag, End Of Packet (EOP) flag and Packet Content Length (PKTL) field.
- These fields are valid only in the first descriptor of a packet: HDRL (packet content length in the header buffer); SPH (header is identified for header split functionality) and HBO (header buffer overflow).
- All other fields are valid only in the last descriptor of a packet.

16-Byte Receive Descriptors Write-Back Format

Following is the 16-byte receive descriptor write-back format.



RSV (7 bits) — Reserved.

Status field — Quad Word 1, 19 bits.

Table 38-414.Status Bits (Sheet 1 of 2)

Bits	Name	Functionality
0	DD	Descriptor done indication flag.
1	EOP	End of packet flag is set to 1b indicating that this descriptor is the last one of a packet.
2	L2TAG1P	L2 TAG 1 presence indication while the L2 tag is stripped from the packet and reported in the L2TAG1 field in the descriptor. The type of the tag is defined by the L2TSEL flag in the queue context. The L2TSEL flag selects between the first or second active flags in the SHOWTAG field in the VSI_TSR register of the VSI. The structure of this tag is defined by the matched GL_SWT_L2TAGxxx registers. If the specified L2 tag is not present in the packet the L2TAG1P flag is cleared.
3	L3L4P	For IP packets, this flag indicates that L3 and L4 integrity check is processed by the hardware. For a tunneled packet, the L3L4P relates to the inner IP header. Processing is according to the packet type, which is reported in the PTYPE field. See Section 38.30.4.2 for a description of the supported integrity checks.
4	CRCP	CRCP indicates that the Ethernet CRC is posted with data to the host buffer. Note that strip CRC is enabled by the CRCStrip flag in the queue context. If the RXE error flag is set, the CRC bytes are not stripped regardless of the CRCStrip flag in the queue context. Loopback packets originated by another local VSI for which the HW computes the CRC are never posted with the CRC bytes regardless of the CRCStrip setting in the queue context.
7:5	TSYNINDX	Sampled 1588 timestamp index. bit 7 is the TSYNINDX valid indication. Only when it is set to 1b the lower 2 bits are meaningful. bit 5:6 is the index of the PRTTSYN_RXTIME Register that holds the packet reception timestamp. Note that 1588 packets over UDP can be sampled regardless of a possible checksum error.

Table 38-414.Status Bits (Sheet 2 of 2)

Bits	Name	Functionality
8	EXT_UDP_0	This flag is set for received tunneled packets with outer UDP header on which the outer UDP checksum word equals to zero. Note that UDP checksum zero is an indication that there is no checksum. This option is valid only for IPv4 packets and considered an exception error for IPv6 packets (reported to the stack by the software device driver).
10:9	UMBCAST	Destination address can be one of the following: 00b = Unicast 01b = Multicast 10b = Broadcast 11b = Mirrored packet Non-parsed packets are indicated by PTYPE equals to PAYLOAD (non identified MAC header).
11	FLM	Flow director filter match indication. This flag is set if the received packet matches any of the Flow Director (FD) filters that direct the packet to a specific receive queue. Refer to the RSSSTAT field description for more detail.
13:12	FLTSTAT	The FLTSTAT indicates the reported content in the *Filter Status* field. FLTSTAT has the following encoding (see conditions in the *Filter Status* field): 00b = No Data in the filter status field (The packet does not meet any of the cases below). 01b = FD filter ID (this option is valid only for 16B descriptor while in 32B it is reported elsewhere). 10b = Reserved. 11b = Hash filter signature (RSS).
14	LPBK	Loopback indication that means that the packet is originated from this system rather than the network.
15	IPV6EXADD	Set when an IPv6 packet contains a destination options header or a routing header (See additional details in Section 38.30.4.2.) If the packet contains two IPv6 headers (tunneling), the IPV6EXADD is a logic OR function of the two IP headers.
17:16	Reserved	Reserved.
18	INT_UDP_0	This flag is set for received UDP packets on which the UDP checksum word equals to zero. Note that UDP checksum zero is an indication that there is no checksum. This option is valid only for IPv4 packets and considered an exception error for IPv6 packets (reported to the stack by the software device driver). Note that for tunneled packets with a UDP header, this flag relates to the checksum field in the inner UDP header.

Error field — Qword 1, 8 bits.

Table 38-415.Error Bits

Bits	Name	Functionality
0	RXE	The RXE error bit is an indication for any of the following MAC errors: CRC, alignment, oversize, undersize, or length error. Packets with RXE are posted to host memory to Rx queue 0 of the VSI defined by the PRT_SBPVSI register, if enabled by the SBP flag in the same register. If the RXE flag is set, then any other status fields reporting the content of the packet are meaningless.
1	RSV	Reserved.
2	HBO	Header Buffer Overflow. This flag is set when using header split buffers or split always buffers and the identified packet header is larger than the header buffer. See Table 38-429 for details.
5:3	L3L4E	For IP packets processed by hardware, the L3L4E flag has the following encoding: Bit 3 = IPE: IP checksum error indication (for tunneled packets it is the most inner IP header indication). Bit 4 = L4E: L4 integrity error indication (most inner L4 header in case of UDP tunneling). Bit 5 = EIPE: External (most outer) IP header or UDP checksum error. ***Note:*** For the purpose of these bits, a tunneled packet is a VXLAN?NVGRE/Geneve packet.
6	OVERSIZE	Oversize packet error indicates that the packet is larger than five descriptors in normal operation or larger than one descriptor in PXE mode. In this case, the portions of the packet that exceeds the permitted number of descriptor(s) is not posted to host memory.
7	RSV	Reserved.

MIRR (Qword 0, 14 bits) — Multiplexed field between mirroring parameters as listed in Table 38-416. If the packet is not mirrored then this field equals zero.

Table 38-416.Mirroring Functionality

Bits	Name	Mirroring Functionality (UMBCAST field equals to 11b)
7:0	RSV	Reserved (equals 0x7 for mirrored packets).
13:8	MIRR	Matched mirror rule ID that directed the packet to this queue.

L2TAG1 (Qword 0, 16 bits) — Stripped L2 tag from the receive packet. This field is valid if the L2TAG1P flag in this descriptor is set (see additional description of the L2TAG1P flag).

Filter status (Qword 0, 32 bits) — Multiplexed field between RSS (hash filter) and FD filter ID as indicated by the FLTSTAT field. Note that the FD filter ID is reported in this field only in 16-byte descriptors. In 32-byte descriptors, the data is reported in a different field.

- If the packet matches a FD filter that enables its FD filter ID reporting while using a 16-byte descriptor, then FLTSTAT equals 01b and this field contains the programmed FD filter ID.
- Else, if the packet matches the hash filter, then FLTSTAT equals 11b and this field contains the hash signature (RSS).
- Else, FLTSTAT equals 00b and this field is set to zero.

Length — Qword 1, 26 bits.

Table 38-417.Length Bits

Bits	Name	Functionality
13:0	PKTL	Packet content length in the packet buffer defined in byte units.
24:14	HDRL	Packet content length in the header buffer defined in byte units.
25	SPH	The split header flag is an indication that the device identified the packet header. See Section 38.30.4.1 for a complete description of packet types identified for header split and conditions for usage of the header and data buffers.

PTYPE (Qword 1, 8 bits) — *Packet Type* field encode supported packet types as listed in Table 38-418. Note that in Table 38-418, any UDP tunneling (Teredo, VXLAN) and GRE tunneling are reported as GRENAT.

Table 38-418.Packet Types (Sheet 1 of 4)

PTYPE	Description	PTYPE	Description
L2 Packet Types			
0	Reserved	11	MAC, ARP
1	MAC, PAY2	12	MAC, PAY3
2	MAC, TimeSync, PAY2	13	MAC, FCDATA, PAY3
3	MAC, FIP, PAY2	14	MAC, FCRDY, PAY3
4	Reserved	15	MAC, FCRSP, PAY3
5	Reserved	16	MAC, FCOTHER, PAY3
6	MAC, LLDP, PAY2	17	MAC, VFT, PAY3
7	MAC, ECP, PAY2	18	MAC, VFT, FCDATA, PAY3
8	Reserved	19	MAC, VFT, FCRDY, PAY3
9	Reserved	20	MAC, VFT, FCRSP, PAY3

Table 38-418.Packet Types (Sheet 2 of 4)

PTYPE	Description	PTYPE	Description
10	MAC, EAPOL, PAY2	21	MAC, VFT, FCOTHER, PAY3
Non Tunneled IPv4		Non Tunneled IPv6	
22	MAC, IPV4FRAG, PAY3	88	MAC, IPv6+IPV6FRAG, PAY3
23	MAC, IPV4, PAY3	89	MAC, IPV6, PAY3
24	MAC, IPV4, UDP, PAY4	90	MAC, IPV6, UDP, PAY4
25	Reserved	91	Reserved
26	MAC, IPV4, TCP, PAY4	92	MAC, IPV6, TCP, PAY4
27	MAC, IPV4, SCTP, PAY4	93	MAC, IPV6, SCTP, PAY4
28	MAC, IPV4, ICMP, PAY4	94	MAC, IPV6, ICMP, PAY4
IPv4 --> IPv4		IPv4 --> IPv6	
29	MAC, IPV4, IPV4FRAG, PAY3	36	MAC, IPV4, IPv6+IPV6FRAG, PAY3
30	MAC, IPV4, IPV4, PAY3	37	MAC, IPV4, IPV6, PAY3
31	MAC, IPV4, IPV4, UDP, PAY4	38	MAC, IPV4, IPV6, UDP, PAY4
32	Reserved	39	Reserved
33	MAC, IPV4, IPV4, TCP, PAY4	40	MAC, IPV4, IPV6, TCP, PAY4
34	MAC, IPV4, IPV4, SCTP, PAY4	41	MAC, IPV4, IPV6, SCTP, PAY4
35	MAC, IPV4, IPV4, ICMP, PAY4	42	MAC, IPV4, IPV6, ICMP, PAY4
IPv4 --> GRE/Teredo/VXLAN			
43	MAC, IPV4, GRENAT, PAY3		
IPv4 --> GRE/Teredo/VXLAN --> IPv4		IPv4 --> GRE/Teredo/VXLAN --> IPv6	
44	MAC, IPV4, GRENAT, IPV4FRAG, PAY3	51	MAC, IPV4, GRENAT, IPv6+IPV6FRAG, PAY3
45	MAC, IPV4, GRENAT, IPV4, PAY3	52	MAC, IPV4, GRENAT, IPV6, PAY3
46	MAC, IPV4, GRENAT, IPV4, UDP, PAY4,	53	MAC, IPV4, GRENAT, IPV6, UDP, PAY4,
47	Reserved	54	Reserved
48	MAC, IPV4, GRENAT, IPV4, TCP, PAY4	55	MAC, IPV4, GRENAT, IPV6, TCP, PAY4
49	MAC, IPV4, GRENAT, IPV4, SCTP, PAY4	56	MAC, IPV4, GRENAT, IPV6, SCTP, PAY4
50	MAC, IPV4, GRENAT, IPV4, ICMP, PAY4	57	MAC, IPV4, GRENAT, IPV6, ICMP, PAY4
IPv4 --> GRE/Teredo/VXLAN --> MAC			
58	MAC, IPV4, GRENAT, MAC, PAY3		
IPv4 --> GRE/Teredo/VXLAN --> MAC --> IPv4		IPv4 --> GRE/Teredo/VXLAN --> MAC --> IPv6	
59	MAC, IPV4, GRENAT, MAC, IPV4FRAG, PAY3	66	MAC, IPV4, GRENAT, MAC, IPv6+IPV6FRAG, PAY3
60	MAC, IPV4, GRENAT, MAC, IPV4, PAY3,	67	MAC, IPV4, GRENAT, MAC, IPV6, PAY3,
61	MAC, IPV4, GRENAT, MAC, IPV4, UDP, PAY4	68	MAC, IPV4, GRENAT, MAC, IPV6, UDP, PAY4
62	Reserved	69	Reserved
63	MAC, IPV4, GRENAT, MAC, IPV4, TCP, PAY4	70	MAC, IPV4, GRENAT, MAC, IPV6, TCP, PAY4
64	MAC, IPV4, GRENAT, MAC, IPV4, SCTP, PAY4	71	MAC, IPV4, GRENAT, MAC, IPV6, SCTP, PAY4
65	MAC, IPV4, GRENAT, MAC, IPV4, ICMP, PAY4	72	MAC, IPV4, GRENAT, MAC, IPV6, ICMP, PAY4
IPv4 --> GRE/Teredo/VXLAN --> MAC/VLAN			
73	MAC, IPV4, GRENAT, MACVLAN, PAY3		
IPv4 --> GRE/Teredo/VXLAN --> MAC/VLAN --> IPv4		IPv4 --> GRE/Teredo/VXLAN --> MAC/VLAN --> IPv6	
74	MAC, IPV4, GRENAT, MACVLAN, IPV4FRAG, PAY3,	81	MAC, IPV4, GRENAT, MACVLAN, IPv6+IPV6FRAG, PAY3

Table 38-418.Packet Types (Sheet 3 of 4)

PTYPE	Description	PTYPE	Description
75	MAC, IPV4, GRENAT, MACVLAN, IPV4, PAY3,	82	MAC, IPV4, GRENAT, MACVLAN, IPV6, PAY3,
76	MAC, IPV4, GRENAT, MACVLAN, IPV4, UDP, PAY4	83	MAC, IPV4, GRENAT, MACVLAN, IPV6, UDP, PAY4
77	Reserved	84	Reserved
78	MAC, IPV4, GRENAT, MACVLAN, IPV4, TCP, PAY4	85	MAC, IPV4, GRENAT, MACVLAN, IPV6, TCP, PAY4
79	MAC, IPV4, GRENAT, MACVLAN, IPV4, SCTP, PAY4	86	MAC, IPV4, GRENAT, MACVLAN, IPV6, SCTP, PAY4
80	MAC, IPV4, GRENAT, MACVLAN, IPV4, ICMP, PAY4	87	MAC, IPV4, GRENAT, MACVLAN, IPV6, ICMP, PAY4
IPv6 --> IPv4		IPv6 --> IPv6	
95	MAC, IPV6, IPV4FRAG, PAY3	102	MAC, IPV6, IPv6+IPV6FRAG, PAY3
96	MAC, IPV6, IPV4, PAY3	103	MAC, IPV6, IPV6, PAY3
97	MAC, IPV6, IPV4, UDP, PAY4	104	MAC, IPV6, IPV6, UDP, PAY4
98	Reserved	105	Reserved
99	MAC, IPV6, IPV4, TCP, PAY4	106	MAC, IPV6, IPV6, TCP, PAY4
100	MAC, IPV6, IPV4, SCTP, PAY4	107	MAC, IPV6, IPV6, SCTP, PAY4
101	MAC, IPV6, IPV4, ICMP, PAY4	108	MAC, IPV6, IPV6, ICMP, PAY4
IPv6 --> GRE/Teredo/VXLAN			
109	MAC, IPV6, GRENAT, PAY3		
IPv6 --> GRE/Teredo/VXLAN --> IPv4		IPv6 --> GRE/Teredo/VXLAN --> IPv6	
110	MAC, IPV6, GRENAT, IPV4FRAG, PAY3	117	MAC, IPV6, GRENAT, IPv6+IPV6FRAG, PAY3
111	MAC, IPV6, GRENAT, IPV4, PAY3	118	MAC, IPV6, GRENAT, IPV6, PAY3
112	MAC, IPV6, GRENAT, IPV4, UDP, PAY4	119	MAC, IPV6, GRENAT, IPV6, UDP, PAY4
113	Reserved	120	Reserved
114	MAC, IPV6, GRENAT, IPV4, TCP, PAY4	121	MAC, IPV6, GRENAT, IPV6, TCP, PAY4
115	MAC, IPV6, GRENAT, IPV4, SCTP, PAY4	122	MAC, IPV6, GRENAT, IPV6, SCTP, PAY4
116	MAC, IPV6, GRENAT, IPV4, ICMP, PAY4	123	MAC, IPV6, GRENAT, IPV6, ICMP, PAY4
IPv6 --> GRE/Teredo/VXLAN --> MAC			
124	MAC, IPV6, GRENAT, MAC, PAY3		
IPv6 --> GRE/Teredo/VXLAN --> MAC --> IPv4		IPv6 --> GRE/Teredo/VXLAN --> MAC --> IPv6	
125	MAC, IPV6, GRENAT, MAC, IPV4FRAG, PAY3	132	MAC, IPV6, GRENAT, MAC, IPv6+IPV6FRAG, PAY3
126	MAC, IPV6, GRENAT, MAC, IPV4, PAY3	133	MAC, IPV6, GRENAT, MAC, IPV6, PAY3
127	MAC, IPV6, GRENAT, MAC, IPV4, UDP, PAY4	134	MAC, IPV6, GRENAT, MAC, IPV6, UDP, PAY4
128	Reserved	135	Reserved
129	MAC, IPV6, GRENAT, MAC, IPV4, TCP, PAY4	136	MAC, IPV6, GRENAT, MAC, IPV6, TCP, PAY4
130	MAC, IPV6, GRENAT, MAC, IPV4, SCTP, PAY4	137	MAC, IPV6, GRENAT, MAC, IPV6, SCTP, PAY4
131	MAC, IPV6, GRENAT, MAC, IPV4, ICMP, PAY4	138	MAC, IPV6, GRENAT, MAC, IPV6, ICMP, PAY4
IPv6 --> GRE/Teredo/VXLAN --> MAC/VLAV			
139	MAC, IPV6, GRENAT, MACVLAN, PAY3		
IPv6 --> GRE/Teredo/VXLAN --> MAC/VLAN --> IPv4		IPv6 --> GRE/Teredo/VXLAN --> MAC/VLAN --> IPv6	
140	MAC, IPV6, GRENAT, MACVLAN, IPV4FRAG, PAY3,	147	MAC, IPV6, GRENAT, MACVLAN, IPv6+IPV6FRAG, PAY3
141	MAC, IPV6, GRENAT, MACVLAN, IPV4, PAY3,	148	MAC, IPV6, GRENAT, MACVLAN, IPV6, PAY3,

Table 38-418.Packet Types (Sheet 4 of 4)

PTYPE	Description	PTYPE	Description
142	MAC, IPV6, GRENAT, MACVLAN, IPV4, UDP, PAY4	149	MAC, IPV6, GRENAT, MACVLAN, IPV6, UDP, PAY4
143	Reserved	150	Reserved
144	MAC, IPV6, GRENAT, MACVLAN, IPV4, TCP, PAY4	151	MAC, IPV6, GRENAT, MACVLAN, IPV6, TCP, PAY4
145	MAC, IPV6, GRENAT, MACVLAN, IPV4, SCTP, PAY4	152	MAC, IPV6, GRENAT, MACVLAN, IPV6, SCTP, PAY4
146	MAC, IPV6, GRENAT, MACVLAN, IPV4, ICMP, PAY4	153	MAC, IPV6, GRENAT, MACVLAN, IPV6, ICMP, PAY4
Parser Aborted Code		Else	
255	Parser Aborted	Else	Reserved

32-Byte Receive Descriptors Write-Back Format

The 32-byte descriptor is composed of four Qwords. The first two Qwords are identical to the 16-byte descriptor write back other than the FD filter ID that is moved from the *Filter Status* field to the FD Filter ID / *Flexible Bytes High* field.

Quad Word	63:44	43:38	37:32	31:30	29:20	19:0
0	Filter Status		L2TAG1	rsv	MIRR	
1	Length	PTYPE	rsv	Error	Status	
2	L2TAG2 (2nd)	L2TAG2 (1st)	rsv	rsv	Ext_Status	
3	FD Filter ID / Flexible Bytes High		Flexible Bytes Low			

Ext_Status — Qword 2, 12 bits.

Table 38-419.Ext_Status Bits (Sheet 1 of 2)

Bits	Name	Functionality
0	L2TAG2P	L2 TAG 2 presence indication while the L2 Tag is stripped from the packet to the L2TAG2 (1st) field in the descriptor. The type of tag is defined by the L2TSEL bit in the queue context. This flag selects between the first or second active flags in the *SHOWTAG* field in the VSI_TSR register of the VSI. The structure of this tag is defined by the matched GL_SWT_L2TAGxxx registers. If the stripped tag is larger than one word, then the first word is posted to the L2TAG2 (1st) field and the second word is posted to the L2TAG2 (2nd) field. If the specified L2 tag is not present in the packet the L2TAG2P flag is cleared.
1	RSV	Reserved.
3:2	FLEXBL_STAT	The FLEXBL_STAT indicates the content of the *Flexible Bytes Low* field as follows: 00b = The field is set to zero. 01b = Flexible first four bytes defined by the FD filter are posted in the *Flexible Bytes Low* field. 10b = Reserved. 11b = Reserved.
5:4	FLEXBH_STAT	The FLEXBH_STAT indicates the content of the *FD Filter ID / Flexible Bytes High* field as follows: 00b - The field is set to zero. 01b - FD filter ID is reported in the *FD Filter ID / Flexible Bytes High* field. 10b - Flexible second 4 bytes defined by the FD filter are posted in the *FD Filter ID / Flexible Bytes High* field. 11b - Reserved.

Table 38-419.Ext_Status Bits (Sheet 2 of 2)

Bits	Name	Functionality
8:6	RSV	Reserved.
9	FDLONGB	The FDLONGB flag is set if the matched FD filter's index within its bucket is above the threshold. If the packet is searched in the FD filter and it is not found, the FDLONGB represents the bucket length relative to the same threshold. The threshold is defined by the MAXFDBLEN parameter in the GLQF_CTL register.
10	Reserved	Reserved.
11	RSV	Reserved.

L2 tags — Qword 2, 32 bits.

Table 38-420.L2 Tag Bits

Bits	Name	Functionality
15:0	L2TAG2 (1st)	Extracted L2 Tag 2 from the packet (see L2TAG2P flag).
31:16	L2TAG2 (2nd)	Extracted second word of the L2 tag 2 from the packet (see L2TAG2P flag).

Flexible bytes low (Qword 3, 32 bits) — Flexible bytes as controlled by the FLEXBL_STAT parameter in this descriptor as listed in Table . If the packet matches a FD filter that enables reporting flexible bytes from the packet, then FLEXBL_STAT equals 01b. Otherwise, FLEXBL_STAT equals 00b and this field is set to zero.

Table 38-421.Flexible Bytes Low Bits

Bits	Name	FD Filter Flexible Bytes (FLEXBL_STAT = 01b)
31:0	FLEXBL	This field contains four bytes from the flexible payload in the field vector (extracted from the packet). The word offset (in the field vector) of these bytes are defined by the programmed FLEXOFF parameter of the FD filter. The reported bytes in this field are defined in little endian notation while bits 0:7 are the LS byte, which is the last byte on the wire.

FD filter ID / flexible bytes high (Qword 3, 32 bits) — Multiplexed field between the *FD Filter ID* / *Flexible Bytes High* fields as follows:

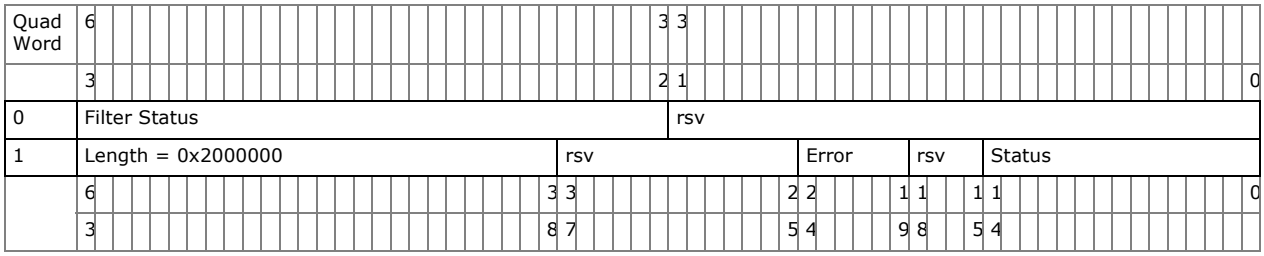
- If the packet matches a FD filter that enables reporting the FD filter ID, then FLEXBH_STAT equals 01b and this field contains the programmed FD filter ID.
- Else, if the packet matches a FD filter that enables reporting eight flexible bytes from the packet, then FLEXBL_STAT equals 10b and this field contains the four bytes from the flexible payload in the field vector (extracted from the packet). The reported bytes in this field are defined in little endian notation while bits 24:31 are the MS byte of the entire 8-byte field, which is the first byte on the wire.
- Else, FLEXBH_STAT equals 00b and this field is set to zero.

Programming Status Descriptor Write-Back Format

The programming status descriptor provides an indication of FD filter programming as listed in Table . Programming status vs. packet's descriptor is identified by the *Length* field set to 0x2000000. The programming status is provided by a 16-byte structure. When using 32-byte descriptors, only the first 16 bytes are meaningful while the last 16 bytes are reserved zeros.

Table 38-422.FD Filter Programming Indications

Context Type	Action	Fail/OK	Programming Indication	Reported Receive Queue
FD Filter	Add Filter	OK	No Reporting	N/A
FD Filter	Add Filter	Fail	Programming Status Descriptor	Queue zero of the PF.
FD Filter	Remove Filter	OK	No Reporting	N/A
FD Filter	remove Filter	Fail	Programming Status Descriptor	Queue zero of the PF.
DDP Context	Add Context	OK	No Reporting	N/A
DDP Context	Add Context	Fail	Programming Status Descriptor	Queue zero of the VSI.
DDP Context	Invalidate Context	OK	Programming Status Descriptor	Queue zero of the VSI.
DDP Context	Invalidate Context	Fail	Programming Status Descriptor	Queue zero of the VSI.



Status field (Qword 1, 15 bits) — status field.

Table 38-423.Status Field Bits

Bits	Name	Functionality
0	DD	Descriptor done indication flag.
1	RSV	Reserved.
4:2	PROG_ID	The PROG_ID indicates the status reported by this descriptor as follows: 001b = FD filter status. 010b = Reserved. 100b = Reserved.
14:5	RSV	Reserved.

Error Field — Qword 1, 6 bits.

Bits	Name	Functionality
0	Failed FD Programming	FD filter programming failed. It could be due to no space in the FD table or since the function exhausted its quota.
1	Failed FD Removal	FD filter removal failed. It could be a result of an attempt to remove non-existent entry.
3:2	Reserved	Reserved.
5:4	RSV	Reserved

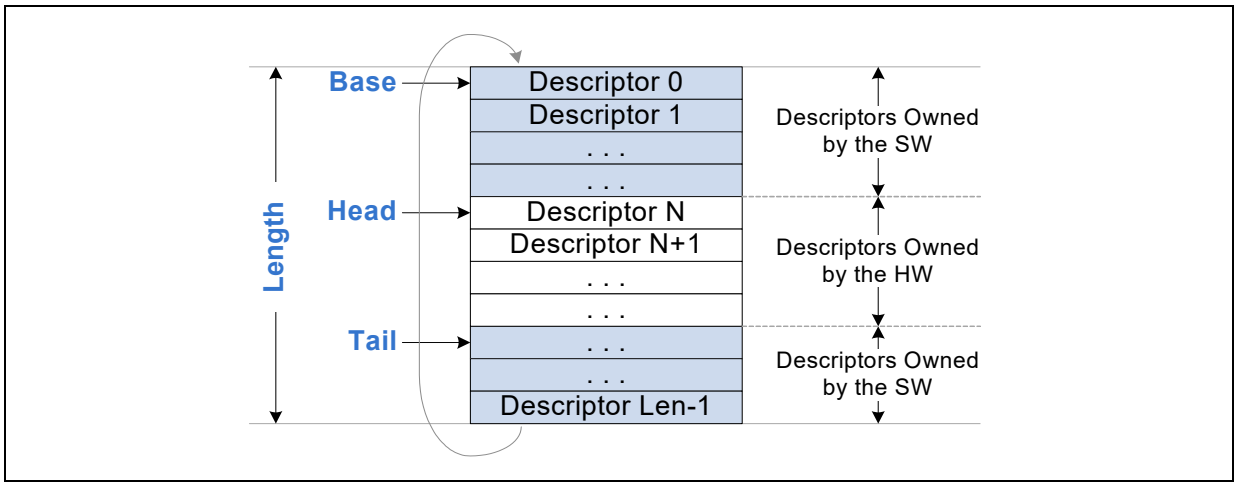
Filter status (Qword 0, 32 bits) — Equals the *FD Filter ID* field for FD filter programming status.

38.30.3.3 LAN Receive Queue (Ring)

Received packets are posted to host memory through a set of queues. Each queue is a cyclic ring made of a sequence of receive descriptors in contiguous memory. These queues are also called descriptor rings. The 10 GbE controller supports up to 1536 receive queues allocated to PFs and VFs (see Section 38.30.2). Receive queues are defined by a set of parameters called the queue context. The main parameters are the queue pointers shown in Figure 38-97. The queue context includes additional parameters that define the queue functionality as detailed in Section 38.30.3.4. Part of these context parameters are kept in hardware (like the Tail register, queue enable / disable flags and interrupt related context). Most of the queue context parameters are stored in FPM, fetched to an internal cache when required.

The software interface to the queue for its initialization, during normal operation as well as queue disable flow, is described in Section 38.30.3.3.1. The 10 GbE controller includes additional global setting option parameters for the entire device or per function that affect multiple queues described in Section 38.30.3.3.1.

Figure 38-97.Receive Descriptor Ring Structure



38.30.3.3.1 Receive Queue Programming

Queue enable and disable flows are described in the following subsection and listed in the Table 38-424.

Table 38-424.Receive Queue Enable / Disable Flags

QRX_ENA[n] Register		Receive Queue State
QENA_REQ	QENA_STAT	
0	0	Queue is not enabled.
1	0	Queue enable request by software.
1	1	Queue is enabled.
0	1	Queue disable request by software.

38.30.3.3.2 Receive Queue Enable Flow

Queue enable flow is executed by the PF software for PF queues as well as for the queues of its VFs. The flow below assumes that the queue is already allocated as detailed in Section 38.30.2.1.2.

- Software steps
- The function that owns the queue (PF or VF) allocates contiguous memory in its own memory space for the receive ring. It then program the receive descriptors in the ring so they are ready for new packet reception.
- If the queue might have been disabled recently, then the software should wait at least 50msec from the completion of the queue disable flow before proceeding to the next step.
- Prepare the queue context in the FPM in the PF memory space. See a examples for queue context setting in Section 38.30.3.4.3. At pre-boot time, memory allocation might be tight and the FPM could exceed the budget. There is a mechanism to program the queue context directly to the device bypassing the FPM as described in the section that follows.
- Clear the tail pointer in the QRX_TAIL[n] register and then set the tail pointer to the end of the descriptor ring (‘n’ is the queue index within the PF space).
- Set the QENA_REQ flag in the QRX_ENA[n] register (‘n’ is the queue index within the PF space).
- As a response, hardware sets the QENA_STAT flag in the QRX_ENA[n] register. The QENA_STAT follows the QENA_REQ almost instantly and not more than 10 µs after that. Once the QENA_STAT flag in the QRX_ENA[n] register is set, software can start using the queue.
- If the queue is targeted for a VF, PF software should also program the matched entry in the VFQ_TABLE. Then it is expected to inform the VF software that the queue is enabled. Note that the VFQ_TABLE is shared for the receive and transmit queues. So the PF software should enable the transmit queue before communicating to the VF.

Direct Queue Context Programming

This section describes direct queue context programming, bypassing the FPM for pre-boot flow.

Context programming flow:

- Write the four PFCM_LANCTXDATA registers with a total of 16 bytes of context data.
- Write the PFCM_LANCTXCTL register identifying the queue and the specific sub-line to be programmed.
- Poll the CTX_DONE flag in the PFCM_LANCTXSTAT register until done is indicated.
- Loop back to the first step until all sub-lines of the queue context are programmed.

Context invalidation flow. Note that this flow is not needed during normal operation.

- Write the following fields in the PFCM_LANCTXCTL register: QUEUE_NUM and QUEUE_TYPE, identify the queue and the OP_CODE fields should be set to Invalidate.
- Poll the CTX_DONE flag in the PFCM_LANCTXSTAT register until done is indicated.

38.30.3.3.3 Receive Queue Disable Flow

Queue disable flow is executed by PF software for PF queues as well as for the queues of its VFs.

- Remove the queue from the interrupt linked list.
- The PF software clears the QENA_REQ flag in the QRX_ENA[n] register, while 'n' is the queue index within the PF space.
- Hardware generates a queue disable marker to the receive pipe.
- Eventually, the queue disable marker gets to the top of the pipe. At this point, it is guaranteed that the pipe does not contain any additional receive packets for the specific queue.
- Hardware waits for completion of all outstanding requests from the specific queue on the PCIe bus.
- The queue context is invalidated from the internal cache without updating the FPM and the QENA_STAT flag in the QRX_ENA[n] register is cleared.
- Once the QENA_STAT flag in the QRX_ENA[n] register is cleared, software can release all memory structures of the queue.

38.30.3.3.4 Fast Receive Queue Disable Flow

Fast queue disable flow should be executed by software only as part of a VF reset flow (or VM reset flow). Following a PFR, the device does all this automatically.

- It is assumed that a VFR was initiated by the PF software and the matched VFRD flag in the VPGEN_VFRSTAT register is already active. Or a VMR was initiated by PF software and the matched VMRD flag in the VSIGEN_RSTAT register is already active.
- The PF software sets the FAST_QDIS flag (and clear the QENA_REQ flag) in the QRX_ENA[n] register where 'n' is the queue indexes in the PF space for all VF or VM queues.
- The queue contexts are invalidated instantly from the internal cache and the QENA_STAT flag in the matched QRX_ENA[n] register is cleared.

38.30.3.3.5 Software Fast Path Programming

During normal operation, the function that owns the queue (PF or VF) accesses the hardware directly.

- Prepare receive descriptors by clearing the *DD* bit and setting the buffer pointer(s). Start at the descriptor indicated by the tail pointer in the relevant QRX_TAIL register.
- Software should never set the tail to a value above the descriptors owned by the hardware minus one. The descriptors considered as owned by the hardware are those ones already indicated to hardware but not yet reported as completed.
- Bump the tail to the last prepared descriptor plus one (the descriptor index incremented the next time).
- The number of free descriptors owned by hardware is defined by the tail minus the head. If the number of free descriptors becomes lower than LRXQTRESH, then an immediate interrupt is triggered. The head and LRXQTRESH are listed in the Table 38-425.

38.30.3.4 Receive Queue Context Parameters

This section describes setting options of LAN receive queue parameters called the queue context. Some queue context parameters reside in dedicated hardware registers and some are stored in host memory (FPM). Active queues are kept in the cache.

38.30.3.4.1 Receive Queue Context in Hardware Registers

The queue context parameters that software accesses during fast past programming as well as its enablement reside in hardware registers:

- Queue enablement flags in the QRX_ENA[n] registers while 'n' is the queue index of the PF. Queue enable and disable flow by the PF are explained in Section 38.30.3.3.1.
- The tail pointer in the QRX_TAIL[n] registers for the PF and VFQRX_TAIL[n] registers for the VFs while 'n' is the queue index of the function (PF or VF). The usage of the Tail register is detailed in Section 38.30.3.3.4.

38.30.3.4.2 Receive Queue Context in FPM

The receive queue context parameters that are stored in host memory (FPM) are fetched for the active queues to the internal cache. These parameters are listed in Table 38-425. The queue context is a contiguous vector of 256 bits (32 bytes).

Table 38-425.LAN Rx Queue Context in the Private Host Memory (174 Bits = 25 Bytes) (Sheet 1 of 2)

Alias	Width [bits]	LS Bit	MS Bit	Type	Software Init	Description
HEAD	13	0	12	Dynamic	0x0	Receive Queue Head. An index relative to the beginning of the queue that defines the next descriptor to be used. During idle time, all descriptors starting by the head up to (excluding) the RTAIL are owned by hardware and the rest are owned by software. During dynamic operation it is not guaranteed that all descriptors below the head complete.
CPUID	8	13	20	Dynamic	0x0	CPU Socket ID for TPH. The CPU socket ID is updated by hardware.
RSV	11	21	31	N/A	0x0	Reserved.
BASE	57	32	88	Static	BASE	Receive Queue Base Address. Indicates the starting address of the descriptor queue defined in 12-byte units.
QLEN	13	89	101	Static	QLEN	Receive Queue Length. Defines the size of the descriptor queue in descriptors units from 8 descriptors (QLEN=0x8) up to 8 KB descriptors minus 32 (QLEN=0x1FE0). QLEN restrictions: When the PXE_MODE flag in the GLLAN_RCTL_0 register is cleared, the QLEN must be a whole number of 32 descriptors. When the PXE_MODE flag is set, the QLEN can be one of the following options: • Up to 4 PFs, QLEN can be set to: 8, 16, 24 or 32 descriptors.
DBUFF	7	102	108	Static	DBUFF	Receive Packet Data Buffer Size. The packet data buffer size is defined in 128-byte units. It must be at least 1 KB bytes and up to 16 KB minus 128 bytes.
HBUFF	5	109	113	Static	HBUFF	Receive Packet Header Buffer Size. The header buffer size is defined in 64-byte units enabling a buffer size up to 2 KB minus 64 bytes.
DType	2	114	115	Static	DType	Descriptor type as listed in Table 38-426.

Table 38-425.LAN Rx Queue Context in the Private Host Memory (174 Bits = 25 Bytes) (Sheet 2 of 2)

Alias	Width [bits]	LS Bit	MS Bit	Type	Software Init	Description
DSize	1	116	116	Static	DSize	Descriptor Size Flag. 0b for 16-byte descriptors and 1b for 32-byte descriptors.
CRCStrip	1	117	117	Static	CRCStrip	CRC Strip. Strip the Ethernet CRC bytes before the packet is posted to host memory. Note that no CRC strip option can only work properly if the entire packet is posted to the data buffer(s) in host memory with no other strip option.
Reserved	1	118	118	Static	Reserved	Reserved.
L2TSEL	1	119	119	Static	0x0	L2TSEL defines the reported L2 tags in the receive descriptor. When the L2TSEL flag is cleared, the second L2 tag defined by the *SHOWTAG* field in the VSI_TSR register of the VSI is posted to the L2TAG1 field. The first tag is reported in the L2TAG2 field. When the L2TSEL flag is set to 1b, the previous L2 tags are switched between the *L2TAG1* and *L2TAG2* fields.
HSPLIT_0	4	120	123	Static	HSPLIT_0	Header split 0 control as listed in Table 38-427.
HSPLIT_1	2	124	125	Static	HSPLIT_1	Header Split 1 control as listed in Table 38-428.
RSV	1	126	126	N/A	0x0	Reserved.
SHOWIV	1	127	127	Static	SHOWIV	The VLAN in the inner L2 header is stripped to the receive descriptor if enabled by this flag.
RSV	46	132	173	Static	0x0	Reserved
RXMAX	14	174	187	Static	RXMAX	Max packet size for this queue defined in byte units. The "rxmax" parameter defines the whole packet size starting at the L2 header up to including the Ethernet CRC. The RXMAX must not be set to a larger value than 5 x DBUFF (since receive packet must never span on more than 5 buffers). Received packet larger than RXMAX is dropped and counted by the GLV_REPC counter of the VSI. Note that packets larger than MFS defined per MAC are dropped by the MAC even before it gets to the receive queue.
TPHRDesc	1	193	193	Static	TPHRDesc	Read Descriptor TPH Ena (descriptor fetch).
TPHWDesc	1	194	194	Static	TPHWDesc	Write Descriptor TPH Ena (descriptor writeback).
TPHData	1	195	195	Static	TPHData	Packet Data TPH Ena.
TPHHead	1	196	196	Static	TPHHead	Packet Header TPH Ena.
RSV	1	197	197	Static	0x0	Reserved
LRXQTRESH	3	198	200	Static	LRXQTRESH	Low receive queue threshold defined in 64 descriptor units. When the number of free descriptors (defined by tail minus head) goes below the LRXQTRESH, an immediate interrupt is triggered.
RSV	1	201	201	Static	RSV	Reserved
RSV	54	202	255	Static	000b	Reserved.

38.30.3.4.3 Receive Queue Context Setting Example

Dword Context Offset	7	6	5	4	3	2	1	0
Context Bit Index	255 ... 224	223 ... 192	191 ... 160	159 ... 128	127 ... 96	95 ... 64	63 ... 32	31 ... 0
Init Hex Value	00000000	0000021E	01800000	00000000	00200301	00000000	001579A0	00000000

38.30.3.4.4 Receive Buffer Size Setting Example

BASE = 0x001579A0	HBUFF = 0	HSPLIT = 0 (no split|)	LRXQTRESH = 2 (no threshold)
QLEN = 0x80 (128 descriptors)	DType = 00b (no split	RXMAX = 0x600 (1536 bytes)	CRCStrip = 1
DBUFF = 12 (1536 bytes)	DSize = 0 (16 bytes)	TPH = 0xF (enable all flags)	FCENA = 0

Operating System	Queue Type	HSPLIT	DBUFF	HBUFF	Notes
Linux*	Standard	0x0	1536	-	
	Standard	enabled	2048	256	
	Jumbo	0x0	Multiple of 1 KB	-	
	Jumbo	enabled	2048	256	
Windows*	Any	0x0	Multiple of 1 KB	-	Up to 10 KB.
	VMQ	enabled	2048	64	L2 header split.
FreeBSD*	Any	0x0	2 KB, 4 KB, 8 KB, or 10K	-	
	Any	enabled	2 KB, 4 KB, 8 KB, or 10 KB	256	
ESX NPA	Any	enabled	4096	1984	max header buffer.
Solaris*	Any	0x0	256, 1 KB, 2 KB	-	
XEN*					Same as Linux.
KVM					Same as Linux.

38.30.4 Stateless Receive Offloads

38.30.4.1 Header Split

This feature consists of splitting a received packet into two separate regions based on the packet content. Splitting is usually between the packet header that can be posted to a dedicated buffer and the packet payload that can be posted to a different buffer (or multiple buffers). The size of these buffers are defined by the DBUFF and HBUFF parameters in the receive queue context. This kind of splitting is useful when different buffer allocation rules might apply to these buffers or different rules for TPH enablement. Header split is enabled per receive queue by the *DTYPE*, *HSPLIT_0* and *HSPLIT_1* fields in the receive queue context as listed in Table 38-426, Table 38-427 and Table 38-428 and shown in Figure 38-98. A split between the header buffer and the payload buffers and the status reporting is listed in Table 38-429. The physical pointers to the header and payload buffers are defined in the receive descriptor.

Some rules regarding header split:

- A packet that has a MAC error (reported by the RXE flag) is posted as a whole to the packet buffers with no split. Note that posting such packets to the host is enabled on in save bad frame mode (enabled by the SBP flag in the PRT_SBPVSI register).
- For tunneled packets, the rules defined by the HSPLIT_1 parameter take precedence on those ones defined by the HSPLIT_0 parameter. This means that if any flag in HSPLIT_1 is enabled and the packet matches that setting, then the packet is split according to HSPLIT_1 regardless of HSPLIT_0 setting.

- In each individual register: HSPLIT_1 or HSPLIT_0, the packets are split according to the lowest matched entry in the tables that follow. If both HSPLIT_0 and HSPLIT_1 are set to zero, then DTYPE in the receive queue context must be set to 00b. Otherwise (any header split is enabled by these registers), DTYPE must be set to one of the header split options: 01b or 10b (explained in the sub-sections that follow).
- If the packet is posted to multiple descriptors, only the header buffer of the first one is used.
- The packet header cannot span across buffers. If the header buffer is smaller than the received header, the header is posted together with the packet payload. See Table 38-429.
- The header of a fragmented IP packet is defined up to including the IP header regardless if the fragment includes the L4 header. For IPv6 header, the IP header is defined up to including the fragmented extension header.
- When a packet is replicated to multiple receive queues, the packet can be split differently on each according to the queues settings.
- Header split is supported for packets received from the network as well as local VM to VM traffic.

Figure 38-98.Header Split Option by Packet Formats

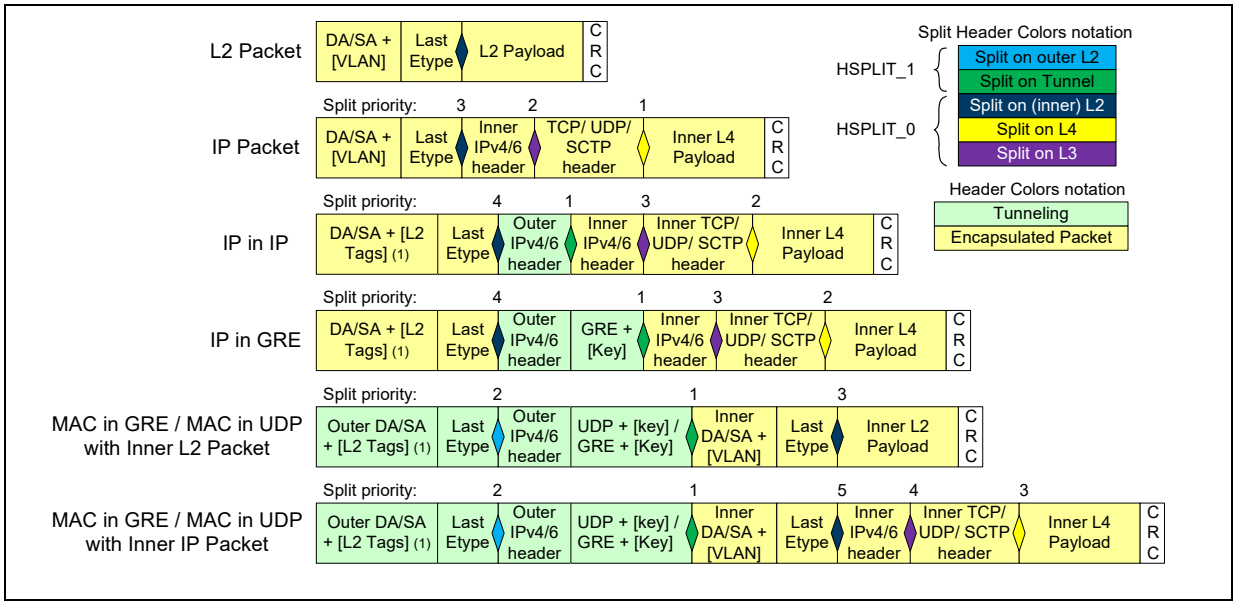


Table 38-426.Header Split Modes Defined by the DTYPE Field (Sheet 1 of 2)

DTYPE	Functionality
00b = Single buffer descriptors.	No header split mode.
01b = Header split descriptors.	Header split is enabled for packets that the hardware identifies their headers as enabled by the HSPLIT field. The packet content up to including the most inner header enabled by the HSPLIT is posted to the header buffer. The rest of the packet is posted to the payload buffer. See Table 38-429 for rules of the header and payload buffers usage.
10b = Split always	Header split always is enabled regardless of the HSPLIT setting. If the packet header is identified as defined by the HSPLIT, the packet content up to including the most inner header enabled by the HSPLIT is posted to the header buffer. See Table 38-429 for rules of the header and payload buffers usage.

Table 38-426.Header Split Modes Defined by the DTYPE Field (Sheet 2 of 2)

DTYPE	Functionality
11b	Reserved.

Table 38-427.Split Headers Enabled by the HSPLIT_0 Field

HSPLIT_0	Functionality
0000b	No header is enabled by HSPLIT_0.
xxx1b	Enable split after L2 header. In case of L2 tunneling it is the second (inner) L2 header.
xx1xb	Enable split after the IP header. In case of tunneling it is the second IP header. In case of option/extension IP headers the split is after these headers.
x1xxb	Enable split after the UDP and TCP header (in case of UDP tunneling it is the second UDP header).
1xxxb	Enable split after the SCTP header.

Table 38-428.Split Headers Enabled by the HSPLIT_1 Field

HSPLIT_1	Functionality
00b	No split on tunneling headers.
x1b	Enable split after the outer (tunneling) L2 header.
1xb	Enable split on the tunneling header as follows: Non-tunneled packet, no impact. IP in IP, enable split after the outer IP header. IP in GRE or MAC in GRE, enable split after the GRE header. MAC in UDP, enable split after the tunneling UDP header (including its private UDP header).

Table 38-429.Header Split vs. Packet and Buffer Sizes

DTYPE	Condition	Header and Payload DMA	SPH	HBO	PKTL[1]	HDRL[a,2]
00b = Single buffer.	None	Post the entire packet to the packet buffer(s).	0	0	Size of the entire packet.	0x0
01b = Header split.	Header is not identified	Post the entire packet to the packet buffer(s).	0	0	Size of the entire packet.	0x0
	Header size ≤ HBUFF[b]	Post the header to header buffer and payload to the packet buffer(s).	1	0	Size of the packet payload.	Header size.
	Header size > HBUFF[b]	Post the entire packet to the packet buffer(s).	1	1	Size of the entire packet.	Header size.
10b = Split always.	Packet length ≤ HBUFF	Post the entire packet to the header buffer(s).	0	0	0x0	Packet length.
	Header is not identified and packet length > HBUFF	Post the entire packet to the header buffer plus packet buffer(s).	0	0	Size of the entire packet minus HBUFF.	HBUFF
	Header size ≤ HBUFF	Post the header to the header buffer and the payload to the packet buffer(s).	1	0	Size of the packet payload.	Header size.
	Header size > HBUFF	Post the entire packet to the header plus packet buffer(s).	1	1	Size of the entire packet minus HBUFF.	Header size.

Notes:

1. If the data posted to the packet buffer is larger than PKTL, multiple buffers (descriptors) are used.
   a. All buffers but the last one are full (PKTL = DBUFF) while the last one contains the rest of the data.
   b. Only the header buffer of the first descriptor is used.

2. For split conditions, the packet and header lengths are taken from the received packets including all L2 tags (VLAN for example) as well as the CRC bytes. Regardless if these tags are posted to the buffers or posted to the receive descriptors or extracted from the packet. Orthogonal to this condition, the reported HDRL parameter represents the actual data posted to the buffer.

38.30.4.2 Receive L3 and L4 Integrity Check Offload

The 10 GbE controller offloads the following L3 and L4 integrity checks: IPv4 header(s) checksum, outer UDP checksum, inner TCP or UDP checksum and SCTP CRC integrity (see Table 38-430). The 10 GbE controller identifies the packet type and then checks the matched integrity scheme. The identified packet type is reported on the *PTYPE* field in the receive descriptor. Processing indication of the L3 and L4 headers is reported on the L3L4P flag in the receive descriptor. Potential IPv4 checksum error, L4 integrity error and outer IPv4/UDP checksum error are reported by the IPE, L4E and the EIPE error flags in the receive descriptor, respectively.

Note: Outer UDP checksum is supported for all supported encapsulation over UDP (VXLAN, Geneve, and Teredo).

Some rules for integrity check offload are listed as follows. If the following rules are not met, integrity offload is not provided and L3L4P is not set.

- IPv4 header is assumed to be at least 20 bytes long (the length of the basic header).
- IPv4 headers might have any IP option headers that fit within the maximum header size (60 bytes).
- IPv6 support — The pseudo header for the L4 checksum takes into account the addresses in the IPv6 header ignoring the optional extension headers. Packets with routing header type 2 and destination options header with home address option contain an alternative IP address in the extension header. Therefore, checksum calculation for such packets most probably results in an erroneous value. The 10 GbE controller indicates the existence of a destination options header or a routing header in the *IPV6EXADD* bit of the Rx descriptor. Software can then do one of the following:
  - — Ignore the checksum done by the device.
  - — Parse the extension header and identify if it contains an IP address. Then ignore the checksum done by the device only in this case.
  - — Tunneled packets without internal IP header — The *L3L4P* bit is cleared for these packets indicating no checksum offload.
  - — Fragmented packets — The 10 GbE controller parses fragmented receive packets up to including the IP header (for IPv4) or up to including the fragmentation extension header (for IPv6).
  - — L4 checksum offload is not supported for IPv6 fragmented packets and the L3L4P flag in the receive descriptor is not set.
  - — Fragmented IPv4 packet is offloaded up to including the IP header.
  - — TCP header is assumed to be at least 20 bytes long (the length of the basic header).
  - — The TCP header might have any option headers that fit within the maximum header size (60 bytes).
  - — VM-to-VM loopback traffic is processed by the hardware for L3/L4 integrity check as any other packet received from the network.

Table 38-430 lists all supported packet formats and the processed integrity. The table uses the following notations:

- IP is a generic term for IPv4 header or IPv6 header. The IPv4 header can have IP option headers and the IPv6 header can have IPv6 extension headers.

- L4 is a generic term for UDP, TCP or SCTP headers.
- IP checksum is meaningful only for IPv4.
- Checksum is a generic term for UDP and TCP checksum as well as SCTP CRC integrity.
- Zero UDP checksum — Zero UDP checksum for IPv4 packet is treated as no checksum and is reported by hardware as no error. Zero UDP checksum for IPv6 packet is illegal and is reported by hardware as L4 checksum error.

Table 38-430.Integrity Offload Check for Receive Packet Types

Packet Type	Supported Integrity Offload	Reported L3L4P
IP ≥ [data / Unknown / fragmented].	IP checksum offload.	1 (for IPv4) / 0 (for IPv6).
IP ≥ L4.	IP and L4 checksum offload.	1.
IP ≥ IP ≥ [data / Unknown / fragmented].	2 x IP checksum offload.	1 if at least one of the IP headers is IPv4.
IP ≥ IP ≥ L4.	IP and L4 checksum offload.[1]	1.
IP ≥ [tunnel header] ≥ IP ≥ data / Unknown / fragmented.	Only IP checksum offload.	1 if at least one of the IP headers is IPv4.
IP ≥ [tunnel header] ≥ IP ≥ L4.	IP and L4 checksum offload.[2]	1.
IP ≥ [tunnel header] ≥ data and IP ≥ [tunnel header] ≥ MAC ≥ data	None.	0
IP ≥ [tunnel header] ≥ MAC ≥ IP ≥ data.	IP checksum (relevant only for IPv4) and outer UDP checksum.	1 if at least one of the IP headers is IPv4.
IP ≥ [tunnel header] ≥ MAC ≥ IP ≥ L4.	IP, outer UDP and L4 checksum offload[a].	1.

Notes:

1. The L4 checksum offload relates to the inner header:
   a. For UDP or TCP protocols, hardware calculates the expected checksum including the pseudo IP header.
   b. For SCTP protocol, hardware calculates the expected SCTP CRC.
2. Tunneling headers could be one of the following: GRE, Teredo, VXLAN UDP header.

38.31 LAN Transmit Data Path

The LAN transmit data path section covers the following major topics:

- Transmit packets stored in system memory and indicated to hardware by descriptors.
- Transmit descriptor queues that are called also descriptor rings and transmit arbitration queue lists.
- Stateless transmit offloads.

38.31.1 Transmit Packet in System Memory

Transmit packets are made up of data buffers in host memory indicated to hardware by descriptors (16-byte structures described in Section 38.31.2.2). These descriptors include pointer and length pairs to the data buffers as well as control fields for the transmit data processing. In some cases additional control parameters that cannot fit within the data descriptors are needed to process the packet(s). In this case, additional context descriptor(s) are needed in front of the data descriptors. A few examples for needed context descriptor(s) are: transmit segmentation (TSO) and FD filter programming. Refer to an example of a transmit packet in host memory composed of two buffers (header buffer and payload buffer), indicated by two matched data descriptors and optional context descriptor previously provided.

A few rules related to the transmit packet in host memory are:

- The total size of a single packet in host memory must be at least 17 bytes and up to the max frame size of the port as configured by the Set MAC Config admin command.
  - — Packets outside this range are considered malicious. The respective queue is stopped and an interrupt is issued to the PF. The relevant event is a bad single send size”
  - — This rule applies for single packet send as well as any packet within a transmit segmentation (TSO).
- A single transmit packet might span up to eight buffers (up to eight data descriptors per packet including both the header and payload buffers).
- The total number of data descriptors for the entire TSO (explained later on in this section) is unlimited as long as each segment within the TSO obeys the previous rule (up to eight data descriptors per segment for both the TSO header and the segment payload buffers).
- If a packet or TSO spans on multiple transmit data descriptors, the fields in all the data descriptors must be valid.

38.31.2 Transmit Descriptors

This section describes the descriptors provided on the LAN transmit queues. The 10 GbE controller supports the following transmit descriptor types.

LAN descriptors: In many cases only the data descriptor is needed to indicate a packet. Multiple descriptors are needed for FD filter programming, for TSO and/or packets that require offloads like tunneling. In those cases, the descriptor’s order must follow the order listed in Table 38-431.

Table 38-431.LAN Descriptor Types

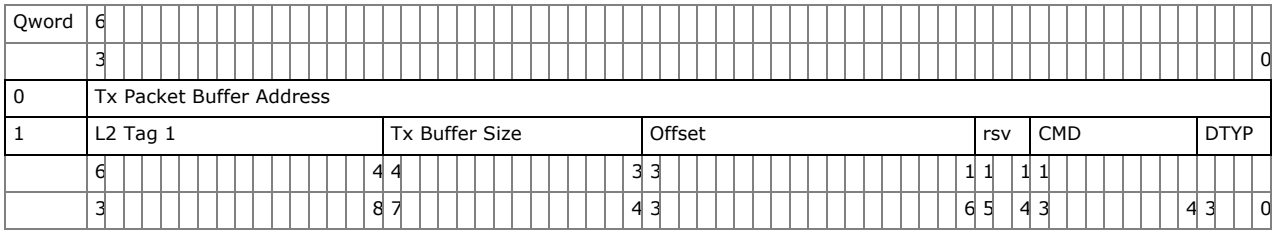
Type	DTYP Value	Description	Reference
Optional NOP Descriptors	0x1	NOP descriptors can be used to align descriptors to cache lines (optional usage)	Section 38.31.2.1.2
LAN Transmit Context Descriptor	0x1	Used as a companion to the transmit data descriptor to provide more information on the packet.	Section 38.31.2.2
FD Filter Programming Descriptor	0x8	Used to program FD filters.	Section 38.31.2.3
Transmit Data Descriptor	0x0	Regular data descriptor used to send LAN packets.	Section 38.31.2.1

The following sections describe these descriptors.

Note: For all descriptors: fields indicated as rsv (reserved) should be set to zero by software at programming time.

38.31.2.1 General Descriptors

38.31.2.1.1 Transmit Data Descriptor



Descriptor type - DTYP (Qword 1, bits 0:3) — 0x0 stands for a transmit data descriptor.

Command field - CMD — Qword 1, bits 4:15.

Bits	Name	Functionality
0	EOP	End Of Packet. The EOP flag is set in the last descriptor of a packet or TSO.
1	RS	Report Status. When set, hardware reports the DMA completion of the transmit descriptor and its data buffer. Completion is reported by setting the *DTYP* field to 0xF or by head write back as configured by the HEAD_WBEN flag in the transmit context. The RS flag can be set only on a last transmit data descriptor of a packet or last transmit data descriptor of a TSO or last transmit data descriptor of a filter.
2	RSV	Reserved
3	IL2TAG1	Insert an L2 tag from the *L2TAG1* field in this descriptor. For MAC in UDP or MAC in GRE tunneling, the *L2TAG1* field is relevant to the outer L2 header. See IL2TAG_IL2H for VLAN tag insertion to the inner L2 header. The type of the tag is defined by the *L2TAG1INSERTID* field in the VSI_L2TAGSTXVALID register of the VSI. The structure of this tag is defined by the matched GL_SWT_L2TAGxxx registers. If the type of L2 tag requires more than two variable bytes, then additional bytes are taken from the *L2TAG2* field while *L2TAG1* is first on the wire. In this case, the IL2TAG2 flag must be cleared.
4	DUMMY	When the dummy flag is set, the packet is not transmitted (internal and external). The dummy indication can be useful for context programming purposes such as FD filters. Note that when using the dummy option, the packet does not have to have a correct checksum. Software should set all the fields in the data descriptor and context descriptors describing the packet structure as it does for normal packets.
6:5	IIPT	The IP header type and its offload. In case of tunneling, the IIPT relates to the inner IP header. Refer to the *EIPT* field for the outer (external) IP header offload. 00b = non IP packet or packet type is not defined by software. 01b = IPv6 packet. 10b = IPv4 packet with no IP checksum offload. 11b = IPv4 packet with IP checksum offload.
7	Reserved	Reserved.
9:8	L4T	L4T is the L4 packet type: 00b = Unknown / fragmented packet. 01b = TCP. 10b = SCTP. 11b = UDP. When L4T is set to other values than 00b, L4LEN must be defined as well. When set to UDP or TCP, hardware inserts the L4 checksum and when set to SCTP hardware inserts the L4 CRC.

Header offset parameters — OFFSET (Qword 1, bits 16:33).

Bits	Name	Functionality
6:0	MACLEN	MAC header length defined in words. In case of a tunnel packet, MACLEN defines the outer L2 header. MACLEN defines the L2 header length up to including the Ether-type. If L2 tag(s) are provided in the data buffers, then they are included in MACLEN. Any MPLS header should be included in MACLEN.
13:7	IPLEN	IP header length (including IP optional/extended headers) in the Tx buffer defined in Dwords. In case of a tunnel packet, IPLEN defines the most inner IP header length.
17:14	L4LEN	L4LEN is the L4 header length in the Tx buffer defined in Dwords. In case of a tunnel packet, L4LEN defines the most inner L4 header length. L4LEN should obey the following rule: When the *L4T* field is set to 00b, L4LEN must be set to zero. Otherwise, it should be set to 8 / 12 for UDP / SCTP, respectively and should be equal or larger than 20 for TCP.

L2 Tag 1 - L2TAG1 - (Qword 1, 48:63) — A 16-bit tag to be inserted into the packet if the IL2TAG1 flag is set. If IL2TAG1 is cleared, L2TAG1 should be set by software to zero.

Transmit buffer description — BUFF (Qword 0, bits 0:63; Qword 1, bits 34:47).

Table 38-432.Transmit Buffer

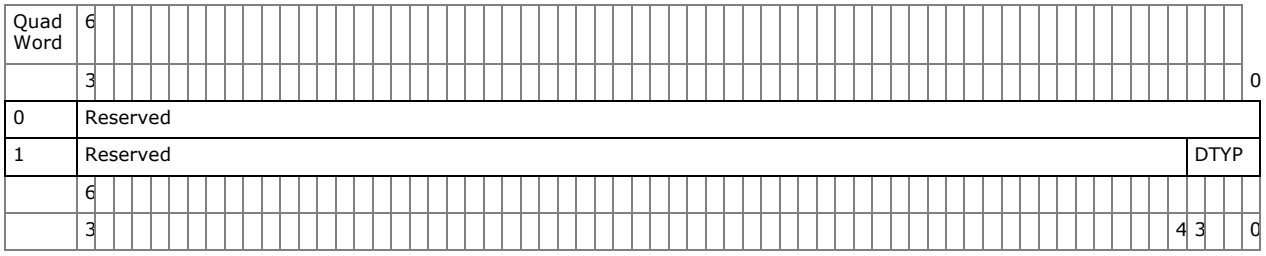
Bits	Name	Functionality
47:34	BSIZE	Buffer size in byte units from 1 byte up to 16 KB minus 1.
63:0	BADDR	Buffer address in byte granularity.

38.31.2.1.2 NOP Descriptor

A NOP descriptor can be used if there is a software need to align descriptors. An NOP descriptor does not cause any activity other then processing the descriptor.

NOP descriptors are is implemented by a null setting of a context descriptor as follows: DTYP should be set to 0x1 (LAN context descriptor) and all other fields should be cleared. Note that NOP descriptors are permitted only between commands.

38.31.2.1.3 Transmit Descriptors Write-Back Format



Descriptor Type - DTYP — Qword 1, bits 0:3.

- Hardware indicates a completed descriptor by setting the *DTYP* field to a value of 0xF. The write-back status is the same for all transmit descriptors used for LAN traffic and filter programming.
- Hardware reports this status in the following two cases:
  - — For descriptors with the *RS* bit set (Report Status) while the HEAD_WBEN flag in the transmit context is set to descriptor write-back mode. Refer to the rules for setting the *RS* flag in the transmit descriptors in Section 38.31.2.1.1.
  - — For the last descriptor of a command that was executed before an interrupt of the queue is initiated.

38.31.2.2 LAN Transmit Context Descriptors

38.31.2.2.1 Transmit Context Descriptor

A context descriptor might have additional setting options for a single packet or TSO defined by the data descriptor(s) that follows. Following the previously described packet or TSO, the context provided by this descriptor is lost.

See Table 38.31.4 for field settings as a function of required offload.

Quad Word	6 3	4 4 8 7	3 3 2 1	2 2 4 3	0
0	rsv	L2TAG2 (STag / VEXT)	rsv	Tunneling Parameters	
1	MSS / TARGET_VSI	rsv	TSO Total Length	rsv	rsv	CMD	DTYP

	6 3	5 4 0 9	4 4 8 7	3 2 0 9	1 1 8 7	1 1 1 0	4 3	0

Descriptor Type - DTYP (Qword 1, bits 0:3) — 0x1 stands for a LAN context descriptor.

Command Field - CMD — Qword 1, bits 4:10.

Bits	Name	Functionality
0	TSO	Transmit segmentation offload is activated when the TSO flag is set.
1	TSYN	1588 Time Stamp. When set, hardware samples the packet transmission time in the PRTTSYN_TXTIME register. The TSYN flag is processed by the device only if TimeSync is enabled by the TSYNENA flag in the transmit queue context. Note that the TSYN flag is aimed only for a single send with no TSO.
2	IL2TAG2	Insert the L2 tag from the *L2TAG2* field in this descriptor. For MAC in UDP or MAC in GRE tunneling, the *L2TAG2* field is relevant to the outer L2 header. See IL2TAG_IL2H for VLAN tag insertion to the inner L2 header. The type of the tag is defined by the *L2TAG2INSERTID* field in the VSI_L2TAGSTXVALID register of the VSI. The structure of this tag is defined by the matched GL_SWT_L2TAGxxx registers. See also IL2TAG1 in the data descriptor for 32-bit tag handling.
3	IL2TAG_IL2H	Insert VLAN to the inner L2 header from the *L2TAG2* field in this descriptor. The VLAN is added after the header defined by L4TUNLEN. *IL2TAG2* and *IL2TAG_IL2H* are mutually exclusive and software might set only one of them. *IL2TAG_IL2H* can be set to 1b only for L4 tunneled packets with non-zero L4TUNLEN.
5:4	SWTCH	Switch Control Tag (Can be set to non-zero only by control VSI as programmed by the *Allow Destination Override* flag per VSI): 00b = No switch control tag. The packet is routed according to hardware filters. 01b = Uplink packet. The packet is transmitted to the network bypassing hardware filters. 10b = Local packet. The packet is transmitted only to local VSIs according to the hardware filters. 11b = Target VSI. The packet is transmitted to a specific VSI defined in the *MSS/VSI* field in this descriptor. TSO is mutually exclusive with this option.
6	Reserved	Reserved.

Segmentation parameters / switching parameter — Qword 1, bits 30:47; bits 50:63.

Bits	Name	Functionality
47:30	TLEN	TSO Total Length. This field defines the L4 payload bytes that should be segmented. Note that the sum of all transmit buffer sizes of the TSO should match exactly the TLEN plus the TSO header size in host memory. If the TSO flag is cleared, the TLEN should be set by software to zero. If the TSO flag is set, the TLEN must be set by software to a larger value than zero.

Bits	Name	Functionality
63:50	MSS / TARGET_VSI	When the TSO flag is set, this field functions as MSS. When the *SWTCH* field is set to 11b, this field function as TARGET_VSI. If the TSO flag is set then the *SWTCH* field should not be set to 11b and vice versa. If both the TSO flag is cleared and the *SWTCH* field is not equal to 11b then this field should be set to zero. When the TSO flag is set, the *MSS* field defines the MSS of the packet's payload in the TSO (excluding the L2, L3 and L4 headers). In case of tunneling, the MSS relates to the inner payload. In this case, the MSS should not be set to a lower value than 64 or larger than 9668 bytes. When the *SWTCH* field equals to 11b, it is the destination VSI of the packet. This option is valid only for control VSIs on which the *Allow Destination Override* flag is set.

Tunneling parameters — Qword 0, bits 0:23.

Bits	Name	Functionality
1:0	EIPT	The external (outer) IP header type and its offload: 00b = No external IP header. 01b = External IPv6. 10b = External IPv4 with no checksum offload. 11b = External IPv4 with checksum offload.
8:2	EIPLEN	External (outer) IP header length (including IP optional/extended headers) defined in Dwords. When the packet has no outer IP header (EIPT equals to zero), this field must be set to zero.
10:9	L4TUNT	L4 tunneling type (Teredo / GRE header / VXLAN header) indication: 00b = No UDP / GRE tunneling (field must be set to zero if EIPT equals to zero). 01b = UDP tunneling header (Any UDP tunneling, VXLAN and Geneve). 10b = GRE tunneling header. Else = Reserved.
11	RSV	Reserved, set to zero.
18:12	L4TUNLEN	L4 tunneling length (Teredo / GRE header / VXLAN header) defined in words (field must be set to zero if L4TUNT equals to zero). For standard Teredo headers with no additional header payload it should be set to four, which equals eight bytes. If the tunneling header includes proprietary content it should be included as well. For IP in GRE it should be set to the length of the GRE header. For MAC in GRE or MAC in UDP it should be set to the length of the GRE or UDP headers plus the inner MAC up to including its last Ether-type. Note that this field represents the length of data provided in host memory buffers. If the L4TUNT is cleared, this field must be set to zero.
22:19	DECTTL	Decrement TTL in the inner IP header by DECTTL and drop the packet if the original TTL was not greater than DECTTL. If the EIPT is cleared, this field must be set to zero.
23	L4T_CS	Calculate the outer L4 checksum. Must be set only if L4TUNT = 01b and EIPT is not zero. Supported only if L4 protocol is UDP.

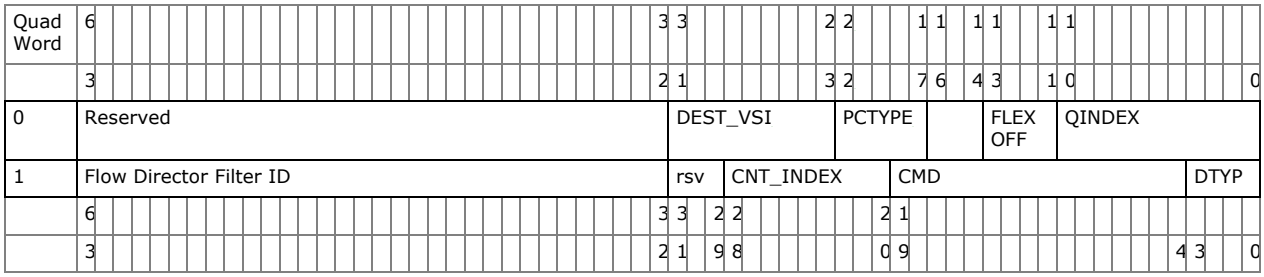
L2 tag 2 — Qword 0, bits 32:47.

Bits	Name	Functionality
47:32	L2TAG2	A 16-bit tag to be inserted to the packet if the IL2TAG2 flag is set or if IL2TAG1 is set while L2 tag 1 include four variable bytes or if the IL2TAG_IL2H flag is set. If the previous conditions are not met, L2TAG2 should be set by software to zero.

Note: RSV, reserved bits that must be set to zero.

38.31.2.3 Filter Programming Descriptor

The filter programming descriptor should be followed by data descriptor(s).



Qword 0.

Bits	Name	Functionality
10:0	QINDEX	Destination queue index for this filter. The queue index is defined in the range of the VSI.
13:11	FLEXOFF	Flexible Offset. This is a word offset in the flexible payload" in the field vector relative to the beginning of the flexible payload space from which bytes are extracted to the 32-byte receive descriptor. The FD_STATUS flag enables either four or eight bytes or none. The (first) four bytes indicated by FLEXOFF are posted to the *Flexible Bytes Low* field in the receive descriptor and the (last) four bytes are posted (if enabled) to the *Flexible Bytes High* field. Note that the FLEX bytes are supported only by 32-byte receive descriptors. FLEXOFF must not be larger than six when four bytes are enabled and must not be larger than four when eight bytes are enabled.
16:14	RSV	Reserved.
22:17	PCTYPE	Matched Packet Type. When the FD_AUTO_PCTYPE flag in the GLQF_CTL register is cleared, this field defines the PCTYPE of the programmed FD filter. When the FD_AUTO_PCTYPE flag in the GLQF_CTL register is set, this field is meaningless and should be set to zero. In this case, the device concludes the PCTYPE by the packet used to program the FD filter.
31:23	DEST_VSI	Expected VSI Index. 0x0 = 0x17F. Matched VSI index of the received packet. Else = Reserved Note that hardware does not check DEST_VSI at programming time while it is software's responsibility to program DEST_VSI, which is owned by the PF.
63:22	RSV	Reserved.

Qword 1.

Bits	Name	Functionality
3:0	DTYPE	0x8 for a FD filter programming descriptor.
19:4	CMD	Described in the table that follows.
28:20	CNT_INDEX	Statistic counter index for packets that match this filter. This field is meaningful only when the CNT_ENA flag is set in this descriptor.
31:29	RSV	Reserved.
63:32	FDID	Flow Director Filter ID. Note that FDID is opaque to hardware. The programmed FDID can be reported back in the received descriptor of a matched received packet (if enabled by the FD_STATUS parameter in this descriptor). If so, it can be used by software to associate the received packet with its matched filter.

Command field - CMD — Qword 1, bits [4:19].

Bits	Name	Functionality
2:0	PCMD	Filter Programming Command: 001b = Add filter[1]. 010b = Remove filter. Else = Reserved.
4:3	DEST	Matched Packet Destination Control. 00b = Drop packet. 01b = Direct packet to LAN queue defined by the QINDEX. 10b = Direct packet to LAN while the queue is defined by other filters. 11b = Reserved.
7	CNT_ENA	Statistic Counter Enable Flag (relevant only for FD filter). This flag enables packet counting by the statistic counter defined by the statistic counter index in this command.
8	RSV	Reserved.
10:9	FD_STATUS	The FD_STATUS flag enables status indication of the FD filter ID and extracting flexible bytes in the receive descriptor as follows: 00b = FD ID and flexible bytes are not enabled. 01b = FD ID is enabled. 10b = FD ID and 4 Flexible bytes are enabled. 11b = Eight flexible bytes are enabled.
13:11	RSV	Reserved.
14	ATR	If set, this filter is candidate for automatic eviction.
15	RSV	Reserved.

Notes:

1. If the filter entry does not exist, the filter is added. If the filter already exists, the filter parameters are updated.

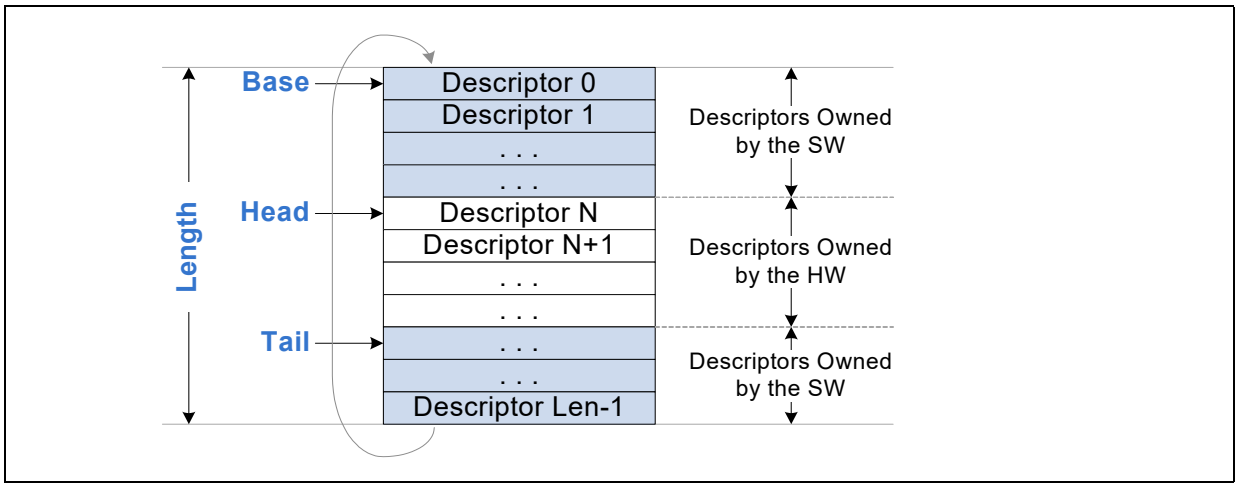
38.31.3 LAN Transmit Queue (Ring)

Software prepares structures for transmission in system memory indicated to hardware by a list of consecutive descriptors. These descriptors are organized in a contiguous memory handled as a cyclic queue, which is also called a descriptor ring. The 10 GbE controller supports up to 1536 transmit queues allocated to PFs and VFs as described in Section 38.30.2.

Transmit queues are defined by a set of parameters called the queue context. The main parameters are the queue pointers shown in the Figure 38-99. The queue context includes additional parameters that define the queue functionality as detailed in Section 38.31.3.4. Part of these context parameters are kept in hardware (like the Tail register, queue enable flag and interrupt related context). Most of the queue context parameters are stored in FPM, fetched to an internal cache when required.

the software interface to the queue (for initialization, normal operation, and queue disable flow) is described in Section 38.31.3.1. The 10 GbE controller has additional global parameters for the entire device or per function parameters that affect multiple queues. See Section 38.30.3.3.1 for more detail.

Figure 38-99.Transmit Descriptor Ring Structure



38.31.3.1 Transmit Queue Programming

Queue enable and disable flows are described in the following section and listed in Table 38-433.

Table 38-433.Transmit Queue Enable / Disable Flags

QTX_ENA[n] Register		Transmit Queue State
QENA_REQ	QENA_STAT	
0	0	Queue is not enabled.
1	0	Queue enable request by software.
1	1	Queue is enabled.
0	1	Queue disable request by software.

38.31.3.1.1 Transmit Queue Enable Flow

Queue enable flow is executed by PF software for PF queues as well as for the queues of its VFs.

- The function that owns the queue (PF or VF) allocates contiguous memory in its own memory space for the transmit ring.
- If the queue might have been disabled recently then software should wait at least 50 ms from the completion of the queue disable flow before proceeding to the next step.
- Software should clear the queue disable flag by setting the CLEAR_QDIS flag in the matched GLLAN_TXPRE_QDIS[n] register. The QINDX field in the GLLAN_TXPRE_QDIS[n] register should be set to the absolute queue index (equals to the queue index within the PF plus the FIRSTQ field in the PFLAN_QALLOC register of the PF).
- Software prepares the queue context in the FPM in PF memory space, clears the QTX_HEAD[n] register and then sets the QENA_REQ flag in the QTX_ENA[n] register, while 'n' is the queue index within PF space. Before that, the PF also defines the function that owns the queue in the PFVF_FUN; PF_INDX; VFVM_INDX parameters in the matched QTX_CTL[n] register. See several examples for queue context settings in Section 38.31.3.4.3. At pre-boot time, memory allocation might be tight and the FPM could exceed the budget.

Note: There is a mechanism to program the queue context directly to the device bypassing the FPM as described in Section , "Direct Queue Context Programming".

- As a response, hardware sets the QENA_STAT flag in the QTX_ENA[n] register. The QENA_STAT follows the QENA_REQ almost instantly and not more than 10 µs after that.
- Once the QENA_STAT flag in the QTX_ENA[n] register is set, software can start using the queue.
- If the queue is targeted for a VF, PF software should also program the matched entry in the VFQ_TABLE. Then it is expected to signal VF software that the queue is enabled. Note that the VFQ_TABLE is shared for the receive and transmit queues. So PF software should also enable the receive queue before communicating it to the VF.

38.31.3.1.2 Transmit Queue Disable Flow

Queue disable flow is executed by PF software for PF queues as well as for the queues of its VFs.

- Remove the queue from the interrupt linked list.
- PF software sets the SET_QDIS flag in the matched GLLAN_TXPRE_QDIS[n] register. The DINDX field in the GLLAN_TXPRE_QDIS[n] register should be set to the absolute queue index (equals to the queue index within the PF plus the *FIRSTQ* field in the PFLAN_QALLOC register of the PF). Software should guarantee a gap of at least 100 µs at 2 x 10 GbE and above or 400 µs at 1 x 10 GbE before the next step executes. Note that software could execute this step for multiple transmit queues and then proceed to the next step for these queues concurrently.
- PF software clears the QENA_REQ flag in the QTX_ENA[n] register, while 'n' is the queue index within the PF space.
- Hardware schedules this event like it does for any other software update of the tail. Further updates of the tail register are ignored by hardware.
- Hardware posts a queue disable marker in the transmit pipe.
- Any preceding commands in the transmit pipe are processed while their buffers are fetched from host memory.
- Eventually, the queue disable marker gets to the end of the pipe. At this point it is guaranteed that the pipe does not contain any additional transmit commands that might require the host memory.
- Hardware invalidates the transmit queue from the internal cache and also clears the QENA_STAT flag in the QTX_ENA[n] register.
- Once the QENA_STAT flag in the QTX_ENA[n] register is cleared, software releases all memory structures of the queue. If the QENA_STAT flag in the QTX_ENA[n] register is not cleared within a reasonable time it might be an indication for too long pause packets from the network.

Note: The progress of the queue disable marker following clearing the QENA_REQ flag is subjected to scheduling in the transmit data path the same as any other packet. This means that processing latency might take longer time if the TC is paused.

38.31.3.1.3 Fast Transmit Queue Disable Flow

Fast queue disable flow is executed by hardware following a PFR, VFR or VMR resets. Once these resets are completed by hardware, all transmit queues of the function or the VM are disabled. All queue contexts of the matched function or the VM are invalidated from the internal cache and their QENA_STAT flag in the matched QTX_ENA[n] register are cleared while the queue context is not updated in the FPM.

38.31.3.1.4 Software Fast Path Programming

During normal operation, the function that owns the queue (PF or VF) accesses the hardware directly.

- Prepare transmit descriptors starting at the descriptor indicated by the TQTAIL pointer in the relevant QTX_TAIL register. Note that software should update the TQTAIL at whole-structured boundaries. For TSO, it is the entire TSO, for a single packet, it is the entire packet and for filter, it is the end of the packet that is associated to the filter programming. Failing to do so might hang the activity of the queue.
- Software should never set the TQTAIL to a value above the descriptors owned by hardware minus one. Descriptors considered as owned by hardware are those already indicated to hardware but are not yet reported as completed.
- Hardware reports completed descriptors only for those ones indicated by an *RS* bit in the CMD field in the descriptor. Completion indication is provided in one of two modes as programmed by the HEAD_WBEN parameter in the queue context as follows:
  - Descriptor write back — Hardware changes the value of the *DTYP* field in the completed descriptor to a 0xF (*DTYP* equals to 0xF is reserved for completed descriptor indication). It also sets the rest of the descriptors to all zero's.
  - Head write back — Hardware indicates the head pointer of the next descriptor that follows a completed descriptor (with active *RS* bit). Hardware post the four bytes of the head pointer at the address defined by the HEAD_WBADDR parameter in the queue context.
- Bump the tail to the last prepared descriptor plus one (the descriptor index to be added in the next time).

38.31.3.2 Transmit During Link Down

In most usage models when the link becomes inactive, it is not required to preserve the transmit packets that were not sent. During link down, hardware continues to fetch transmit descriptors and data regardless of the link status. Packets directed to the network are discarded while packets directed to local VSI port are forwarded successfully.

Note that when the link becomes inactive, PFs on this port get a link status change interrupt. It is expected that the VFs get the link status change indication from their parent PF.

38.31.3.3 Transmit Arbitration Queue Set

Transmit queues are grouped into arbitration queue sets. The 10 GbE controller supports 1024 LAN queue sets (indicated by the RDYList parameter in the transmit queue context). Queue arbitration within the same queue sets is a simple round robin scheme. A queue context is fetched on demand (if not already in the cache) when it is scheduled. Then hardware fetches the transmit descriptors (if not already in the cache) and then it fetches the transmit data.

38.31.3.4 Transmit Queue Context Parameters

This section describes the setting options of the LAN transmit queue parameters named as queue context. Part of the queue context parameters reside in dedicated hardware registers and part of the context is stored in host memory (FPM) while the active queues are kept in cache.

38.31.3.4.1 Transmit Queue Context in Hardware Registers

The queue context parameters that software accesses during fast past programming as well as its enablement reside in hardware registers:

- Queue enablement flags in the QTX_ENA[n] registers while 'n' is the queue index of the PF. Queue enable and disable flow by the PF are explained in the sub-sections of Section 38.31.3.1.
- The tail pointer in the QTX_TAIL[n] registers while 'n' is the relative queue index of the function (PF or VF). The usage of the Tail register is detailed in Section 38.31.3.1.4.
- Association of the queue to a function (PF and its VF or VM) is programmed by the QTX_CTL[n] registers while 'n' is the relative queue index of the PF. The QTX_CTL includes the following parameters that are programmed by the PF:
  - PF_INDX is the absolute index of the PF (between 0 and 15).
  - PFVF_Q flag associate the queue to the PF, the EMP or to one of its VFs or its VMs.
  - VFVM_INDX is the absolute index of the VF (between 0 and 127) if the queue belongs to a VF or the absolute index of the VSI (between 0 and 383) if the queue belongs to a VM. Otherwise, the VFVM_INDX should be set to zero.

38.31.3.4.2 Transmit Queue Context in FPM

The transmit queue context parameters that are stored in host memory (FPM) are fetched for the active queues to the internal cache. These parameters are listed in Table 38-434. The queue context is structured as contiguous lines of 128 bits (16 bytes) each, as follows:

Table 38-434.LAN Tx Queue Context in the Private Host Memory (Sheet 1 of 3)

Alias	Width [Bits]	LS Bit	MS Bit	Type	SW Init	Description
Line 0						
HEAD	13	0	12	Dynamic	0x0	Transmit Queue Head. An index updated by hardware relative to the beginning of the queue that defines the next descriptor to be used. All descriptors starting by the head up to (excluding) the TTAIL are owned by hardware and the rest are owned by software. At idle time the head equals TTAIL.
RSV	16	13	29	Dynamic	0x0	Reserved.
New_Context	1	30	30	Dynamic	0x1	New Context Flag. Should be set to 1b by software at queue context programming.
RSV	1	31	31	N/A	0x0	Reserved.
BASE	57	32	88	Static	BASE	Transmit Queue Base Address. Indicates the starting address of the descriptor queue defined in 128-byte units.
Reserved	1	89	89	Static	Reserved	Reserved.

Table 38-434.LAN Tx Queue Context in the Private Host Memory (Sheet 2 of 3)

Alias	Width [Bits]	LS Bit	MS Bit	Type	SW Init	Description
TSYNENA	1	90	90	Static	TSYNENA	TimeSync Enable. If the TSYNENA flag is set, then setting the TSYN flag in the transmit context descriptor is processed by the hardware sampling the packet timestamp in the PRTTSYN_TXTIME register.
FDENA	1	91	91	Static	FDENA	Enable FD filter programming by this queue.
ALT_VLAN	1	92	92	Static	ALT_VLAN	Alternate VLAN Tag Selects. At 0b the PORT_TAG_ID tag in the VSI_TIR is disabled and at 1b the PORT_TAG_ID tag in the VSI_TAIR register is enabled.
RSV	3	93	95	N/A	0x0	Reserved.
CPUID	8	96	103	Dynamic	CPUID	CPU socket ID for TPH. The CPU socket ID is updated by hardware.
RSV	4	104	127	N/A	0x0	Reserved.
Line 1						
THEAD_WB	13	0	12	Dynamic	0x0	Reflection of the reported head at head write back.
RSV	19	13	31	N/A	0x0	Reserved.
HEAD_WBEN	1	32	32	Static	HEAD_WBEN	This flag selects between head write back and transmit descriptor write back: 0b = Descriptor write back 1b = Head write back
QLEN	13	33	45	Static	QLEN	Transmit Queue Length (QLEN). Defines the size of the descriptor queue in descriptors units from 8 descriptors (QLEN=0x8) up to 8 KB-32 descriptors (QLEN=0x1FE0). QLEN restrictions: At smaller queue size than 32 descriptors the QLEN must be a whole number of 8 descriptors. At a larger size than 32 descriptors, QLEN must be a whole number of 32 descriptors.
TPHRDesc	1	46	46	Static	TPHRDesc	Descriptor Read TPH Ena flag for descriptors fetch.
TPHRPacket	1	47	47	Static	TPHRPacket	Packet Content TPH Ena flag for the packet header and payload.
TPHWDesc	1	48	48	Static	TPHWDesc	Head WB / Descriptor Completion Status Write Back TPH Ena flag.
RSV	15	49	63	N/A	0x0	Reserved
HEAD_WBADDR	64	64	127	Static	HEAD_WBADDR	When HEAD_WBEN is set to 1b, this field defines the HEAD write-back address. Must be set to 0x0 if HEAD_WBEN is cleared.
Line 2						
RSV	128	0	127	N/A	0x0	Reserved.
Line 3						
RSV	128	0	127	Tx_Desc	0x0	Reserved for internal parameters.
Line 4						
RSV	128	0	127	Internal	0x0	Reserved for internal parameters.
Line 5						
RSV	128	0	127	Internal	0x0	Reserved for internal parameters.
Line 6						
RSV	128	0	127	Internal	0x0	Reserved for internal parameters.

Table 38-434.LAN Tx Queue Context in the Private Host Memory (Sheet 3 of 3)

Alias	Width [Bits]	LS Bit	MS Bit	Type	SW Init	Description
Line 7						
RSV	84	0	83	Internal	0x0	Reserved.
RDYList	10	84	93	Static	RDYList	Transmit arbitration queue set. The RDYList index is absolute so it should be set to those RDYList allocated to the function.
RSV	34	94	127	Internal	0x0	Reserved.

38.31.3.4.3 Examples for Transmit Queue Context Setting

Following is an example of a transmit queue context setting:

PF LAN transmit queue context example =

Offset	Even_L.0	Even_L.1	Even_L.2	Even_L.3	Odd_L.0	Odd_L.1	Odd_L.2	Odd_L.3
Line 0, 1:	0x40000000	001579A0	08000000	00000000	00000000	0001C200	00000000	00000000
Line 2, 3:	0x00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
Line 4, 5:	0x00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
Line 6, 7:	0x00000000	00000000	00000000	00000000	FFFFFFFF	480FFFFF	00000000	00000000

BASE = 0x001579A0 (0x0ABCD000)	HEAD_WBEN = 0
FCENA = 0	QLEN = 512 (0x200)
TSYNENA = 0	TPH = 111b (enable all TPHxxx flags)
FDENA = 1	HEAD_WBADDR = 0x00...0
ALT_VLAN = 0	RDYList = 0x80 (128)

38.31.4 Stateless Transmit Offloads

38.31.4.1 Insert Ethernet CRC Bytes

38.31.4.2 Transmit L3 and L4 Integrity Offload

The 10 GbE controller can offload the following L3 and L4 integrity checks: IPv4 header(s) checksum for simple and tunneled packets, inner TCP or UDP checksum, tunneling UDP checksum, and SCTP CRC integrity. If a checksum is required, software should provide it as well as the inner checksum value(s) that are required for the outer checksum. In order to request L3 and L4 integrity offloads, software should define the packet format and the required offload in the transmit descriptors as shown in Figure 38-100 and Table 38-436.

Some rules for integrity offload are:

- Offloads for inner headers to an IPv6 header with a fragment extension header or fragmented IPv4 packets do not make sense. Note that this rule is not enforced by the 10 GbE controller.
- IPv6 support — The pseudo header for the L4 checksum takes into account the addresses in the IPv6 header ignoring the optional extension headers. Packets with

routing header type 2 and destination options header with home address option contain an alternative IP address in the extension header. Therefore the operating system should not request a checksum nor segmentation offload for packets with a routing extension header type 2.

- IP header length requirements (IPLEN and EIPLEN).
  - In the case of a non-tunneled packet, IPLEN defines the IP header length in Dword units and the IIPT defines the IP header type.
  - In the case of tunneled packets, IPLEN defines the inner IP header length and EIPLEN defines the outer (external) IP header length. IIPT defines the inner IP header type and EIPT defines the outer IP header type.
  - The *Length* field in the IP header is prepared by the software in the packet buffers. This length field includes the payload of the IP header. In MAC tunneling, the inner MAC header (including optional VLAN tag) is part of the outer IP header payload. This optional VLAN tag can be embedded in the packet buffers or by using the IL2TAG_IL2H option in the transmit context descriptor. Regardless of the previous, the optional VLAN tag should be taken into account in the length field of the outer IP header.
  - For IPv4 it should be at least 20 bytes (basic header size), and not more than 60 bytes.
  - For IPv6 it should be at least 40 bytes (basic header size), and up to the maximum size enabled by the parsing fields for basic IPv6 header and its extension headers.
- L4 header length requirements (L4LEN):
  - For TCP, it should be at least 20 bytes (basic header size), and not more than 60 bytes.
  - For SCTP, it should be set to 12 (SCTP common header size).
  - For UDP, it should be set to 8 (UDP header size).

Table 38-436 lists all supported packet formats and the processed integrity. The table uses the following notations:

- IP is a generic term for an IPv4 header or IPv6 header. The IPv4 header can have IP option headers and the IPv6 header can have IPv6 extension headers.
- L4 is a generic term for UDP, TCP or SCTP headers.
- IP checksum is meaningful only for IPv4.
- Checksum is a generic term for UDP and TCP checksum as well as SCTP CRC integrity.

Offload details:

- IPv4 checksum calculation (both inner and outer IP header for tunneled packets).
  - Software should set the IPv4 checksum to zero.
  - Hardware calculates the IPv4 header checksum starting at the beginning of the IPv4 header up to the end of the header.
- Outer UDP checksum calculation (UDP tunneling).
  - Software provides the pseudo IP header checksum in the outer UDP header.
  - Hardware calculates the UDP checksum starting at the beginning of the tunnel header up to the end of the packet, which includes the pseudo header provided by software.
- UDP and TCP checksum calculation (inner L4 header in case of UDP / GRE tunneling).
  - Software provides the pseudo IP header checksum in the L4 header.

  - — Hardware calculates the L4 checksum starting at the beginning of the L4 offset up to the end of the packet, which includes the pseudo header provided by software.
- See Table 38-435 for a list of fields that the software device driver should fill when sending a packet.
- SCTP CRC calculation (inner L4 header in case of UDP / GRE tunneling).
  - — Software should set the CRC in the header to zero.
  - — Hardware calculates the CRC according to SCTP standard starting at the beginning of the SCTP header up to the end of the packet.
- Tunneling UDP / GRE header length requirements.
  - — For UDP / GRE tunneling, the header can be any value in 2-byte granularity from 8 bytes (bare Teredo UDP header) and up to 254 bytes (including optional network key and optional inner L2 header up to including the last Ether-type). Note that the header length includes only those bytes provided in the packet buffers in host memory.

Table 38-435.Pseudo Header Pre-Loaded Values

Field	Single Send Packet (No TSO)	Transmit Segmentation Offload (TSO)
Outer IP Length	IPv4: IP header +payload size IPv6: payload size only, including header extensions Should match the data size stored in host memory (hardware modifies it upon offloads).	Don't care (calculated by hardware).
Outer L4 (UDP) Length	UDP header size + payload size. Should match the data size that stored in host memories (hardware modifies it upon offloads).	Don't care (calculated by hardware).
Outer L4 CS Field	IP header pseudo checksum, calculated on IP header fields: IPv4: Source addr +destination addr +protocol (=0x11) + UDP length. IPv6: Source addr +destination addr + NextHeader(=0x11) + UDP length. UDP Length is the payload that follows the IP header. It includes the size of UDP header and data.	IP header pseudo checksum, calculated on IP header fields. Not including UDP length: IPv4: Source addr +destination addr + protocol (=0x11). IPv6: Source addr +destination addr + NextHeader(=0x11).
Inner/Single IP Length	IPv4: IP header +payload size. IPv6: payload size only, including header extensions. Should match the data size stored in host memory (hardware modifies it upon offloads).	Don't care (calculated by hardware).
Inner/Single L4 (UDP/ TCP) Length	Header size + payload size. Should match the data size that stored in host memories (hardware modifies it upon offloads).	Don't care (calculated by hardware).
Inner/Single L4 CS Field	IP header pseudo checksum, calculated on IP header fields: IPv4: Source addr +destination addr + protocol + L4 Length. IPv6: Source addr +destination addr + NextHeader + L4 length. Protocol or NextHeader values are: 0x11 for UDP and 0x6 for TCP. L4 Length is the payload that follows the IP header. It includes the size of L4 header and data.	IP header pseudo checksum, calculated on IP header fields. Not including L4 length: IPv4: Source addr +destination addr + protocol. IPv6: Source addr +destination addr + NextHeader. Protocol or NextHeader values are: 0x11 for UDP and 0x6 for TCP.

Figure 38-100.Transmit L3 and L4 Header Lengths (in Host Memory) for Integrity Offload

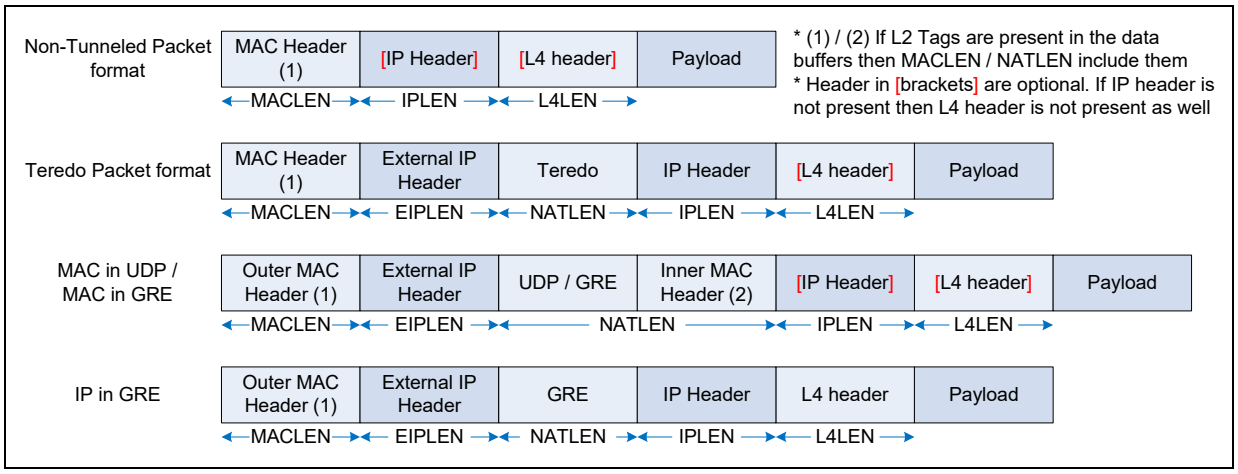


Table 38-436.Transmit Integrity Offload for Packet Types

Packet Type	Parsing Hints and Offload Enablement In The Transmit Descriptors[1,2]	Supported Transmit Checksum Offload
Fragmented IPv4 or IPv4 ≥ Unknown.	IIPT = 10b or 11b. L4T = 00b (unknown).	IPv4 checksum if IIPT = 11b.
Fragmented IPv6 or IPv6 ≥ Unknown.	IIPT = 01b. L4T = 00b (unknown).	None.
IP ≥ L4.	IIPT = 10b or 11b for IPv4. IIPT = 01b for IPv6. IPLEN = the length of the IP header (including IP optional or extension headers). L4T = 11b, 01b, 10b (UDP, TCP, SCTP). L4LEN = L4 header length. EIPT = EIPLEN = L4TUNT = 0 (no tunneling).	IPv4 checksum if IIPT = 11b. L4 checksum if L4LEN is meaningful (like, L4T <> 0).
IP ≥ (Fragmented IP or IP ≥ Unknown).	IIPT, IPLEN, EIPT, EIPLEN, L4TUNT = same as IP ≥ IP ≥ L4. L4T = 00b (unknown).	Inner IPv4 checksum if IIPT = 11b. Outer IPv4 checksum if EIPT = 11b. No L4 checksum.
IP ≥ IP ≥ L4.	IIPT = 10b or 11b for inner IPv4. IIPT = 01 for inner IPv6. IPLEN = the length of the inner IP header (including IP optional or extension headers). L4T = 11b, 01b, 10b (UDP, TCP, SCTP). EIPT = 10b or 11b for outer IPv4. EIPT = 01 for outer IPv6. EIPLEN = the length of the outer IP header (including IP optional or extension headers). L4TUNT = 00b (no L4 tunneling).	Inner IPv4 checksum if IIPT = 11b. Outer IPv4 checksum if EIPT = 11b. L4 checksum if L4LEN is meaningful (like, L4T <> 0).
IP ≥ Teredo UDP / GRE ≥ IP ≥ L4.	IIPT = 10b or 11b for inner IPv4. IIPT = 01 for inner IPv6. IPLEN = the length of the inner IP header (including IP optional or extension headers). L4T = 11b, 01b, 10b (UDP, TCP, SCTP). EIPT = 10b or 11b for outer IPv4. EIPT = 01 for outer IPv6. EIPLEN = the length of the outer IP header (including IP optional or extension headers). L4TUNT = 01b for UDP/GRE tunneling. L4TUNLEN = Tunneling header length.	Same as IP ≥ IP ≥ L4.

Table 38-436.Transmit Integrity Offload for Packet Types

Packet Type	Parsing Hints and Offload Enablement In The Transmit Descriptors[1,2]	Supported Transmit Checksum Offload
IP ≥ Teredo UDP / GRE ≥ (fragmented IP or IP ≥ unknown).	IIPT, IPLEN, EIPT, EIPLEN, L4TUNT, L4TUNLEN = same as IP ≥ IP ≥ L4. L4T = 00b (unknown).	Same as IP ≥ (fragmented IP or IP ≥ unknown).
IP ≥ GRE / UDP ≥ MAC (with/ without VLAN) ≥ IP ≥ L4.	IIPT, IPLEN, L4T, EIPT, EIPLEN = same as IP ≥ IP ≥ L4. L4TUNT = 01b for UDP/GRE tunneling. L4TUNLEN = UDP/GRE header length in the packet buffers up to excluding the inner IP header.	Same as IP ≥ IP ≥ L4 plus tunneling UDP checksum.
IP ≥ GRE / UDP ≥ MAC (with/ without VLAN) ≥ (fragmented IP or IP ≥ unknown).	IIPT, IPLEN, L4T, EIPT, EIPLEN = same as IP ≥ (fragmented IP or IP ≥ unknown). L4TUNT, L4TUNLEN = same as IP ≥ GRE / UDP ≥ MAC (with/without VLAN) -> IP -> L4.	Same as IP ≥ (fragmented IP or IP ≥ unknown).

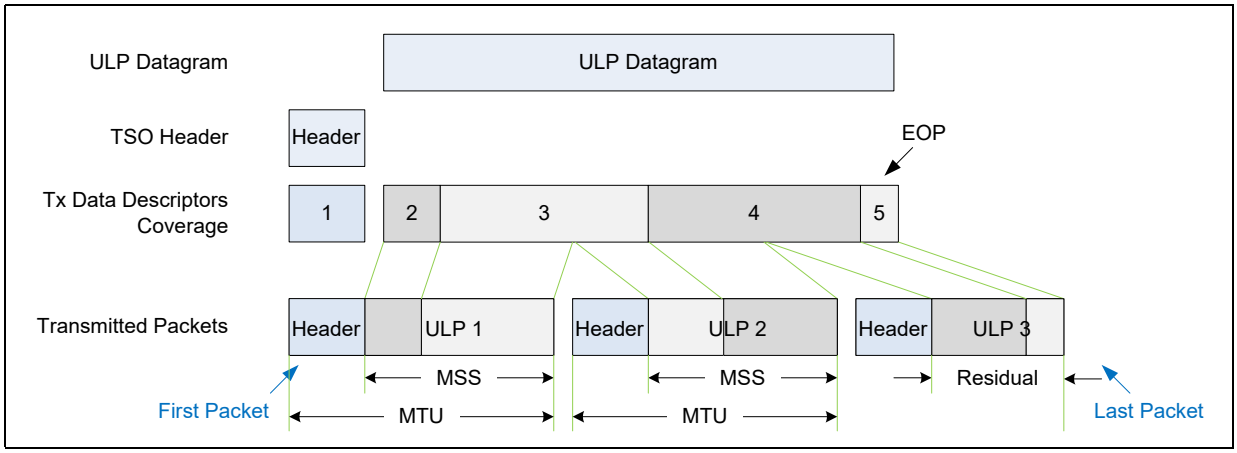
Notes:

1. Common settings to all cases: When context descriptor is used, the TSO flag should be cleared.
2. When IP-IP or other supported tunneling is transmitted, a context descriptor must be used to provide the additional fields types and sizes. For non-tunneled packets the context descriptor might be needed for other offloads than checksum. The values indicated in this table assumes that the context descriptor is used.

38.31.4.3 Transmit Segmentation Offload (TSO)

TSO, also called Large Send offload (LSO) enables the TCP/IP stack to pass to the network device a larger ULP datagram than the Maximum Transmit Unit Size (MTU). The 10 GbE controller divides the large ULP datagram to multiple segments according to the MTU size as shown in the Figure 38-101. The size of the ULP datagram supported for TSO can be as small as a single byte (transmitted on a single segment) and up to 256 KB (2^18) supporting TSO and giant TSO.

Figure 38-101.TSO Functionality (Example)



38.31.4.3.1 Frame Formats and Assumptions

The following packet formats are supported for TSO.

Note: For all the following formats, IP is IPv4 or IPv6 with/without option/extension headers. L4 is TCP.

- IP ≥ L4
- IP ≥ IP ≥ L4
- IP ≥ Teredo ≥ IP ≥ L4

- IP ≥ UDP Tunneling [Key] ≥ MAC ≥ [VLAN] ≥ IP ≥ L4
- IP ≥ GRE [Key] ≥ MAC ≥ [VLAN] ≥ IP ≥ L4
- IP ≥ GRE [Key] ≥ IP ≥ L4

SNAP packet formats are not supported for TSO. The following assumptions apply to the TCP segmentation implementation in the 10 GbE controller:

- TSO is activated by setting the TSO flag in the transmit context descriptor. On top of it, software should follow the data and context descriptor settings for integrity offload as listed in Table 38-436.
- The TSO header (L2, L3 and L4) should be provided by a maximum three descriptors, while still allowed to mix header and data in the last header buffer. The maximum size of the TSO header is 512 bytes.
- The maximum size of a single TSO can be as large as 256 KB minus 1 (defined by the *TLEN* field in the transmit context descriptor).
- The *RS* bit in the data descriptor can be set only on the last data descriptor of the TSO (on which the EOP bit is set).
- It is assumed that the software initializes the pseudo header checksum excluding the TCP length (as opposed to single send on which the pseudo header checksum includes the TCP length).

38.31.4.3.2 Transmit Segmentation Flow

The TSO flow includes the following steps:

- The protocol stack receives from an application a ULP datagram to be transmitted.
- The protocol stack calculates the number of packets required to be transmitted based on the MSS.
- The stack interfaces with the software device driver and passes the block down with the appropriate header information: Ethernet, IP header(s), optional tunneling headers and the L4 header.
- The stack interfaces with the software device driver and commands it to send the entire datagram. The software device driver sets up the interface to the hardware (via descriptors) for segmentation.
- Hardware fetches the segmentation parameters as well as the data and header buffers description by the transmit context and data descriptors.
- Hardware fetches the header and data buffers and then transmits them by segments according to the TSO parameters.
- Dynamic fields set by the software:
  - For IPv4 the IP header checksum should be set to zero.
  - The total *Length* field in the IP header(s) should be set to zero.
  - The IP ID of the first segment to be transmitted.
  - The pseudo header checksum of the TCP header should be calculated and placed as part of the packet data in the TCP or UDP checksum offset.
  - Initial value of the TCP flags (see the sections that follow for its value in the TSO packets).
- Dynamic fields in the IPv4 header(s) that are modified by hardware:
- The total *Length* field should reflect the IP payload size plus the IP header length. The L4 payload size is MSS for all packets but the last one which contains the rest of the data. So for the inner IP header (in case of tunneling), it is the L4 payload +

L4LEN + IPLEN. This is the same rule for the IP header in non tunneling case. For an outer IP header (for tunneling) it equals to: L4 payload + L4LEN + IPLEN + optional L4TUNT length + EIPLEN. Note that for MAC tunneling the length of the outer IP includes the inner L2 header including optional VLAN.

  - The *Identification* field (in the IP header in non-tunneled packets or the inner IP header in tunneled packets) is taken from the TSO header in the first segment and then it is increased by one for each transmitted segment.
  - The *Identification* field (in the external IP header) is taken from the TSO header in the first segment and then it is either increased by one for each transmitted segment or remains constant depending on the EIP_NOINC setting in the transmit context descriptor.
  - The header checksum is calculated after the other parameters in the IP header are updated.

- Dynamic fields in the IPv6 header(s) that are modified by hardware:
  - The payload length should reflect the payload size. It is the MSS for all packets but the last one that contains the rest of the data. The payload length should reflect the IP payload size. So for the inner IP header (in case of tunneling) it is the L4 payload + L4LEN + IP extensions. This is the same rule for the IP header in non-tunneling case. For an outer IP header (in case of tunneling) it equals to: L4 payload + L4LEN + IPLEN + optional L4TUNT length + IP extensions. The IP extensions length equals to IPLEN minus 40 or EIPLEN minus 40 for the inner or external IP headers, respectively. Note that for MAC tunneling, the length of the outer IP includes the inner L2 header including optional VLAN.
- Dynamic fields in the TCP header that are modified by the hardware:
  - The sequence number in the TCP header is taken from the TSO header in the first segment. It is then incremented by MSS for each transmitted segment.
  - The TCP flags are taken from the TSO header. The header is then masked (logic AND) by the *TCPMSKF*, *TCPMSKM* and *TCPMSKL* fields in the GLLAN_TSOMSK_F/M/L global registers. *TCPMSKF* is used for the first segment of the TSO; *TCPMSKL* is used for the last segment of the TSO, and *TCPMSKM* is used for all other segments of the TSO. If the TSO is composed of a single segment, then it is processed the same as the last segment of a multiple segment TSO.
  - The TCP checksum is calculated starting by the initial value (of the pseudo header). The checksum includes the updated TCP header, TCP payload and the calculated TCP length (equals to the payload size plus the TCP header size).
- Dynamic fields in a tunneling UDP header (Teredo or MAC in UDP) that are modified by the hardware:
  - The UDP length reflects the size of the tunneling UDP payload plus 8, which is the size of the UDP header.

38.31.4.3.3 Transmit Arbitration

The 10 GbE controller arbitrates between transmit queues at packet boundaries while enabling each queue to transmit at least pre-defined quanta (as long as the queue is not empty). TSOs are not an exception to this rule. If the queue exhausts its quanta in the middle of a TSO, the 10 GbE controller switches to the next queue in line at the end of the transmitted segment of the TSO. See the Transmit scheduling section for a description of transmit arbitration.

38.31.4.3.4 Segmentation Indication to Hardware

Software indicates a TCP segmentation by a transmit context descriptor just before the data descriptor with the following parameters:

- TSO flag is set, indicating a TSO is requested.
- Insert STag or external VLAN can be set only by the PF (the same as a single send).
- Switch control fields must not be enabled for TSO.
- MSS should be set to the required size of the L4 payload on each segment (MTU minus the size of the headers).
  - MSS outside the range (256 bytes to 9674 bytes) are considered malicious. The respective queue is stopped and an interrupt is issued to the PF. The relevant event is bad LSO MSS.
- TLEN is the total ULP datagram length.
- Other parameters in the data descriptor(s) and the context descriptor are defined the same as a single send.

The data descriptors indicate the TSO header as well as the ULP datagram while following the frame formats and assumptions described in Section 38.31.4.3.1.

38.32 TimeSync (IEEE 1588 and 802.1AS)

38.32.1 Overview

Measurement and control applications are increasingly using distributed system technologies such as network communication, local computing, and distributed objects. The 1588 standard enables accurate synchronization between clocks on distributed systems.

The 10 GbE controller includes 1588 logic per LAN port; the logic is enabled by the TSYNENA flag in the PRTTSYN_CTL registers per port. The PFs can access only the 1588 registers of a port. All PFs on a given port can access the 1588 registers while only one PF is expected to control the logic. The PF ID that owns the 1588 logic of the port is reflected by the PF_ID in the PRTTSYN_CTL0 registers per port. All PFs are permitted to read the 1588 timer registers (on the port they belong).

Software Note: The PF_ID field has no impact on the hardware; it is used only by the software. The field is expected to be loaded from the NVM or set by a management agent. During normal operation the PF software driver is not expected to change the field's setting.

The synchronized clock (PRTTSYN_TIME) is a 96-bit register (per LAN port). Out of it, the upper 64 bits are used for timestamp sampling and synchronized events described later in this section. Tune this register so that the upper 64 bits are defined in ns units. The register then can define the absolute time relative to PTP epoch, which is January 1st 1970 00:00:00 International Atomic Time (TAI).

38.32.1.1 1588 Registers

All 1588 registers are supported per port.

Many 1588 registers are 64 bits wide. The 10 GbE controller provides a 32-bit interface. Therefore, accessing these registers require multiple slave accesses.

- Reading any 64-bit register, software should access the LS register first and then the MS register. Specifically relating to dynamic registers: reading the LS register, the hardware latches internally the value of the MS register until it is fetched as

well. Included in this category are: PRTTSYN_TIME; PRTTSYN_RXTIME; PRTTSYN_TXTIME; PRTTSYN_INC; PRTTSYN_TGT and PRTTSYN_EVNT registers.

- Writing to any 64-bit registers. Software should write the LS register first and then the MS register. Only after writing the MS register is the value latched internally. Included in this category are: the PRTTSYN_TIME, PRTTSYN_INC, PRTTSYN_TGT, and PRTTSYN_EVNT registers. Writing to the PRTTSYN_TIME (*_L and *_H) registers, the device also clears the additional internal lowest 32 bits (usually used for sub ns units). Note that this sub ns register is internal, and not exposed to the software.

38.32.2 Time Synchronization Flow

The operation of a 1588 logic is based on Precision Time Protocol (PTP). This protocol is composed of two stages: initialization and time synchronization. These stages are described in the sections that follow emphasizing hardware and software roles.

38.32.2.1 Initialization Phase

If enabled as a potential master (by a software setting), software transmits sync packets that include the master's clock parameters periodically. Upon receipt of a sync packet, software on any potential master compares the received clock parameters to its own parameters. If the received parameters are better, software transitions to a slave state and stops sending sync packets.

While in a slave state, software selects a particular master. Software continuously compares the received sync packet to its selected master. If the received sync packet belongs to a different master with better clock parameters, software switches to the new master. Eventually only one master (with the best clock parameters) remains active while all other nodes act as slaves listening to the single master.

Every node has a defined time-out interval. If no sync packets are received from the selected master each slave, software switches back to the initialization phase until a new master is chosen. Note that there are more than one option for the previous flow while there are other flows that are based on static master setting. The node can be set statically to a master or slave modes. While in slave mode, the node can be tuned statically to a specific master.

38.32.2.2 Time Synchronization Phase

There are two phases to the synchronization flow:

1. The slave calibrates its clock to the master.
2. The master performs complete synchronization.

38.32.2.2.1 Clocks Calibration

The master sends sync packets periodically (about of 10 packets per second). These packets are followed by Follow_UP packets that indicate the transmission time. The slave captures the reception time of the sync packet so it holds the packet transmission time at the master and its reception time at the slave. Receiving consecutive sync packets, the slave gets the delta T of the master and can calibrate its timer to get the same delta T. During this phase, the slave starts with an initial time calibration that can be used as the transmission delay between the master and the slave.

In order to minimize sampling inaccuracy, both master and slave sample the packets transmission and reception time at a location in the hardware that has as much as possible deterministic delay from the PHY interface.

- Software hardware flow in the master — Sending the sync packet by the master, software indicates this packet to the hardware by setting the TSYN flag in the transmit context descriptor. Setting this flag, hardware samples its transmission time using the PRTTSYN_TXTIME register. After transmission, software should read the PRTTSYN_TXTIME register and sent the transmission time in a Follow_Up packet (according to a two-step flow). Note that software is responsible to read the PRTTSYN_TXTIME register before initiating another packet with an active TSYN flag.
- Software hardware flow in the slave — The sync packet is received by the slave and its reception time is sampled using one of four PRTTSYN_RXTIME registers. The index of the register that sampled the reception time is reported in the *TSYNINDX* field in the receive descriptor. The PRTTSYN_RXTIME register holds the reception timestamp until software reads it. It is then released for sampling another packet reception.

38.32.2.2.2 Time Synchronization Phase

The complete synchronization scheme is shown in Figure 38-102. It relies on measured time stamp of sync packets transmission and reception by the master and the slave. The scheme is based on two assumptions:

- The clocks at both nodes are almost identical (achieved in the first step).
- Transmission delays between the master to the slave and backward are symmetric.

  The master's software sends periodic sync packets to each slave followed by a Follow_Up packet, as explained in the software hardware flow in the master for the clocks calibration phase). The slave samples the TSYN packet reception time in one of the PRTTSYN_RXTIME registers.

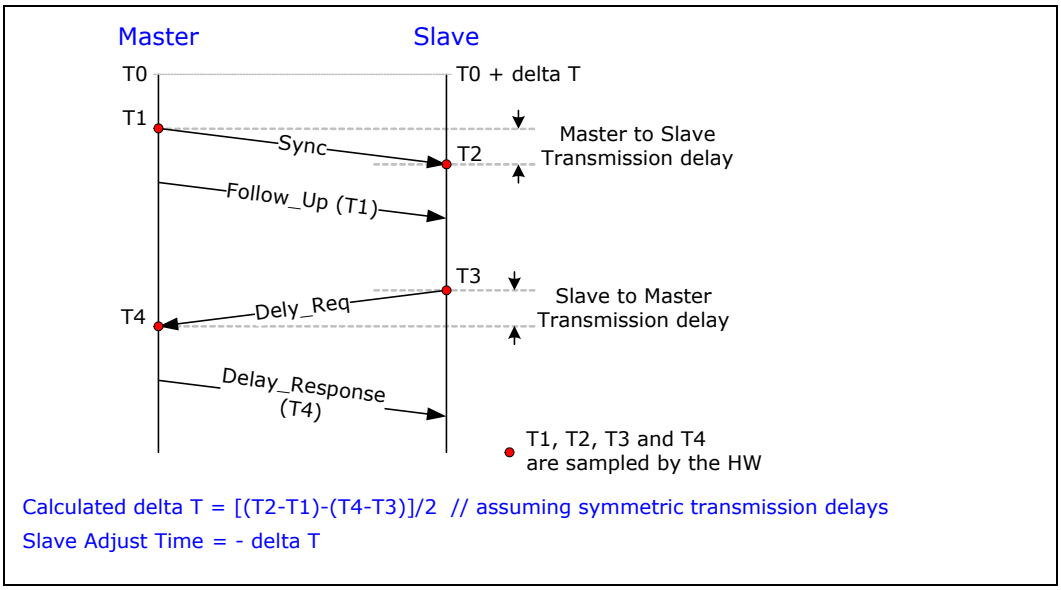
The slave responds, sending a Delay_Request packet to the master. Hardware in the slave samples its transmission time in the PRTTSYN_TXTIME register and hardware in the master samples its reception time in one of the PRTTSYN_RXTIME registers. Software in the slave checks the Delay_Request packet transmission time (reading the PRTTSYN_TXTIME register) and could optionally send a Follow_Up packet that includes the captured transmission time. The master responds back with the Delay_Response packet reporting the reception time of the Delay_Response to the slave. At this point, the slave has all the following parameters (the following notations match the notations used in Figure 38-102):

- T1 — Sync packet transmission time in the master (based on master clock).
- T2 — Sync packet reception time in the slave (based on slave clock).
- T3 — Delay_Request transmission time in the slave (based on slave clock).
- T4 — Delay_Request reception time in the master (based on master clock).

The Slave can adjust its clock using the following equation:

- Slave adjust time = - [(T2-T1) - (T4-T3)] / 2.

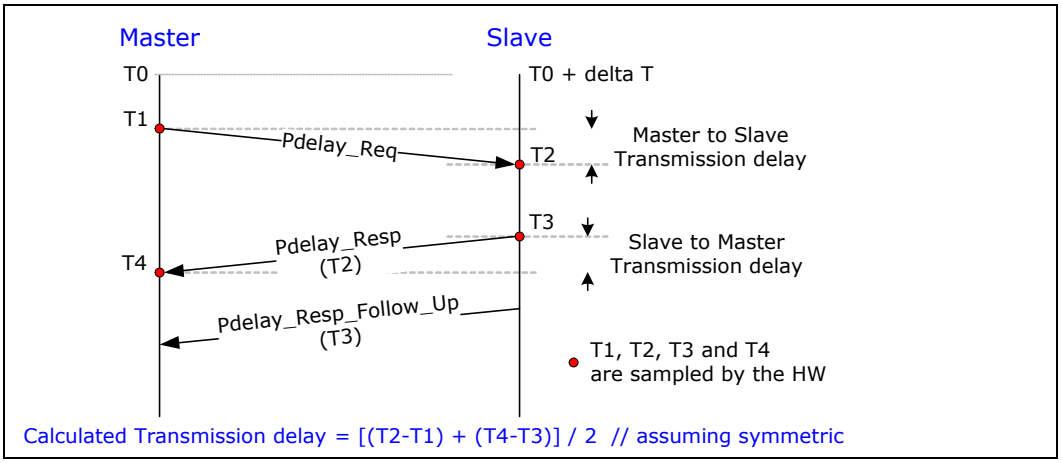
Figure 38-102.Sync Flow and Offset Calculation



PDelay Flow for Dynamic Master Selection

The master sends a Pdelay_Req packet to the slave and the slave response by sending back a Pdelay_Resp packet followed by a Pdelay_Resp_Follow_Up packet. The master uses these packets to calculate the link delay. This process is also used for fast recovery when the master is changed. The process is usually enabled when dynamic master selection is enabled. It can operate asynchronous to the time synchronization flow previously described. Figure 38-103 shows the PDelay flow.

Figure 38-103.PDelay Flow



38.32.3 Time Stamp Indication

Relevant packet transmission time (as sync and Delay_Request) are sampled by hardware at a deterministic as possible affinity to the PHY interface. The sampled time is taken at the beginning of the packet (packet size does not matter) as shown. The latency between packet time sampling and packet on the Ethernet interface pins is listed in Table 38-437.

Figure 38-104.Time Stamp Point

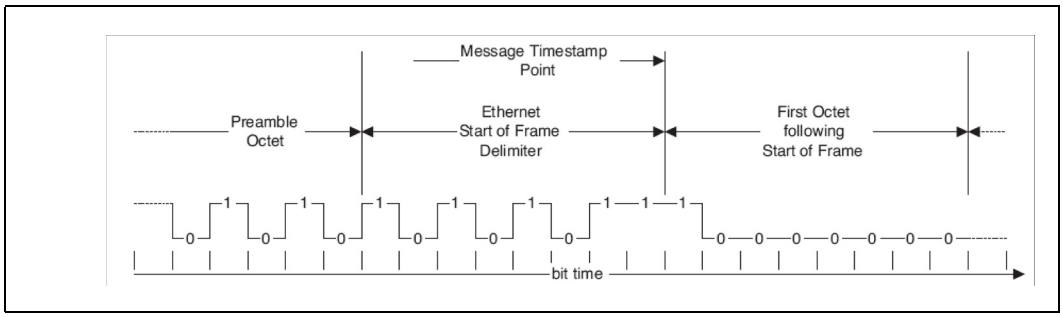


Table 38-437.Packet Time Stamp Latency

Link Speed	Interface	Time Stamp to Packet Transmission Latency	Packet Reception to Time Stamp Latency
10 GbE	KR / SFI		
1 GbE	KX		

38.32.3.1 Transmit Time Stamp

Software indicates to hardware the packets to be sampled by setting the *TSYN* bit in the transmit context descriptor. Hardware samples the transmission time of packets with an active *TSYN* bit, enabled for the queue by the TSYNENA flag in the transmit queue context. The transmission time stamp is sampled in the PRTTSYN_TXTIME register. It is software's responsibility to read the PRTTSYN_TXTIME_L and PRTTSYN_TXTIME_H registers before sending another packet with an active *TSYN* bit.

Note that in order to use the TSYN flag in the transmit descriptor, the TSYNENA flag should be set in the transmit queue context. Software should set the TSYNENA flag only for PF queues so VFs cannot interfere with TimeSync functionality. Software should limit this enablement on a single PF per port that is assigned to control the 1588 logic.

38.32.3.2 Receive Time Stamp

The 10 GbE controller identifies received PTP event packets as described in Section 38.32.9. PTP packet reception time is sampled by one of four PRTTSYN_RXTIME[n] registers per port, which is found free. Such packets are posted to LAN queues with active an 1588 TS Index (TSYNINDX) field in the receive descriptor.

TSYNINDX is the index of the PRTTSYN_RXTIME register that contains the packet reception time stamp. Once a PRTTSYN_RXTIME register samples a packet reception time, the register is locked until software reads its value. When accessing this register software should read first the PRTTSYN_RXTIME_L[n] register and then the PRTTSYN_RXTIME_H[n] register. Following read accesses to the PRTTSYN_RXTIME register, the lock is released and the register becomes free again.

Note that PRTTSYN_RXTIME registers are not accessible to any of the VFs (so they cannot interfere with the proper functionality). PRTTSYN_RXTIME are 64-bit registers that capture the upper 64 bits of the PRTTSYN_TIME register that belong to the LAN port on which the PTP packet was received (accessible to software by 2 x 32-bit accesses).

38.32.4 1588 Clock Registers

The clock driving the time sync elements is the MAC clock on the port. The MAC clock frequency depends on the link speed (Table 38-438). After a change in link speed the PRTTSYN_INC register should change accordingly. Note that software is responsible for this setting.

Table 38-438.PRTTSYN_INC as a Function of MAC Clock Frequency

Link Speed	MAC Clock Frequency	Sampling Clock Precision	Recommended PRTTSYN_INC Defining The 64 MS Bits Of PRTTSYN_TIME In ns Units
40 Gb/s	625 MHz	± 0.8 ns	6871947673 (0x199999999)
10 Gb/s	312.5 MHz	± 1.6 ns	13743895347 (0x333333333)
1 Gb/s	31.25 MHz	± 16 ns	137438953472 (0x2000000000)

The PRTTSYN_TIME is a 96-bit register holding the synchronized clock while its upper 64 bits are meaningful to software accessible by 2 x 32-bit accesses. Each MAC clock, the PRTTSYN_TIME is increment by PRTTSYN_INC. The lower 32 bits of the PRTTSYN_TIME register are internal, not accessible to software.

PRTTSYN_INC is a programmable 38-bit register accessible to software by 2 x 32-bit accesses (the upper 26 bits of this register are reserved and set to zero). When updating this register, software should set first the PRTTSYN_INC_L register and then the PRTTSYN_INC_H register. Hardware updates the internal value of the PRTTSYN_INC only after these two write cycles.

Software in slave mode adjusts PRTTSYN_TIME in order to track the 1588 timer of the master using one of two methods:

- Direct write access to the upper 64 bits of the PRTTSYN_TIME register. This step can be useful for an initial setting of the time during the clocks calibration phase.
- Fine adjustment during the 1588 time synchronization phase using the PRTTSYN_ADJ register. The PRTTSYN_ADJ register is composed of a SIGN flag and *TSYNADJ* field that defines the absolute value of the adjustment. The *TSYNADJ* value is defined in the units of the PRTTSYN_TIME_L. The PRTTSYN_TIME is updated iteratively by the PRTTSYN_ADJ together with the nominal INC value on each MAC clock as shown:

```
If (TSYNADJ = 0) then:

PRTTSYN_TIME += PRTTSYN_INC

Else then:     // TSYNADJ is not equal to zero

If positive adjust (SIGN=0) then: PRTTSYN_TIME += PRTTSYN_INC + 2^32

Else : PRTTSYN_TIME += PRTTSYN_INC - 2^32

TSYNADJ -= 1
```

38.32.5 Synchronized Auxiliary Events

The 10 GbE controller supports the following global auxiliary events for all ports:

- Synchronized events to global I/O signals are described in Section 38.32.5.1.
- Synchronized events to PCI slave access are described in Section 38.32.6.1

38.32.5.1 Auxiliary 1588 I/O Signals

The 10 GbE controller supports 30 global GPIO signals that are controlled by 2 PRTTSYN_AUX[n] registers (n = 0,1). GPIO 'n' functionality is controlled by the GLGEN_GPIO_CTL[n] register (expected to be loaded from the NVM). GPIOs can be assigned to 1588 of a specific LAN port by the PRT_NUM and PIN_FUNC fields in the GLGEN_GPIO_CTL register. Setting the PIN_FUNC field to *Timesync 0* or *Timesync 1* assigns the GPIO to PRTTSYN_AUX[0] or PRTTSYN_AUX[1], respectively.

GPIO signals are set to input or output by the PIN_DIR field in the GLGEN_GPIO_CTL registers. When set to input signal the sampling event is sampled by the PRTTSYN_EVNT[n] registers ('n' is the matched index to the PIN_FUNC field). When set to output signal, the output event time is defined by the PRTTSYN_TGT[n] register ('n' is the matched index to the PIN_FUNC field).

There are several output modes of operation as described in the sections that follow. In any of them, the initial state of the GPIO output level can be controlled by setting the INSTNT and the OUTLVL flags in the PRTTSYN_AUX[n] register. All the output modes are enabled by the OUT_ENA flag in the PRTTSYN_AUX[n] register. There are also several input modes of operation controlled by the *EVNTLVL* field in the PRTTSYN_AUX[n] register.

Synchronized level output — This mode is selected by setting the PRTTSYN_AUX[n].OUT_ENA, setting the PRTTSYN_AUX[n].OUTMOD field to output level mode and the event time is defined by the matched PRTTSYN_TGT[n] register. When the upper 64 bits of the PRTTSYN_TIME crosses the value of the PRTTSYN_TGT[n], the assigned GPIO transits to the requested level defined by the PRTTSYN_AUX[n].OUTLVL.

Synchronized flipped output signal — Defined by the PRTTSYN_TGT[n] and PRTTSYN_AUX[n] registers similar to the synchronized level output mode. The only difference is that for flipped output, the *OUTMOD* field should be set to flipped output mode. In this case, each time the upper 64 bits of the PRTTSYN_TIME crosses the value of the PRTTSYN_TGT[n] the I/O signal flips its output level.

Synchronized output pulse — Defined by the PRTTSYN_TGT[n] and PRTTSYN_AUX[n] registers similar to the synchronized level output mode. The only difference is that the *OUTMOD* field should be set to output pulse mode. In this case, each time the upper 64 bits of PRTTSYN_TIME crosses the value of PRTTSYN_TGT the GPIO signal flips its output state for 16 x PRTTSYN_AUX.PULSEW plus 1 MAC clock and then reverts back to its previous level.

Synchronized clock output — Defined by the PRTTSYN_TGT[n] and PRTTSYN_AUX[n] registers similar to the synchronized level output mode with the following changes: The *OUTMOD* field should be set to output clock mode and the matched PRTTSYN_CLKO[n] register should be set to 50% of the required output clock duration. Each time the upper 64 bits of the PRTTSYN_TIME crosses the value of PRTTSYN_TGT the GPIO signal flips its output level. Then, the PRTTSYN_TGT register is reloaded by hardware to PRTTSYN_TGT[n] plus PRTTSYN_CLKO[n] (while PRTTSYN_CLKO is padded by 32 MS bit zero's). Note that for proper operation the PRTTSYN_CLKO must be larger than PRTTSYN_INC + 2^{32} and for reasonable accuracy the PRTTSYN_CLKO should be significantly larger than PRTTSYN_INC + 2^{32}.

Sampling input event — The 10 GbE controller can capture the time on which an event is sensed on any of the 1588 auxiliary signals. The event time is captured by one of two matched 64-bit event time registers named PRTTSYN_EVNT[n]. The sampled input event type is defined by the *EVNTLVL* field in the PRTTSYN_AUX[n] register, equals to one of the following options: disable, rising edge, falling edge, or any transition. When

the defined transition is sampled by hardware (synchronized to the MAC clock), the upper 64 bits of PRTTSYN_TIME are latched by the matched PRTTSYN_EVNT[n] register. Once the event is sampled, the PRTTSYN_EVNT register is locked, keeping the event time until software reads it.

38.32.6 Synchronization with Host Timer

The time relationship between the CPU time and devices within the system is a key element in many applications. Specifically, the relationship with the 1588 clock is critical. The following sub-section describes the mechanism to measure this relationship.

38.32.6.1 1588 PCI Slave Access

Following a write access to PRTTSYN_AUX[n] while setting the SAMPLE_TIME flag, hardware samples PRTTSYN_TIME to the matched PRTTSYN_EVNT[n] register (the upper 64 bits of the PRTTSYN_TIME). This functionality can be useful to synchronize between a system timer accessible to the CPU and PRTTSYN_TIME.

38.32.7 Interrupts

The 10 GbE controller can trigger an interrupt on any GPIO regardless of whether it is enabled for 1588 functionality. In addition, the 10 GbE controller can generate a 1588 interrupt for one of the following events (if enabled by the TIMESYNC flag in the PFINT_ICR0_ENA register):

- Packet transmission time is latched by the PRTTSYN_TXTIME registers while this interrupt is enabled by the TXTIME_INT_ENA flag in the PRTTSYN_CTL0 register.
- Input event is latched by one of the PRTTSYN_EVNT registers while this interrupt is enabled by the EVENT_INT_ENA flag in the PRTTSYN_CTL0 register.
- One of the time registers has expired while this interrupt is enabled by the TGT_INT_ENA flag in the PRTTSYN_CTL0 register (a target time register expires when the PRTSYN_TIME register is larger or equal than the target time register).

Regardless of interrupt enablement, the previous events are reported in the PRTTSYN_STAT_0 and PRTTSYN_STAT_1 registers. Following an interrupt assertion, software should read these registers to identify the cause. The registers are cleared on read, so they report an updated status since the last time they were read by software.

38.32.8 1588 Initialization Flow

This section describes the PF software initialization flow required to activate the 1588 logic.

- Check the PF_ID in the PRTTSYN_CTL0 register if the PF is expected to control the 1588 logic.
- The 1588 CSRs are initialized by global reset that might not be triggered by a PCI reset. Therefore, PF software is required to initialize most of the 1588 registers listed in the TimeSync (IEEE 1588) registers section:
  - Set PRTTSYN_CTL0 and PRTTSYN_CTL1 to the required 1588 packet type as well as required interrupt functionality.
  - Clear any optional residuals reported in the PRTTSYN_STAT register.
  - Read all receive time stamps releasing them for normal operation (PRTTSYN_RXTIME registers).

  - — Set the PRTTSYN_AUX register as required for the AUX functionality.
  - — The timer and the increment values are expected to be programmed as part of normal operation (PRTTSYN_TIME and PRTTSYN_INC registers).
- Assign a transmit queue for 1588 transmission and enable the TSYNENA flag in its queue context.
- Optionally assign a receive queue. The Ether-type classification filter can be used to direct layer 2 or UDP 1588 packets to a specific LAN receive queue.

38.32.9 PTP Packet Recognition

The time sync implementation supports both the 1588 Version 1 (V1) and Version 2 PTP frame formats (V2). The V1 structure is expected only as UDP payload over IPv4 while the V2 is expected over L2 with its Ether-type or as a UDP payload over IPv4 or IPv6. Note that the 10 GbE controller accepts the expected packet formats but does not enforce these UDP/IP rules.

The 802.1AS uses only the layer 2 V2 format. Note that PTP frame structure is not supported in the 10 GbE controller for tunneled packets. The layer 2 packet format and UDP packet format are shown in Figure 38-105. The PTP message V1 and V2 formats are shown in Figure 38-440. The packet format supported by the 10 GbE controller for receive time stamp is controlled by the parameters shown in Figure 38-439.

Note: Correct UDP checksum is not a criteria for PTP packets recognition by the 10 GbE controller. Still, such a packet is reported to software with a checksum error. Software can ignore the packet, but should read the register that carries the time stamp to release it for upcoming packets.

Note: The 10 GbE controller does not recognize tunneled PTP packets.

Table 38-439.Receive Time Stamp Control Parameters

Parameter	Register	Functionality
TSYNTYPE	PRTTSYN_CTL1	Controls which packets are sampled by the 1588 logic. It can be one of the following: • L2 version 2 packets while message type is defined by the PRTTSYN_CTL1 register. • UDP version 1 packets while the UDP ports are defined by UDP_ENA and message type is defined by the PRTTSYN_CTL1 register. • L2 and UDP version 2 packets while UDP ports and message type are defined as previously detailed. • L2 and UDP version 2 event packets while UDP ports are defined by the UDP_ENA and the message type < 8.
UDP_ENA	PRTTSYN_CTL1	Enable the UDP port numbers that are recognized as 1588 packets. It could be one of the following options: • No UDP packet recognition. • UDP port number equals 0x013F. • UDP port number equals 0x0140. • UDP port numbers equals either 0x013F or 0x0140.
V1MESSTYPE0, V1MESSTYPE1	PRTTSYN_CTL1	Controls the PTP V1 message type for sampled time stamp.
V2MESSTYPE0, V2MESSTYPE1	PRTTSYN_CTL1	Controls the PTP V2 message type for sampled time stamp.

Figure 38-105.1588 Packet Format

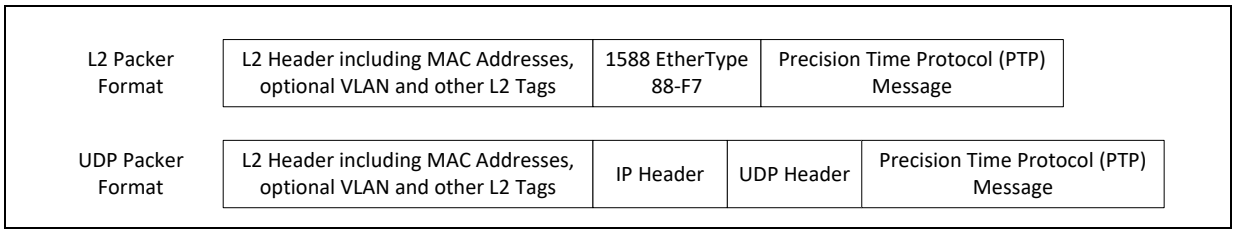


Table 38-440.PTP Header Format (V1 and V2 Formats)

Offset in Bytes	V1 Fields	V2 Fields	
		bits: 7 . . . 4	bits: 3 . . . 0
0	versionPTP	transportSpecific[1]	Message Type
1		Reserved	versionPTP
2 - 3	versionNetwork	messageLength	
4	Subdomain	SubdomainNumber	
5		Reserved	
6 - 7		flags	
8 - 15		Correction Field	
16 - 19		reserved	
20	Message Type	Source Port ID	
21	Source communication technology		
22 - 27	Sourceuuid		
28 - 29	sourceportid		
30 - 31	sequenceId	sequenceId	
32	control	control	
33	reserved	logMessagePeriod	
34 - 35	flags	n/a	

Notes:

1. Should all be zero.

Table 38-441.PTP V2 and V1 Message Types

Message Type	Message Class	V2 Message Type	V1 Message Type
Sync	Event	0x0	0x00
Delay_Req	Event	0x1	0x01
Pdelay_Req	Event	0x2	N/A
Pdeley_Resp	Event	0x3	N/A
Reserved	-	0x4 - 0x7	N/A
Follow_Up	General	0x8	0x02
Delay_Resp	General	0x9	0x03
Pdelay_Resp_Follow_Up	General	0xA	N/A
Announce	General	0xB	N/A
Signaling	General	0xC	N/A
Management	General	0xD	0x04
Reserved	-	0xE, 0xF	0x05 - 0xFF

38.33 Software Timer

The 10 GbE controller includes a 32-bit counter in the GLVFGEN_TIMER register that is based on a free running 1 µs clock. The counter is cleared by Core Reset (CORER) and increments after being cleared. The timer wraps around in about 70 minutes.

38.34 Protocol Engine (PE)

38.34.1 PE Overview

The Protocol Engine (PE) adds Remote Direct Memory Access (RDMA) capabilities to the traditional LAN functionality found in standard NICs. RDMA is a networking performance optimization that enables servers to communicate across a network using high-performance, low-latency, zero-copy DMA semantics. It is designed to reduce host CPU use, host memory bandwidth used for network traffic, and network latency when compared to traditional networking stacks such as sockets with TCP/ IP. Following are some of the RDMA capabilities provided by PE:

- Secure, direct access to PE hardware for user space or kernel applications
  - — Supported in VM environments using SR-IOV
- Translation Protection Table (TPT)
  - — Similar to a CPU Memory Management Unit (MMU)
  - — Enables definition of > 1 M memory regions
  - — A memory region is a virtually or logically contiguous area of application address space registered with the operating system. An enabled memory region enables the PE to perform DMA access for local and (optional) remote requests, and enables user space applications to specify buffers in virtual address space.
  - — 100 M+ virtual-to-physical page translations
- Reliable Connection Transport
  - — 100 K+ connections, analogous to a DMA engine with 100 K+ channels
  - — Sequence number checks
  - — Acknowledgment
  - — Dynamic congestion control
  - — Retransmission
  - — Timers
- IP routable RDMA

38.35 Features

Table 38-442 lists the PE feature set.

Table 38-442.PE features

Category	Description
General	**RDMA Protocol Support (Baseline)** Support for either iWARP or RoCE v2 (both protocols are not simultaneously available). Configure either iWARP or RoCE v2 from the NVM.
General	**RDMA Protocol Support (Enhanced)** Simultaneous support for both iWARP and RoCE v2.
General	**Ethernet Ports Supported** Support for 4 Ethernet ports with 8 TCs, or 8 Ethernet ports with 4 TCs.
General	**IP Version** Support both IPv4 and IPv6 offloaded RDMA connections.
General	**ARP Table** Supports a unique ARP table instance with up to 65536 entries per PCI function (PF or VF). Each entry contains a six byte Ethernet address, plus control/status information for neighbor reachability detection. The ARP table supplies a destination Ethernet address for all transmitted RDMA packets.
General	**IP Datagrams / Fragmentation** RDMA IP datagrams are never transmitted with fragmentation. RDMA IPv4 datagrams are always transmitted with the *Don't Fragment* flag set and the *Fragment Offset* field set to 0b. Received RDMA IP datagram fragments are sent to the LAN stack and not processed by the RDMA offload engine.
General	**Per vPort RDMA Statistics** Implement one set of RDMA statistics for each RDMA-enabled VSI. For example, if the IP supports 8 PCIe PFs and 32 PCIe VFs, then it must implement 40 sets of RDMA statistics.
QP	**Max QP Count** Supports 256 K queue pairs for RDMA that are dynamically assigned to PCI functions at runtime by PE firmware.
QP	**Work Queue Elements (WQEs) - Baseline** These WQE properties are supported: Similar WQE format for SQs and RQs. Each iWARP WQE can vary in size from 1 to 7 fragments, with these corresponding sizes: 32 bytes, 48 bytes, 64 bytes, 80 bytes, 96 bytes, 112 bytes, 128 bytes. Each RoCE WQE can vary in size from 1 to 5 fragments, with these corresponding sizes: 64 bytes, 80 bytes, 96 bytes, 112 bytes, 128 bytes. Fragments are specified with virtual addresses and are virtually contiguous. The PE performs virtual-to-physical translation using its built-in Memory Management Unit (MMU). An iWARP SQ WQE can optionally directly convey 112 bytes of in-line data for small message latency optimization. A RoCE SQ WQE can optionally directly convey 80 bytes or more of in-line data for small message latency optimization.
QP	**Work Queue Elements (WQEs) - Enhanced** Support the features defined by Work Queue Elements (WQEs) - Baseline plus increase the max WQE size from 128 bytes to 256 bytes. This could enable a larger number of fragments per WQE, or a larger amount of in-line data for small message latency optimization. This enhancement appears more valuable on the SQ than the RQ, and might be implemented only on the SQ side. Send queue push mode support is required.
QP	**Work Queues (WQs)** RDMA WQs are mapped into host memory as physically or virtually contiguous ring buffers. Maximum WQ size is 16 K entries x 32 bytes = 512 KB. Maximum WQ depth is configurable on a per-WQ basis. Supported values range from 4 to 16 K 32 bytes WQEs, in power-of-two increments. Dynamic WQ resizing is not supported.
QP	**SQ Operations (baseline)** Support for these baseline SQ operations: Send, Send with Invalidate, Send with Solicited Event, Send with Solicited Event and Invalidate, RDMA Write, RDMA Read (iWARP-style single local SGE), RDMA Read with Local Invalidate (iWARP-style single local SGE), Bind Memory Window, Fast-Register Memory Region, Invalidate Local STag. These operations operate with either iWARP or RoCE v2, except the iWARP-style RDMA Reads, which are iWARP-only.
QP	**SQ Operations (RDMA Write with Immediate Data)** Support for RDMA write with immediate fata. CQE and SQ WQE formats enable 8 bytes of immediate data for both iWARP and RoCE. For iWARP, 8 bytes of immediate data is conveyed on the network. For RoCE, only 4 bytes of immediate data is conveyed on the network.

Table 38-442.PE features

Category	Description
QP	**SQ Operations (RDMA Read special)** Support RDMA read (IB-style with multiple local SGEs). For RoCE v2, this is the only form of RDMA read. iWARP must support all three RDMA read operation types simultaneously on a given QP: RDMA read (iWARP-style single local SGE), RDMA read with local invalidate (iWARP-style single local SGE), RDMA read (IB-style with multiple local SGEs).
QP	**SQ Operations (Send with Immediate)** Support Send with Imm, Send with Inv and Imm, Send with SE and Imm, Send with Inv and SE and Imm
QP	**Number of RDMA Reads** The number of outstanding inbound RDMA reads, defined by Inbound RDMA Read Queue Depth (IRD) is configured independently per QP. These IRD settings are supported: 2, 8, 32, 64, 128, 256. IRD cannot be modified after the QP has been created. The number of outstanding outbound RDMA reads, defined by Outbound RDMA Read Queue Depth (ORD), is also configured independently per QP. These ORD settings are supported: 0 to 127255, in single-step increments. ORD can be modified after the QP has been created, if the proper quiesce conditions are met.
QP	**Send Queue Push Mode** This device supports send queue *Push Mode*. In this mode, the host CPU writes or pushes SQ WQEs with *inline data* to the device memory-mapped address space using CPU write-combining buffers. Each of the device's PCIe PFs must support up to 1024 separate 4 KB push pages. Each of the device's RDMA-enabled VFs must support 16 4 KB push pages. The push pages are typically exposed through an extension of the memory BAR that contains the CSRs. PF push pages must be assignable to VMs via device firmware in para-virtualized driver models.
CQ	**Completion Queues** Supports up to 131072524288 CQs that are dynamically assigned to PCI functions at runtime by device firmware. A CQ/ring buffer can be either virtually or physically contiguous. Each CQE is 32 bytes, and can optionally be padded to a 64-byte cache-line boundary to avoid memory conflicts. Max CQ size is 1 M entries x 32 bytes = 32 MB (or 64 MB if optional 64 byte CQE size is used). Supports user-defined mapping of WQs to CQs. Supports CQ resizing, CQ size can be increased or decreased while the CQ is active. Supports CQ overflow detection.
EQ	**Completion Event Queues (CEQs)** Supports 256 CEQs for RDMA that are dynamically assigned to PCI functions at runtime by device firmware.
EQ	**Asynchronous Event Queues (AEQs)** Supports 48 AEQs for RDMA that are dynamically assigned to PCI functions at runtime by PE firmware.
MMU	**Protection Domains** Each PCIe PF or VF enabled to use RDMA can define up to 32256 K protection domains.
MMU	**Memory Regions and Windows** Each PF or VF enabled to use the PE can allocate up to 4 M (like 2^{22}) memory regions or memory windows. The maximum size of a single memory region or memory window matches the maximum virtually mapped memory supported per PF or VF.
MMU	**Wide Memory Windows** Support wide memory windows for both iWARP and RoCE. Wide memory windows have protection domain scope, whereas standard iWARP memory windows have QP scope.
MMU	**Host Memory Page sizes (baseline)** Host memory page sizes supported: 4 KB, 2 MB. Each memory region can be independently configured for either page size.
MMU	**Host Memory Page sizes (enhanced)** Includes everything in baseline, plus support for 1 GB host memory page size.
MMU	**Physical Buffer List (PBL)** Each RDMA memory region can be physically mapped (such as the region is physically contiguous in host memory) or virtually mapped with a one- or two-level PBL. Each RDMA-enabled PF or VF can allocate up to 256 M (like 2^{28}) PBL entries. A single PBL entry maps a single host memory page.
MMU	**Maximum Virtually Mapped Memory (baseline)** The maximum virtually mapped memory a single PF or VF can register for RDMA is implementation limited to 2^{45} bytes = 32 TB. The maximum might be further reduced depending on host page sizes used. For example, if a PF or VF allocates it's maximum limit of 256 M PBL Entries, then... ...its maximum virtually mapped memory using 100% 2 MB host memory pages is: MIN(2^{45}, 2^{21} x 2^{28}) = 2^{45} bytes = 32 TB ...its maximum virtually mapped memory using 100% 4 KB host memory pages is: MIN(2^{45}, 2^{12} x 2^{28}) = 2^{40} bytes = 1 TB

Table 38-442.PE features

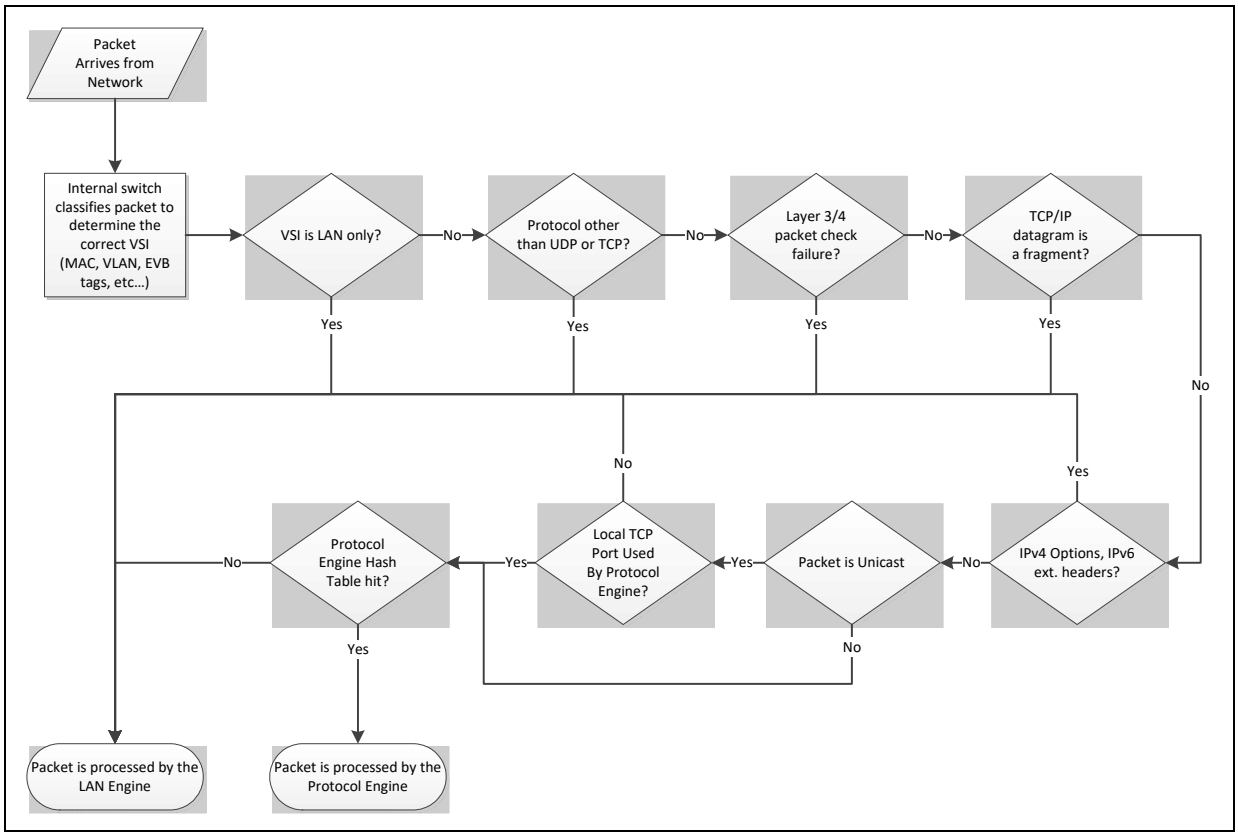
Category	Description
MMU	**Maximum Virtually Mapped Memory (enhanced)** The maximum virtually mapped memory a single PF or VF can register for RDMA depends on host page sizes used. The following examples show this for different page sizes. Supported page sizes are defined elsewhere, and some of these examples might not be relevant. If a PF or VF allocates it's maximum limit of 256 M PBL entries, then... ...its maximum virtually mapped memory using 100% 1 GB host memory pages is: 2^{30} x 2^{28} = 2^{58} bytes = 256 PB ...its maximum virtually mapped memory using 100% 2 MB host memory pages is: 2^{21} x 2^{28} = 2^{49} bytes = 512 TB ...its maximum virtually mapped memory using 100% 4 KB host memory pages is: 2^{12} x 2^{28} = 2^{40} bytes = 1 TB
UD	**Address Handles** Supports {minimum 64K, target 256K} Address Handles per PCIe function.
iWARP	**Standards Compliance** IETF RFC 5040, 5041, 5042, 5044, 6580, 6581, 7306 RDMA Consortium Verbs
iWARP	**Receive Window Size** Each RDMA connection has a configurable Receive Window with maximum size of 1GB-1B.
iWARP	**Non-permissive IETF RNIC** Using a term coined in IETF RFC 5044, this device is a Non-permissive IETF RNIC (an RNIC that implements the IETF protocols but not the RDMAC protocols)
iWARP	**MPA - Baseline support** Support all of the following: Insertion of Transmit markers (can be enabled/disabled per QP). MPA CRC generation for outbound iWARP packets. Transmit up to four FPDUs in a single Ethernet packet. Able to transmit partial FPDUs when there is outstanding (un-acknowledged) data and an MSS change occurs. MPA CRC checking on inbound packets (can be enabled/disabled per QP). Process received Ethernet packets with any number of iWARP FPDUs (bounded by max packet size). Support for detection/handling of received partial FPDUs. Drop out-of-order received FPDUs without ACKing the TCP segment containing them.
iWARP	**MPA - place out-of-order received FPDUs** Support a per-QP setting to configure either 1 or 2 below: 1. Drop out-of-order received FPDUs without ACKing the TCP segment containing them 2. Placement of out-of-order received MPA FPDUs, with one restriction: An out-of-order received MPA FPDU with Send opcode will only be placed if it targets a non-shared RQ with dedicated CQ.
RoCE	**Standards Compliance** InfiniBand Architecture Specification Volume 1 Release 1.3 (support the subset of this specification that applies to RoCE v2) InfiniBand Architecture Specification Annex A17: RoCE v2 Sept 2014

38.36 Functional Description

38.36.1 Packet Classification and the PE

The 10 GbE controller provides several filtering checks that can be used to distinguish LAN traffic that should be handled by the PE from other LAN traffic. The first mechanism that can be used to direct a packet to be processed by the PE is to set the PETCP_ENA bit in the VSIQF_CTL register. Additionally all unicast or multicast UDP traffic might be forwarded to the PE by setting the PEUUDP_ENA and/or PEMUDP_ENA bits in the VSIQF_CTL register. The internal switching components including this special setting are used to identify a Virtual Station Interface (VSI). Once the VSI has been identified, the checks shown in Figure 38-107 determine if a packet is processed by the PE.

Figure 38-106.PE Packet Classification



Once the internal switch packet classification has been performed, the following checks must be passed before a packet is processed by the PE:

1. The VSI configuration option that enables traffic to be routed to the PE must be enabled.
2. The packet must be a protocol that is supported by the PE. The current protocols that the PE is capable of handling are TCP/IP and UDP/IP. Multicast and unicast packets are supported for UDP/IP.
3. Additional checks to validate the IP and TCP or UDP headers must succeed.
4. If the packet is an IP packet it must not be a fragmented IP datagram.
5. The destination port of the TCP or unicast UDP packet header must have its associated bit set in the Accelerated Port Bit Vector Table (APBVT). APBVT is described in Section 38.36.3.5.5.
6. The packet is then sent through a hash table lookup and must be found. The fields from the packet that are used for the lookup vary based on the packet contents. TCP/IP packets that do not have the SYN bit set or have the SYN bit set and ACK bit set use the combination of destination MAC address, VLAN tag if present, source and destination IP addresses and the source and destination port fields. TCP/IP packets with the SYN bit set and ACK bit clear or unicast UDP/IP packets use the destination MAC address, VLAN tag if present, destination IP address and destination port. See Section 38.36.3.5.7 for more details on the PE hash filter. If this check fails, a bit indicating that a hit in the APBVT was found, the IP table index from check number #5 and destination TCP port number are reported in the receive descriptor as status.

7. Finally, the packet must not have IPv4 options or IPv6 extension headers.

If any of the previous checks fail, the packet is processed by the LAN engine.

The PE has some limitations regarding the packet formats that are supported with RDMA and UDA traffic. The PE supports all packet formats that are configured using the internal switch and configured for the VSI. Additional headers (such as an extra inserted IP header, IP and NAT tunneling headers, or IP and Teredo headers) that are inserted for L2 traffic on a per packet basis are not supported for PE traffic. When the VSI is not configured for handling specific header formats, the PE is capable of handling packets with up to one extra packet headers to match the L2 VLAN capability. The VLAN tag specified in QP context matches the definition for L2TAG1 for LAN descriptors.

38.36.2 Verbs Programming Model

The verbs programming model is specified for iWARP from the RDMA Consortium(http://www.rdmaconsortium.org/home/draft-hilland-iwarp-verbs-v1.0-RDMAC.pdf). The 10 GbE controller supports the verbs constructs of QPs, Q (CQs), Shared-Receive Queues (S-RQs), and events. The 10 GbE controller supports verbs events by implementing CEQs and AEQs. Further description of each of these constructs in provided in the following sections. Additional information on the verbs programming model can be found in the link to the RDMA consortium website previously provided.

38.36.2.1 Verbs from a System View

The following sections provide an overview of each verbs construct supported by the 10 GbE controller from a system view. It is expected that the user is generally familiar with the verbs programming interface.

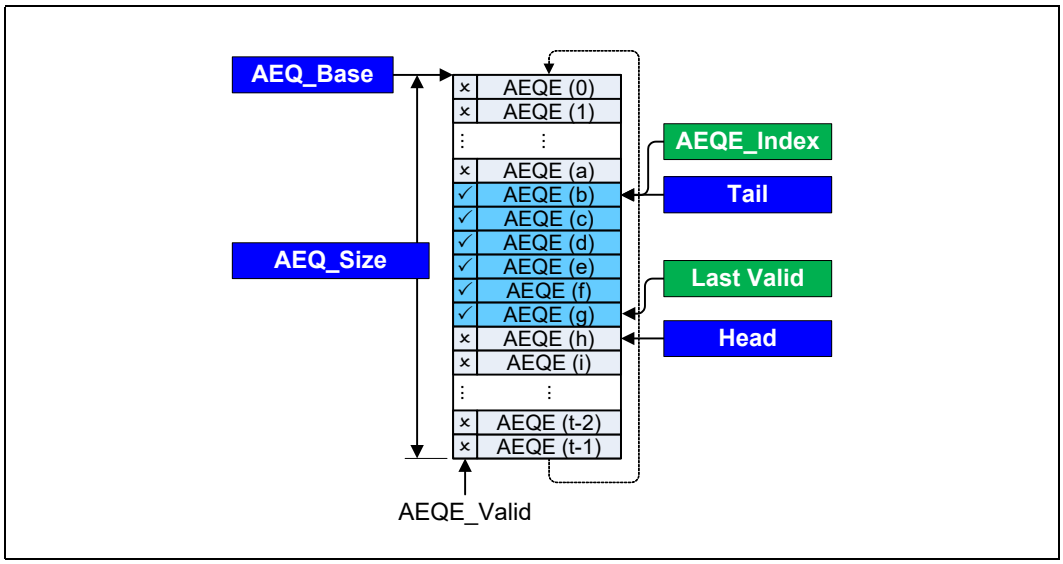
38.36.2.1.1 Asynchronous Event Queue (AEQ)

The 10 GbE controller supports a single AEQ per PE enabled PCI function. AEQs are used to report status and errors associated with PE QPs, CQs, and ARP table entries. AEQs are a packed array of AEQ entries (see Section 38.36.2.5 for the format of an AEQ entry) located in a virtually contiguous buffer in host memory (see Figure 38-107). AEQs should be sized to enable a single entry for every QP, CQ and ARP table entry that is active for the PCI function. Each QP, CQ and ARP table entry ensures that it generates only a single AEQ entry at a time. Software interaction is required for each resource to enable subsequent AEQ entries. If the AEQ is not sized appropriately, AEQ overflows can result [(Head+2)%AEQ_Size=Tail] in which case AEQ entries are lost and the AEQE_Overflow bit in the AEQE is set to notify software that the overflow condition occurred. Once the AEQ has overflowed, no new AEs are delivered to that queue. The AEQ must be discarded and then recreated to resume AE processing.

The initial condition for software shown in Figure 38-107 is AEQE_Index set to 0b. The last valid is shown strictly for discussion purposes. the 10 GbE controller first writes to the AEQE index specified by head. After an AEQE is written, an interrupt is generated if AEQ interrupts are not masked. Once an interrupt has been received that indicate that a new AEQ element (AEQE) is available, software reads the AEQE at AEQE_Index and increments AEQE_Index. Software processes all valid AEQEs until it encounters an invalid entry and stores the index of the invalid entry in the AEQE_Index variable. Subsequent AEQ entries might be generated by the 10 GbE controller after the entry that caused the interrupt while interrupts are masked. For each valid AEQ entry found, the PFPE_AEQALLOC register must be written to notify the 10 GbE controller that the AEQ entry is available for use by hardware. Writing the PFPE_AEQALLOC register causes the 10 GbE controller to increment the on-chip tail context variable. The PFPE_AEQALLOC register supports batching of AEQ entry acknowledgment into a single

write to enable software to minimize the number of register writes necessary to complete AEQ interrupt processing. PE enabled VF must use the VFPE_AEQALLOC registers instead of the PFPE_AEQALLOC.

Figure 38-107.AEQ



If software stops processing the AEQ before it has consumed all valid AEQEs, software must use a software initiated interrupt to return to processing AEQEs. Otherwise, no further interrupts are generated until a new AE is generated. Once the 10 GbE controller has run through the AEQ and wrapped back to AEQ0, the polarity of the AEQE_Valid bit is switched to avoid the need for software to go back and clear the AEQE_Valid bit for each AEQE processed.

38.36.2.1.2 Completion Event Queue (CEQ)

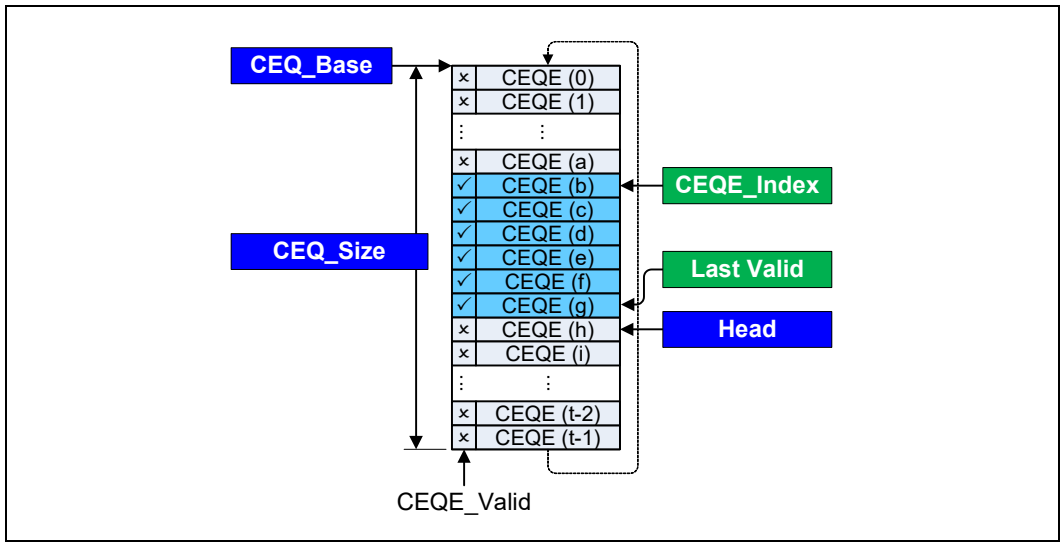
The 10 GbE controller supports one CEQ per MSI-X vector per PE enabled PCI function and a maximum of 256 CEQs total. HMC resource profiles are used to distribute the number of CEQs across the PE enabled PCI functions. Software can determine the number of CEQs and the specific instances assigned to a particular PCI function by reading the GLHMC_CEQPART[n] or GLHMC_VFCEQPART[n] registers after the HMC profile has been selected. Each CEQ is associated with a separate interrupt cause. It is expected that the number of CEQs that software uses is the minimum of the number of MSI-X vectors available, the number of CPU cores, and the value reported from the GLHMC_CEQPART[n].PMCEQSIZE. Software uses multiple CEQs to distribute the completion process workload across multiple CPUs. Each CQ is individually assigned a CEQ via the CreateCQ or ModifyCQ operations defined in Section 38.36.3.3.3.

Figure 38-108 shows that CEQs are a packed array of CEQ elements (see Section 38.36.2.4 for the CEQ element definition) located in a virtually contiguous buffer in host memory. CEQs should be sized according to the maximum number of active CQs that are assigned to the CEQ. Each CQ guarantees that it generates a maximum of one CEQ entry without having software acknowledge that the CEQ entry has been consumed. CEQs are not checked for overflow conditions so it is important that they are sized correctly or completion events are lost.

The initial conditions for software shown in Figure 38-108 is CEQE_Index is set to 0b. The last valid is shown strictly for discussion purposes. When the 10 GbE controller generates a new CE, a CEQE is written to the CEQ at index value of head with the CEQE_Valid bit indicating that a new event is available and an interrupt is generated.

The 10 GbE controller bumps head as part of the CEQE generation process and when the head reaches the end of the CEQ it wraps back to 0b. Software management of CEQE_Index must match the 10 GbE controller's head algorithm. Subsequent CEQEs might be written after the entry that caused the interrupt while CEQ interrupts are masked. Software is required to process all valid CEQEs up to the point where the first invalid entry is found. If software stops processing CEQEs before it has found an invalid entry, software must force the 10 GbE controller to generate a new interrupt using the SWINT_TRIG bit in the interrupt control registers. This is necessary since the 10 GbE controller does not track a tail value for CEQs and therefore cannot determine if a new interrupt is required to process CEQEs that have already been written and not processed by software. Once software has processed a valid CEQ entry, software writes the PFPE_CQACK register to enable the CE to generate new events.

Figure 38-108.CEQ



Once the 10 GbE controller has wrapped back to CEQE0, the polarity of the CEQE_Valid bit is switched to avoid the need for software to go back and clear the CEQE_Valid bit for each CEQE processed.

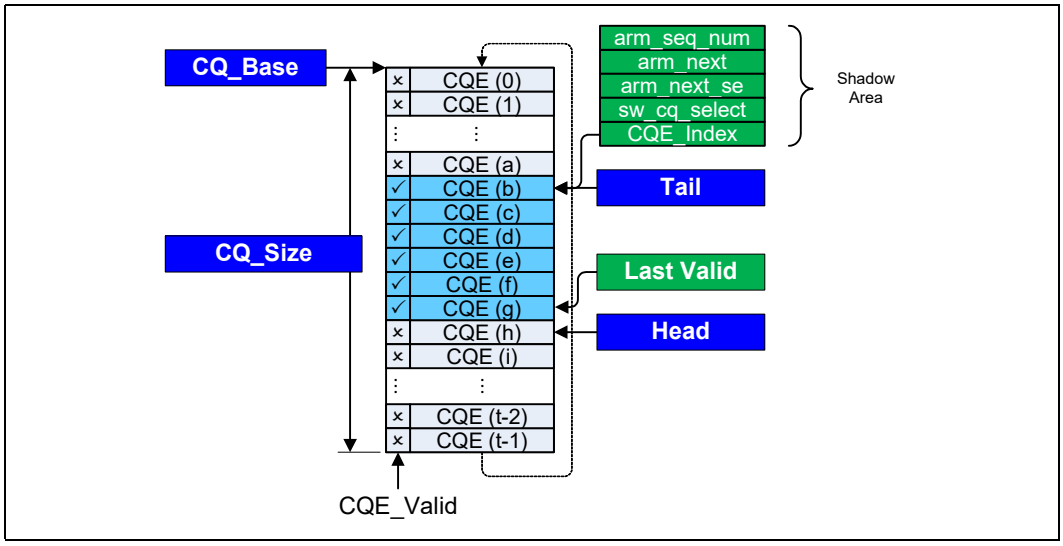
38.36.2.1.3 CE

When work requests submitted to Work Queues (WQs) complete, the 10 GbE controller might post CQEs to associated CQs. QPQP WQs might be SQs(SQs) or Receive Queues (RQs). SQs and RQs are associated with their CQs at QP creation time. This association remains (and cannot be changed) until the QP is destroyed. Each SQ and RQ (or Shared Receive Queue) can be bound to same or separate CQs. CQs can also be shared by multiple s from different QPs.

The 10 GbE controller supports up to 128 KB CQs that are distributed among that active PCI functions using HMC resource profiles. Refer to the section about the resource distribution mechanism used with CQs. The 10 GbE controller maintains the context of each completion queue in CQ context data structures in the HMC's FPM space. The storage elements of CQs reside in system memory. Refer to the section about the 10 GbE controller's usage of host memory for CQ context. As shown in Figure 38-109, CQs are organized as circular array of CQEs. Each CQE is 32 or 64 bytes in length and the format is dependent on the type of the WQ that is associated with the CQE. CQP, RDMA, and UDA WQs all can be related to CQs. CQs are managed using the Create/Modify/DestroyCQ operations defined in Section 38.36.3.3.3.

In addition to the CQ itself, software maintains a shadow area that the 10 GbE controller reads when the CQ is getting low on CQEs for hardware to write or when the CQ has been armed for event generation. The 10 GbE controller supports the verbs interface calls for requesting completion notification based on the next completion or based on the next completion that is associated with an operation that is a solicited event. The shadow area contains variables that must maintained by software in a atomic fashion since the 10 GbE controller could read that area at anytime. Specifically the two 32-bit words in the completion queue doorbell shadow area must be access using atomic 32-bit processor instructions. The arm_seq_num, arm_next and arm_next_se fields are ignored if an arm request has not been made by writing to the PFPE_CQARM register. The PFPE_CQARM register exists in CSR space for kernel mode drivers to access and also in the doorbell page area of the PCI function's BAR space that can be mapped directly to a user space application's memory space for kernel bypass operation. The format of the shadow area for CQs is listed in Table 38-465.

Figure 38-109.CQ



The CQ is organized as a circular queue, written by the 10 GbE controller and read by the driver software. Initially, both the 10 GbE controller and the driver software point to the first entry in the queue. The 10 GbE controller advances the Head context field after it writes a CQE to the CQ. Similarly, when the driver picks up a CQE, it advances the CQE_Index context field in shadow area in host memory. When head approaches tail, the 10 GbE controller reads the CQ shadow area and writes the CQE_Index value from the shadow area to the tail context variable. If head reaches tail-1 when the 10 GbE controller attempts to write a CQE, then an overflow condition is reported and the CQ is put into the error state. CQs generate a single asynchronous AE_CQ_OPERATION_ERROR event at this point and any further CQEs that are attempted to be generated by the 10 GbE controller are lost.

Completion events are generated if the CQ is armed for the appropriate event. The 10 GbE controller generates completion events under the following conditions assuming the tail does not equal head:

1. Any CQE is written (or has been written since the last event) and the CQ is armed for the next completion.
2. CQE is written (or has been written since the last event) that is associated with a solicited event operation and the CQ is armed for next solicited event.
3. CQ is armed on a recently re-sized CQ and software has not moved over to the new CQ yet based on the sw_cq_select value in the shadow area.

Completion events might be deferred if a completion event has been generated for a CQ and the PFPE_CQACK register has not been written by software.

The 10 GbE controller differs slightly from the verbs specification definition of generating completion events in two ways. First, the 10 GbE controller generates a completion event for a CQ that is armed for next completion without waiting for a new completion to be generated if it appears that CQEs have not been processed by software (Head != tail after reading the doorbell shadow area). Secondly, the 10 GbE controller does not track the exact location of the solicited events that have been generated since the last completion event. The 10 GbE controller generates an completion event for a solicited event operation if any solicited event completion has been generated since the last completion event was generated and it appears that CQEs have not been processed by software.

The 10 GbE controller process for arming CQs for event generation is simply to first write to the appropriate bit in the CQ shadow area to enable either next or next solicited completion notification events, increment arm_seq_num, and then write to the PFPE_CQARM register. The 10 GbE controller then reads the shadow area and the CQ context is used to either immediately generate a new completion event if the CQ has unprocessed CQEs remaining or arm the CQ to generate a new event once a subsequent CQE is written. As previously described, completion events can be deferred under certain circumstances. The 10 GbE controller maintains a copy of the last arm_seq_num value that was read during the last arm request in CQ context. The 10 GbE controller compares the value of arm_seq_num in CQ shadow area with the value in CQ context during arm requests and drops arm requests that have the same value in the shadow area and in CQ context. This comparison prevents CQ arm requests from rogue applications from changing the arm state of a CQ unless the application also has access to the CQ shadow area.

CQ resize operations with the 10 GbE controller involve four steps. The first is to allocate a new CQ in host memory based on the new size requested by an application. The second is to issue a ModifyCQ operation to the 10 GbE controller. ModifyCQ operations notify the 10 GbE controller to start using the new CQ for new CQEs. The third step is to completely process CQEs from the old CQ. The final step is to start to process CQEs from the new CQ after freeing the buffer for the old CQ. The old CQ can be considered to be completely processed when an invalid CQE has been found in the old CQ and at least a single valid CQE has been encountered on the new CQ. When the transition to updating CQE_index based on the new CQ occurs, sw_cq_select must be incremented. During the ModifyCQ operation, the 10 GbE controller incremented its cq_select value and switched over to writing CQEs to the new CQ. Since there could be pending CQEs still on the old CQ, the 10 GbE controller compares the CQ shadow area sw_cq_select to the hardware cq_select value during any reads of the CQ shadow area due to arms or CQ tail updates. If hardware cq_select does not match sw_cq_select, the 10 GbE controller ignores CQE_index and does not update CQ tail, assuming that software is still working with the old CQ. Arm requests generate new events immediately since the 10 GbE controller is no longer aware of the state of the old CQ. Once software has properly set CQE_index to reflect progress on the new CQ it must also increment sw_cq_select so the 10 GbE controller starts processing arm requests and CQ tail updates based on the new CQ.

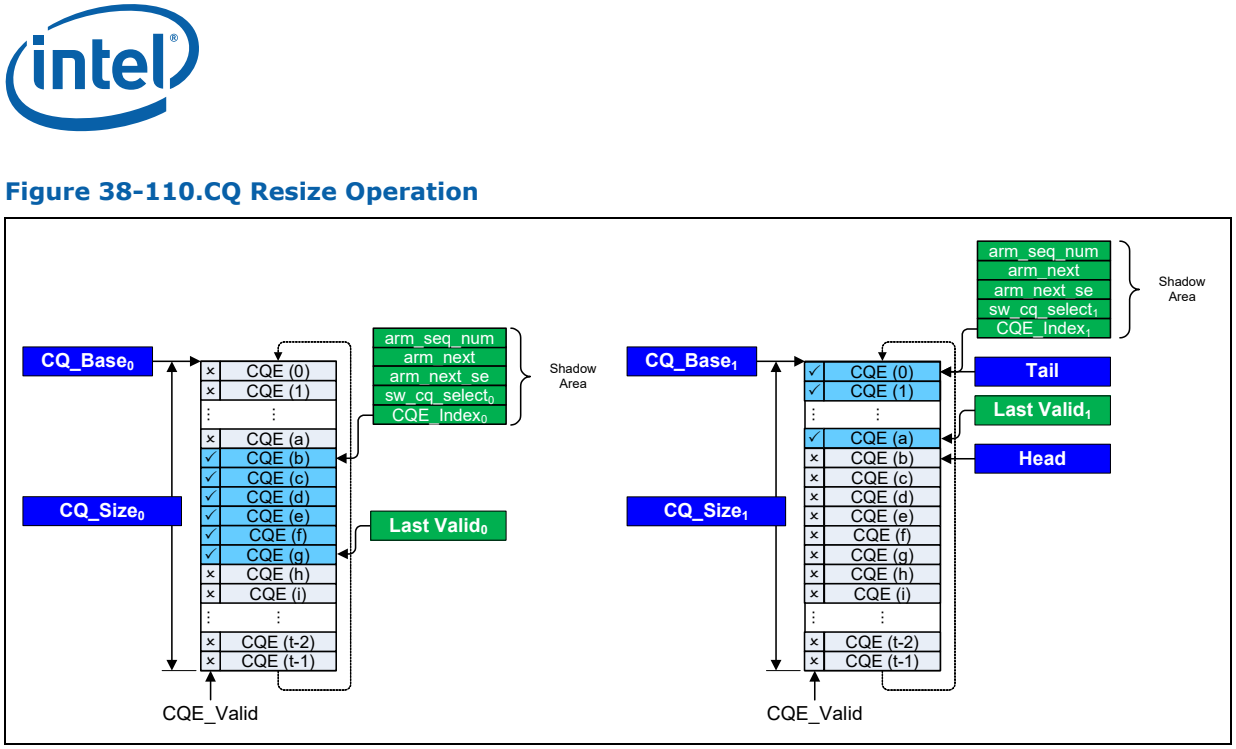


Figure 38-110.CQ Resize Operation

Figure 38-110 shows a CQ during a resize operation. The original CQ is shown on the left side of the diagram. At the time that the ModifyCQ operation occurred for the resize, the CQ had valid CQEs that had not been processed by software. The 10 GbE controller switched over to the new CQ shown on the right side of the diagram at the time of the ModifyCQ and incremented its cq_select value. New CQEs are shown as generated on the new CQ in Figure 38-110. Any arm requests immediately generate new CEs while sw_cq_select does no match the hardware cq_select value. If the doorbell region is read during this time, the CQE_Index$_0$ value is ignored because sw_cq_select0 does not match cq_select. Once software has found an invalid entry in the original CQ and CQE0 from the new CQ is valid, software can reset CQE_Index to 0, increment sw_cq_select, and start processing CQEs on the new CQ. At this time it is safe to free the memory used for the original CQ since the 10 GbE controller no longer writes to the old CQ. While two CQ shadow areas are shown in Figure 38-110, there is really only one. Two are shown to indicated that values of sw_cq_select and CQE_index have changed to signal the change over to the new CQ.

38.36.2.1.4 Memory Registration (Translation/protection)

In today's modern operating systems, applications running at user protection level use virtual addresses and they are not aware of their data buffers' physical addresses. However, traditional I/O devices require physical addresses to transfer data to/from system memory. This gap is typically bridged by the software device driver, which runs in the kernel protection level. Transitioning from user to kernel protection level, and than back to user protection level is not only quite expensive (in terms of MIPs) but also contributes to increased latency. The iWARP architecture solves this inefficiency by:

- Using virtual data buffer addresses in the RNIC programming model.
- Servicing latency critical functions through libraries running at the user protection level.

This approach requires RNICs to implement robust address translation and protection techniques. This section provides in depth descriptions of the address translation and protection mechanism used by the 10 GbE controller. However, designers are assumed to be familiar with the memory management concepts, operations and related terms described by iWARP Verbs specification.

The 10 GbE controller uses a Memory Region Table (MRT) and Physical Buffer Lists (PBLs) to represent virtually contiguous system memory locations that are accessible for iWARP operations. The data structures in MRT are used to describe these memory locations and their protection attributes. PBLs provide the page lists that back each MRT entry. These concepts, supporting data structures and related operations are described in Section , "Address Translation and Protection Overview". In these sections, the actual MRT and PBL construction details through the HMC are intentionally omitted and a virtually contiguous HMC FPM space is assumed.

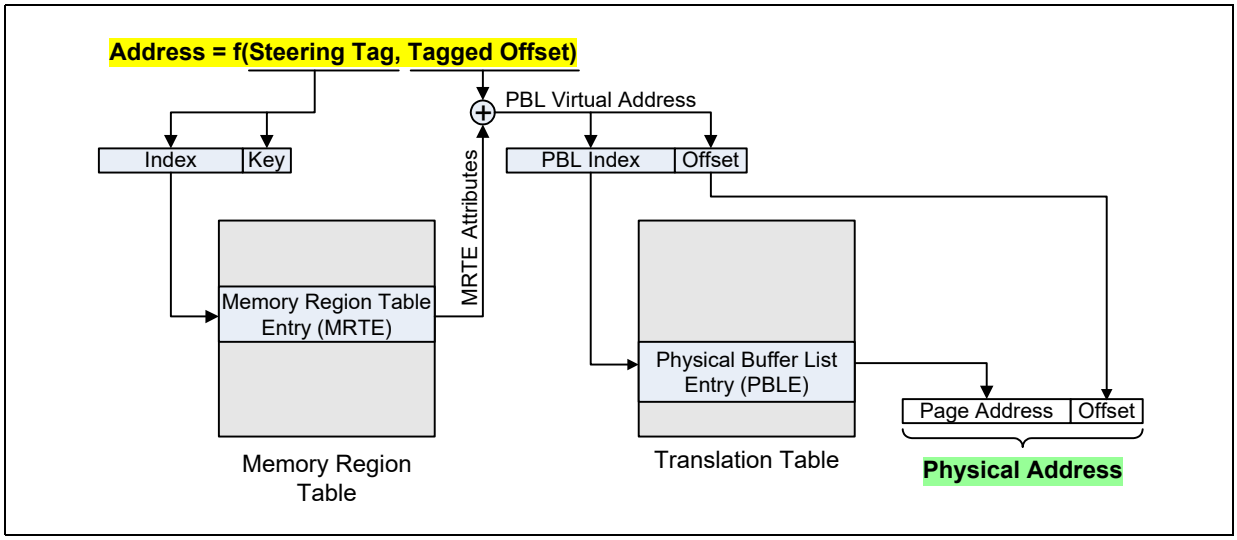
Address Translation and Protection Overview

As shown in Figure 38-111, the 10 GbE controller supports a two level protection and translation flow. The STag and tagged offset pair define the protection attributes and physical address of the system memory being accessed.

The STag portion of the address is used to locate the Memory Region Table Entry (MRTE) in the MRT. This data structure provides the protection attributes (Type, Access Control, Protection Domain and Key), the bounds check attributes (Base Tagged Offset, Length and First Byte Offset), and the address translation attributes (Page Size, and Base PBL Index). These attributes are described in detail later in this section.

After the protection attributes are checked and the address bounds checks are performed, the address translation attributes, along with tagged offset are used to calculate the PBL virtual address. The translation flow is explained in detail later in this section. The PBL Index portion of the PBL virtual address is used to locate the Physical Buffer List Entry (PBLE), which provides the physical page address. Finally, the page address is concatenated with the offset portion of the PBL virtual address to form the physical system address.

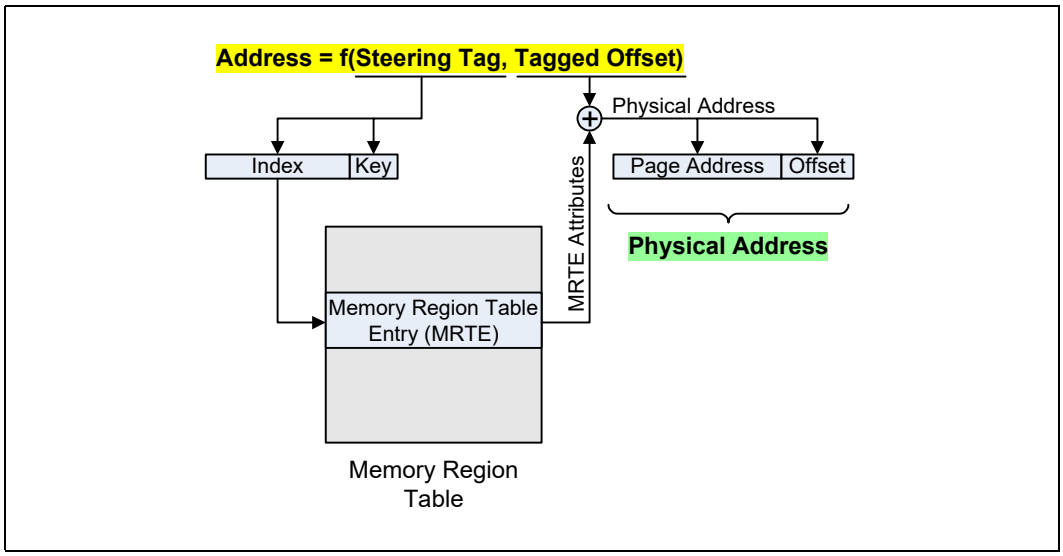
Figure 38-111.STag Decomposition to Physical Address



Depending on its size, each MRTE might be associated with multiple PBLEs. In this case, the PBLEs must be allocated consecutively to form the PBL. The PBLs can also be shared between multiple MRTEs when memory windows or shared memory regions are used. This enables efficient representation of a section of the system memory that is registered multiple times; potentially with different protection attributes.

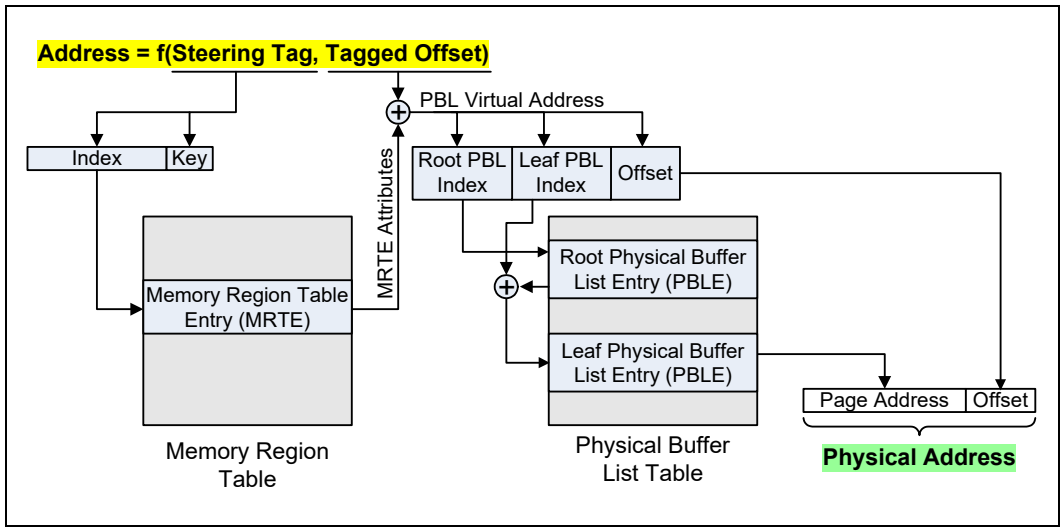
The 10 GbE controller also supports two other modes of address translation. The first is termed direct page translation that can be used when the STag is backed by a physically contiguous buffer is depicted in Figure 38-112. In this case the PBL translation step is skipped, which enables more efficient calculation of the physical address accessed for a given STag.

Figure 38-112.Direct Page Translation



The second additional mode of address translation is designed to enable software to make more efficient use of PBL address space when large numbers of large buffers are used for an application at the expense of additional accesses to host memory in order to calculate the ultimate physical address. Figure 38-113 shows a mode where PBLs are accessed twice by breaking the PBL virtual address into three pieces instead of two. The root PBLE is first accessed in order to find the PBLE associated with the leaf PBL. Once the leaf PBL is identified, the leaf PBL index is used to identify the leaf PBLE address that contains the page address portion of the physical address. In this mode, the leaf PBLs must be either 256 (64 root PBLEs or 32 leaf PBLEs) or 4096 (1024 root PBLEs or 512 leaf PBLEs) bytes in size. The root PBLs can be any size. This mode enables software to slice up the PBL space into fixed size PBLs for large memory regions of up to 2 Gb/s instead of having to reserve large contiguous regions of PBL space for a single STag.

Figure 38-113.Two-Level Page Translation



Each root PBLE is just a 28-bit object index of the starting PBLE object for the associated leaf PBLE. In other words, each root PBLE points to a another PBLE HMC object that is the start of the leaf PBL. Each leaf PBL is an array of 32 or 512 PBLEs. The size of the root PBL is determined from the region size specified in the MRTE. Each leaf PBLE is a 64-bit pointer to a host memory page that can be a 4 KB or a 2 MB page. For 4 KB pages the lower 12 bits of the host memory page pointer are ignored and for 2 MB pages the lower 21 bits are ignored.

Following is a brief overview of commonly used terms that are provided in the sections that follow.

STag

All local and remote memory accesses require use of an STag. The STag, along with a tagged offset is used to identify a memory location within a specific memory region or memory window. STags are found in RQ and SQ WQEs and in tagged iWARP messages received from the wire. As shown in Figure 38-114, 32-bit wide STag is further divided into two fields.

Figure 38-114.STag Format

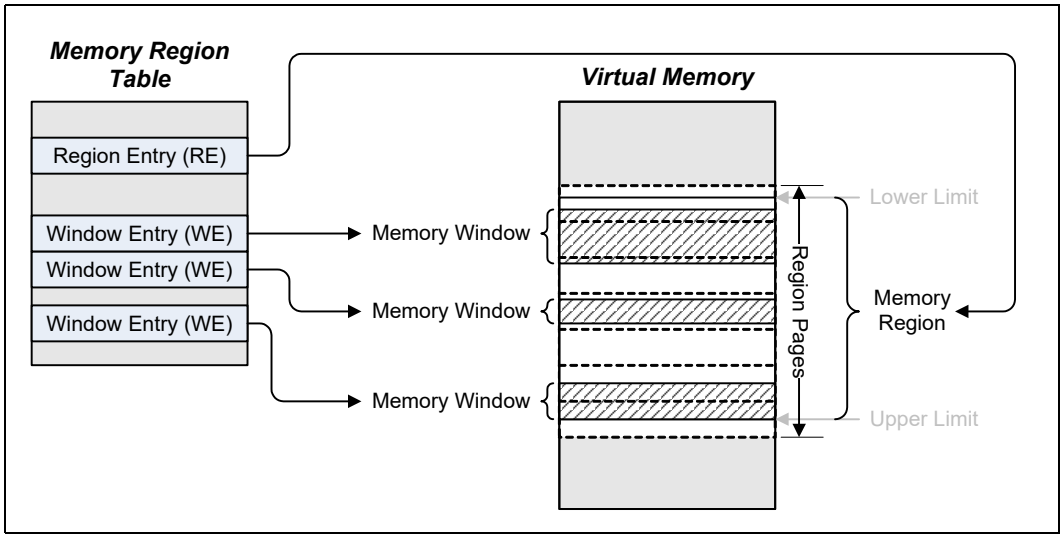
- STag index — The most significant 24-bits of STag is called the STag index. Since this field is used as an index into the MRT, it is also known as the memory region table index or MRT index in short. The 10 GbE controller supports up to 4 MB protection entries. Note that the unused most significant bits of the STag index can be randomized by software to provided reduced predictability for any MRT attacks.
- STag key — The least significant 8-bits of the STag is called the STag key. The STag *Key* field is a user or driver provided key that provides an additional level of security for the STag protection check.

Memory Region and Memory Window

Applications must register memory regions prior to accessing the system memory either locally or remotely. Memory regions, and their translation and protection attributes are represented by properly formatted MRT entries. MRT entries of this type are also called region entries or RE in short. Memory region registration and de-registration requests are communicated to the 10 GbE controller by:

1. Using CQP commands (like: register memory region and de-register memory region).
2. Working a request through a SQ (like: fast register non-shared memory region and invalidate STag).

Figure 38-115.Memory Region to Memory Window Relationship



The Verbs specification also defines the memory windows concept. Memory windows can be placed anywhere within a valid memory region. Multiple memory windows are allowed within a memory region. Memory windows are allowed to overlap, or completely included within another memory window. However, memory windows are not allowed to cross memory region boundaries. The 10 GbE controller supports memory windows through another type of protection entry. These protection entries are also referred as Window Entry (WE). Figure 38-115 shows the relations between memory regions and memory windows.

Memory windows can only be bound to memory regions that can be binded. A memory window's protection attributes are constrained by the protection attributes of the memory region. The 10 GbE controller supports byte-level lower and upper address limits for both memory regions and memory windows.

Tagged Offset (TO)

The 64-bit wide *TO* field specifies the first byte of the buffer being addressed in a memory region or memory window. Memory regions and memory windows have a base attribute as either zero-based TO or Virtual Address (VA) based TO. For a VA based TO, the TO of the first memory location associated with the memory region equals the base virtual address value specified when the memory region is registered. For a zero-based TO, the TO of the first memory location associated with the memory region equals zero.

Page Size

Each memory region and memory window has a page size attribute. Memory windows inherit the page size attribute of their associated memory regions. The 10 GbE controller supports two different page sizes: 4 KB and 2 MB. All system memory pages that are part of a memory region or memory window must be of the same size.

38.36.2.1.5 QP

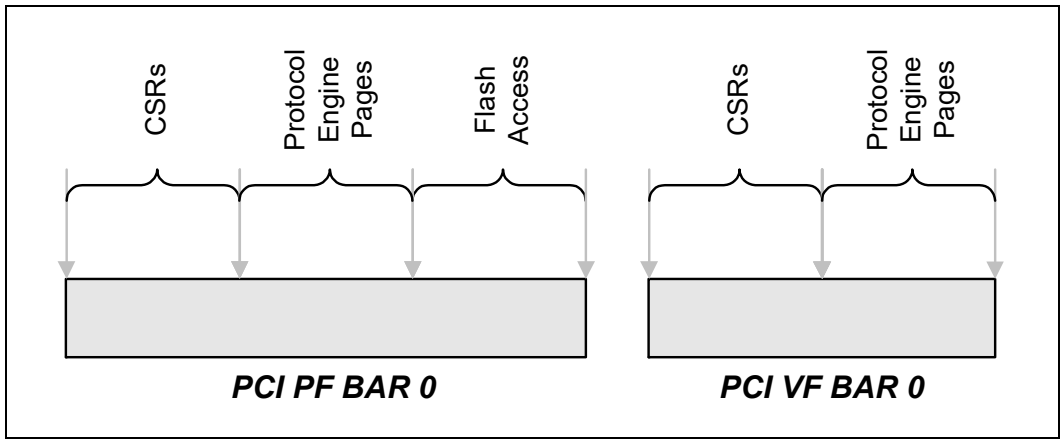
The PE in the 10 GbE controller supports the verbs QP. QPs are made up of a pair of WQs.

QP WQs are either SQs or RQs. SQs and RQs are associated with their CQs at QP creation time. The 10 GbE controller supports QPs for PE administrative commands (control QP or CQP), RDMA supported by iWARP, and UDA. Each QP type has unique operation types that are described in Section 38.36.3.3, Section 38.36.4.5, and Section 38.36.5.8, respectively. Note that CQP does not implement a RQ. The 10 GbE controller supports a maximum of 256 KB QPs that are distributed among that active PCI functions using HMC resource profiles. Refer to the section describing the resource distribution mechanism used with QPs. The 10 GbE controller maintains the context of each QP in QP Context data structures in the HMC's FPM space. The storage elements of QP context reside in system memory. Refer to the section about the 10 GbE controller's usage of host memory for QP context.

38.36.2.1.6 Doorbell Pages

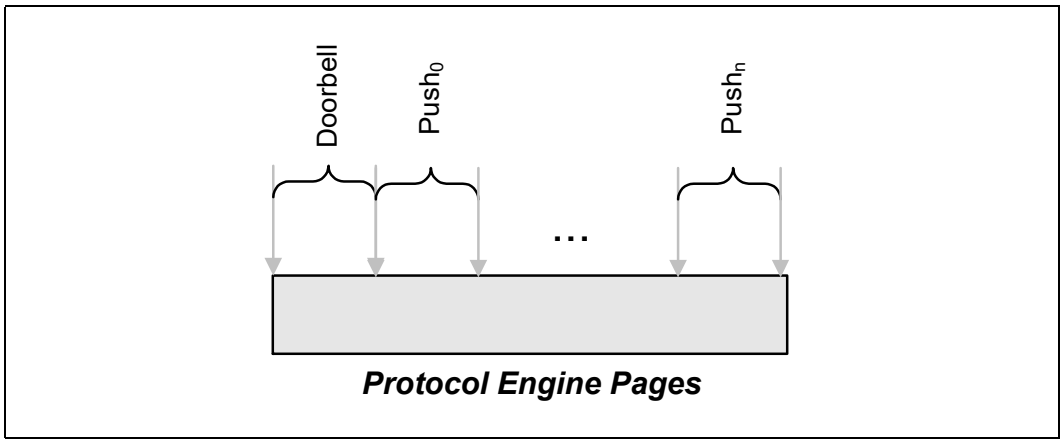
In order to support direct access to the 10 GbE controller hardware for PE functionality, the 10 GbE controller implements adapter memory that can be mapped to a user-space process. The memory is exposed to the system through the PCI BAR registers in a similar manner to the CSRs. Each PCI function optionally has a number of PE pages as shown in Figure 38-116.

Figure 38-116.PE Page Location in BARs



As shown in Figure 38-117, the first PE page is used for submitting work to PE SQs and also for arming CQs from user space. The remaining pages are used to support low latency push mode operation.

Figure 38-117.PE Pages



The format of the doorbell portion of the PE pages is listed in Table 38-443. The details of each register found in the PE pages are documented in the programming interface section.

Table 38-443.Doorbell Page Register Summary

Offset / Alias Offset	Abbreviation	Name	Block	RW
PE Registers				
0x00000000	PFPE_WQEALLOC	PE WQE Allocate Register	PE	RW
0x00000040	PFPE_CQARM	PE CQ Arm	PE	RW

38.36.2.1.7 SQ

As shown in Figure 38-118, SQs are organized as circular array of SQ Elements (SQEs). The host memory used for SQs might be physically contiguous or discontiguous. Each SQE is 32 bytes in size and can have up to 6 additional fragments for a total of 7. This approach enables Work Requests (WR) to consume a variable amount of SQ space that enables software to advertise 7 fragments as the maximum supported but opportunistically pack multiple WRs into a smaller space if the application needed fewer fragments for a given WR. The 10 GbE controller takes advantage of variable size WRs by reading 128 bytes at a time from the SQ when processing work. If a WR used all 7 fragments, it would consume the 128 bytes and would transmitted by itself. If instead the application happened to be using 4 consecutive single fragments WRs, all 4 WRs would be fetched (assuming software posted the WRs before the 10 GbE controller managed to read the first WR) with a single read of the SQ which boosts bus efficiency. The minimum size of a SQ is four maximum sized WQEs for proper operations. For example, this means that if a WR can ever use 7 fragments, then the SQ must be 512 bytes. An additional optimization (iWARP only) that the 10 GbE controller provides is that WRs that are small enough to fit into a single Ethernet packet and fetched as part of the same read of the SQ are sent as a single Ethernet packet for efficiency on the Ethernet fabric.

Figure 38-118.PE SQ

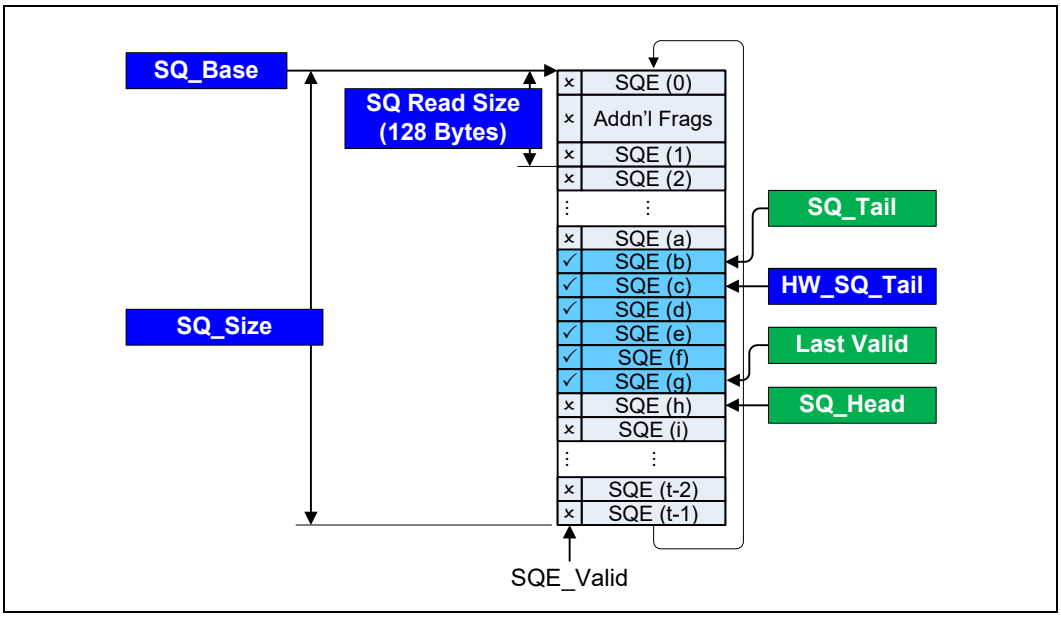


Figure 38-118 shows a typical SQ where software is tracking SQ_head and SQ_Tail and the 10 GbE controller tracks HW_SQ_tail in QP context. Software must size the SQ large enough to support every SQE with the maximum number of additional fragments. The amount of space required per work request is listed in Table 38-444. Essentially the WR space requirements on the SQ end up rounding up to the next multiple of 32 bytes. SQE 0 is shown using four additional fragments and there for takes up 96 bytes. Every SQE must start on an offset that is a multiple of 32 bytes within the SQ and an SQE and all of related additional fragments must fit into a single 128 byte SQ read block. NOP SQEs can be added if necessary to pad SQEs out when necessary to satisfy this requirement. The 32-byte boundary requirements for SQEs means that there is no difference in SQ size between a WR with 4 fragments and a WR with 5 fragments. If SQE 1 required any additional fragments, a single NOP SQE would need to be added in place of SQE 1 and all subsequent SQEs would move down in the SQ so that the SQE 1 and all related additional fragments would fit in the same 128 byte SQ read block.

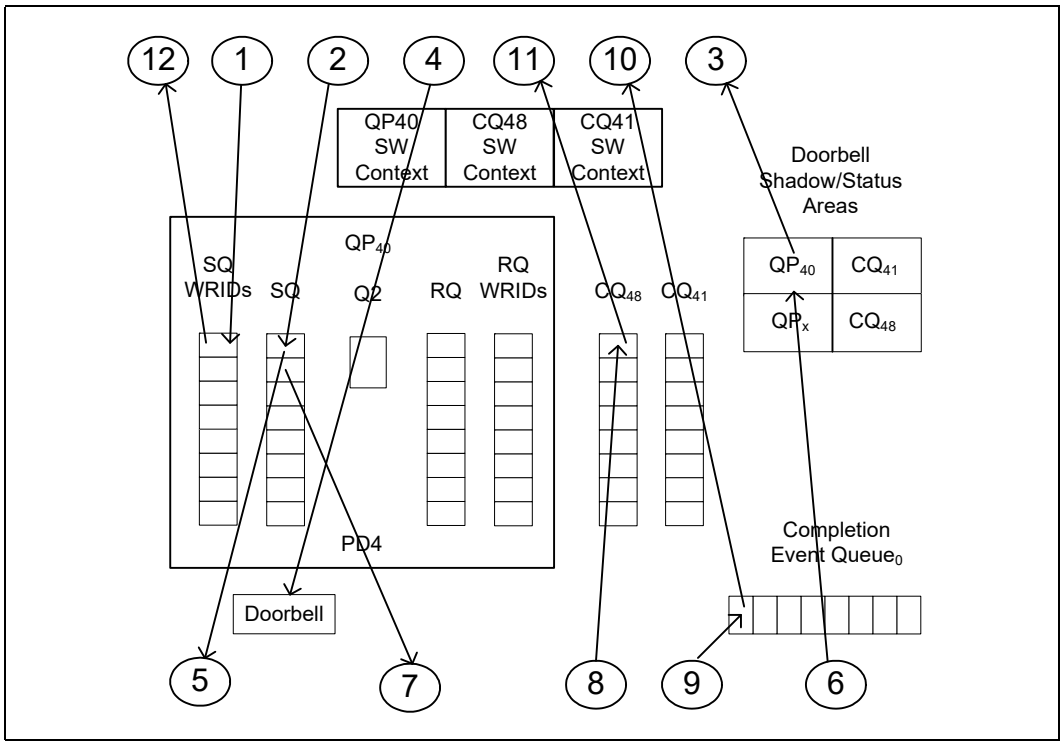
Table 38-444.QP WR Size Requirements

Number of Fragments Requested at QP Creation	Size Required per Work Request
1	32 bytes (one SQE)
2-3	64 bytes (one SQE + two additional fragments)
4-5 (SQ only)	96 bytes (one SQE + four additional fragments)
6-7	128 bytes (one SQE + six additional fragments)

Figure 38-119 shows a system view of how the 10 GbE controller processes PE QP SQs. Before describing the flow in Figure 38-119, software must perform a fair amount of initialization. Software starts by populating all QP related HMC objects necessary for CQs, QPs as well as related memory regions. Next software allocates buffers for software QP and CQ context, SQ and RQ WRID tracking arrays (software context and WRID tracking arrays do not need to be pinned, QP40, CQ48, and CQ41 in Figure 38-119), QP and CQ doorbell shadow/status areas, Q2 areas, and finally the SQ, RQ, and CQ buffers themselves. The doorbell shadow/status areas, Q2 and WQs all

need to be pinned and if they are physically dis-contiguous, the page list for each of the pinned buffers must be collected. CQP operations are used to create the CQs (if they do not already exist) and QPs.

Figure 38-119.PE Operation: SQ



Now that all structures have been created and defined to the 10 GbE controller, the steps shown in Figure 38-119 to process a post-send verbs request are the following:

1. Software stores the WRID supplied by the application on the post send request to the SQ WRID.
2. Software creates a SQE along with any additional fragments necessary to represent the WR.
   a. The valid bit in the SQE must only be set once all fields are valid since the 10 GbE controller can read SQEs at any time. Note that the valid bit is a generational valid bit, which means that the first (and all subsequent odd) iteration through the SQ, 1 means valid on all even iterations through the SQ, 0 indicates that the SQE is valid.

- Software reads HW_SQ_Tail from the doorbell status area for the QP.
- Software then compares the SQE_Head with HW_SQ_Tail.
  a. if the two are equal, the PFPE_WQEALLOC register is written to notify the 10 GbE controller that SQ work is available.
  b. SQE_Head is incremented by the number of SQEs consumed work requests (WR size in bytes/32 bytes).

3. The 10 GbE controller reads the 128-byte SQ read block that contains the SQE indicated by QP context value of HW_SQ_Tail.
   a. The 10 GbE controller processes all valid SQEs contained in the 128 byte SQ read block. In this example it's assumed that only SQE 0 is valid, if no SQEs are valid, proceed to step 6.

b. The 10 GbE controller processes SQE 0,which likely includes fetching data from host memory, sending packets through the Ethernet port, and waiting for TCP acknowledgments.

c. The 10 GbE controller increments the HW_SQ_Tail in the doorbell shadow/status area.

d. If the last SQE in the 128-byte SQ read block is valid, step 5 is repeated, which reads the next 128-byte SQ read bock.

4. The 10 GbE controller writes the doorbell status area for the QP with the current value of HW_SQ_Tail.
5. The 10 GbE controller reads the 128-byte SQ read block one more time to ensure that software has not posted new work.

   a. If a new valid SQE is found, proceed back to step 5a.

6. The 10 GbE controller optionally writes to the CQ associated with the SQ.
7. The 10 GbE controller optionally writes to the CEQ if the CQ was written and generates an interrupt.
8. Software then fields the interrupt if one was generated and reads the CEQ to determine the CQ that generated the event (an application might be polling the CQ directly in which case steps 9 and 10 are skipped). Typically the CEQE contains the most significant 63 bits of the virtual address of CQ context to make CE processing efficient.
9. Software then reads the CQ to determine the QP and WQ that generated the work completion, the SQE index of the WR is reported in the CQE along with a 64-bit pointer that is typically set to the software QP context address at QP creation time.
10. Software then reads the SQ WRID array for the SQE index from the completion, the resulting WRID is returned to the application as an indication that the original work request completed.

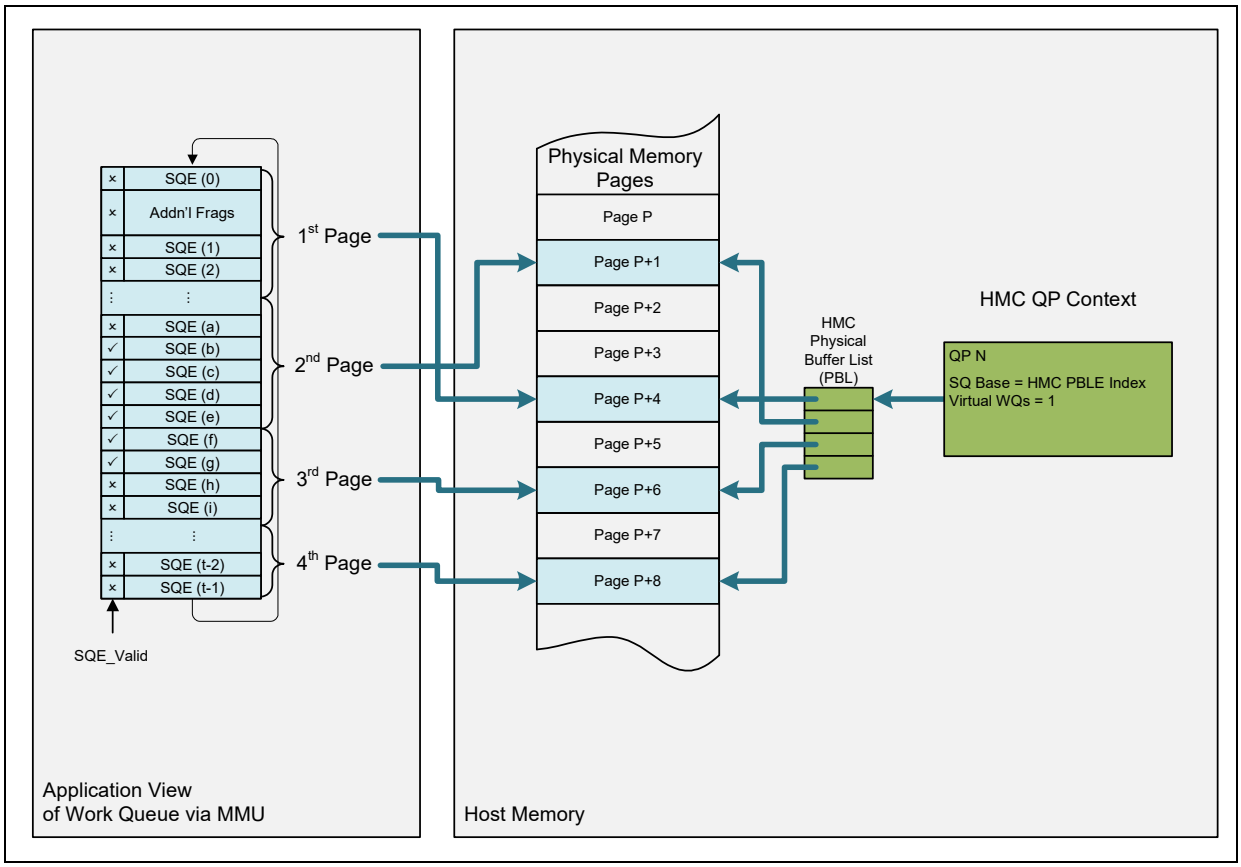
Note: While steps 6 and 8 are shown as sequential, the 10 GbE controller starts both steps simultaneously. Also step 1 can be initiated by software at any time as well.

Virtual Queues

The 10 GbE controller supports physically contiguous and physically dis-contiguous WQs for PE QPs. CQs, and CEQs, and shared receive queues also support both physically contiguous and dis-contiguous buffers. In the case of QPs, the support for physically dis-contiguous WQs is referred to a virtual WQ support. This section describes an example of a SQ.

Figure 38-120 shows a virtually contiguous but physically dis-contiguous SQ. The left side of the figure shows the applications view of the SQ. This particular SQ is composed of four host memory physical pages that are not contiguous. In this scenario, the virtual buffer is pinned and the 1 level page list is retrieved from the operating system. Then an unused page list must be allocated from the HMC PBLE object space with 4 contiguous PBLEs. This PBL is now populated with the page list retrieved from the operating system. When the QP is created, the Virtual_WQs bit is set in CQP Create QP WQE and the SQ Base in QP context is set to the starting PBLE HMC object index instead of a host physical address of the SQ. The RQ for a QP with the Virtual_WQs bit set must also use a PBL even if it happens to be physically contiguous. Note that the SQ and RQ in this case must be allocated a host page boundary.

Figure 38-120.PE Operation: Virtual SQ



SQ Push Mode

Push mode reduces processing latency of the short message posted for transmission on the low-latency QP by eliminating a read of WQE or WQE with inline data from the host memory. This is achieved by software writing WQEs to the adapter memory mapped address space using processor write-combining buffers. Processor write-combined buffers allow combine writes and make transactions on the bus more efficient. Software is not expected to use this mode to stream high rate of the short messages or use this mode to post a large messages. The size of the message posted using push mode should not exceed a configured scheduling Quanta. This feature can be either exposed to the application and have an application responsible for the potential QP bandwidth degradation caused by overuse of this feature, or a verb layer can be instrumented to use information provided by hardware and opportunistically take advantage of push mode messages for the given connection based on true low-latency messaging requirements.

In modern systems, a memory type can be specified within an operating system page structure using an index in the PAT register. This way an individual page of adapter memory can be mapped to the application address space as a write-combined memory. Writes to such page would be combined in the processor write combining buffers. These buffers are flushed on PCI bus in cache lines. In some systems, the same page can be mapped as an uncached memory page that enables software to use uncached write (or doorbell ring) to flush push mode WQEs from processor write-combining buffers to the PCIe bus. Adapter memory mapped pages used to for push mode WQEs are called push pages. The number of push pages exposed to the driver depends on the chip configuration and PCIe function. In most systems, it is not possible to map the same

page as both uncached (non-write-combined) and write-combined. In this case the GLPE_PSHCFG.PSHCFG_DB_SPLIT must be set, which forces the doorbell pages to be treated separately from the page used to push the WQEs and data. When GLPE_PSHCFG.PSHCFG_DB_SPLIT is set, two push pages are mapped to the application address space where the odd number push page is mapped as uncached (used as the push doorbell page) and the even numbered push page is mapped as write-combined. This reduces the amount of push pages available to applications by half since a single application requires two separate push pages for proper operation.

Combining writes using dedicated write-combining buffers is a very old feature of the processors. This feature was mainly used for the ancient graphic adapters. Now days all graphics adapters use their own DMA engines to pull the data from the host memory, rather than having it pushed with CPU.

Processors specify a write combining memory region using several mechanisms.

- The MTRR register specifies a memory type per region. Granularity of memory region sizes does not allow the use of MTRR for wide deployment of write-combining for the 10 GbE controller.
- Page Attribute Table (PAT) in conjunction with flags in Page Table Entry (PTE) enables write-combining on a per-page base, which makes it feasible for the 10 GbE controller deployment. Relatively recent distribution of Linux added an interface allowing change page attributes to write-combined page.

Processors have a set of buffers for write-combined accesses. Each buffer has a size of several cache lines. Writes to write-combined memory are intended to be delayed to enable better memory access efficiency and reduced overhead.

Write-combined buffers are evicted or flushed by serialization events, such as SFENCE, LOCK, interrupt, etc. and read/write to uncached memory. Partial write-combined buffer eviction can be a result of the serialization event. Eviction of a full write-combined buffer is implementation dependent. Intel assumes that serialization events are evicting/flushing both full and partially filled write-combined buffers. Intention is to use door bell ring (write to uncached memory) as a serialization event to evict write combined buffers.

All types of the PE SQ WQEs can posted using a push mechanism. The maximum size of data that can be pushed together with WQE as an inline data is limited to the 112-byte matching the maximum size of data that can be posted using WQE with inline data. Larger messages can be pushed using regular WQE, eliminating a need to read a WQE from the host memory. If an application requested post send, verb layer can decide whether to push the message to adapter based on the message size, configurable threshold and outstanding WQEs previously posted to SQ. In any case, verb layer should always first post WQE to the SQ (without ringing the door bell) and only then write WQE push mode page, followed by uncached write to the associated push mode page. Posting WQE to the SQ required for the proper re-transmission process, and enable hardware opportunistically discard push mode messages and transmit those using regular SQ processing mechanism.

To avoid security implications, a push mode page cannot be shared by different processes. If it was allowed to be shared, then ProcessA could guess the QP number of ProcessB and other QP attributes, like the push mode page, and make ProcessB send a message pushed by ProcessA as its own. This can lead to severe security violation. Multiple threads residing in the same process share a process virtual space and belong to the same security domain. The number of push pages for VFs is limited to 15 to reduce the amount of BAR space that the operating system is required to allocate for the 10 GbE controller. In order to overcome this restriction, a para-virtualized driver is allowed to map PF push pages to the VF for use.

Push mode is an opportunistic latency optimization that supposed to be used on lightly loaded QPs. If adapter runs out of resources or software attempts to use push mode WQEs too aggressively, adapter can discard a push mode WQE, and transmit it later as a regular WQE with inline data posted to SQ.

Software is allowed to post any SQ message using push mechanism. The size of the message should not exceed a configurable Quanta value (default of 4 KB). Messages specified as inline operations by application can be pushed together with data. A total of 112 bytes of the message with inline data can be pushed to the 10 GbE controller to avoid overhead of the SQE and data fetches. Larger messages would avoid a fetch of the SQE element, but still requires a data fetch. Software can post multiple messages with single push operation, as long as all messages are continuously located and do not cross a boundary of the push page SQE element, shown on Figure 38-121 or Figure 38-122. Push mode does not require any change in WQE format, except for setting a push bit in the SQ WQE. A process is enabled for push mode by allocating a push page shown in Figure 38-121 or a pair of push pages as shown in Figure 38-122 (see Section 38.36.3.3.10 for the CQP WQE format) from the 10 GbE controller's BAR to a QP and including the push page Index in QP context when the QP is created. Each PCI function has a number of push mode pages that must be assigned. See Figure 38-116 and Figure 38-117 for more details on how the 10 GbE controller exposes push pages to the system. The 10 GbE controller validates that the push operation is associated with the correct push page and push SQE index as well as PCI function before acting on the operation.

Figure 38-121.Combined Push Page Detail

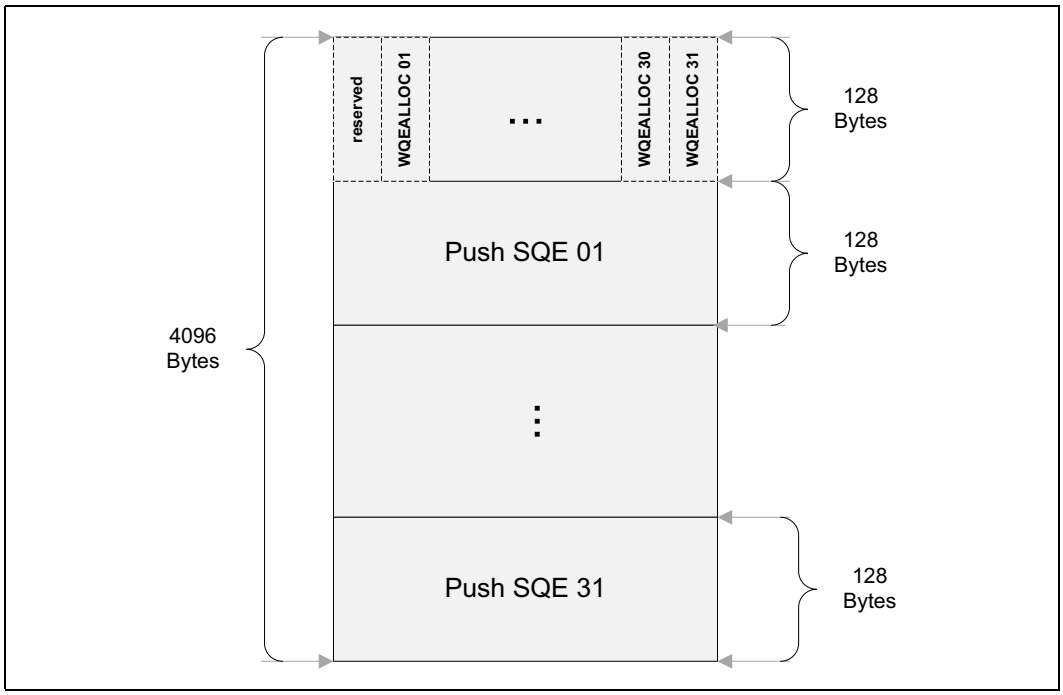
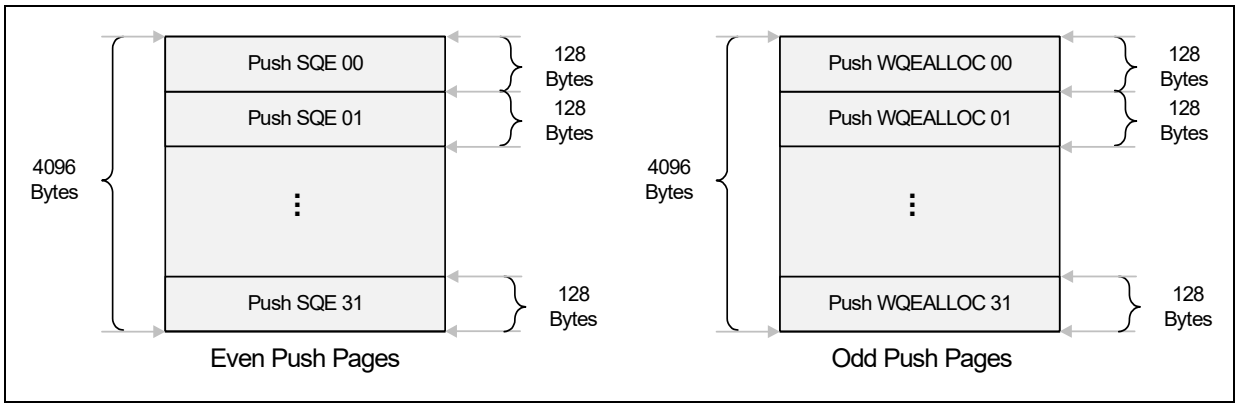


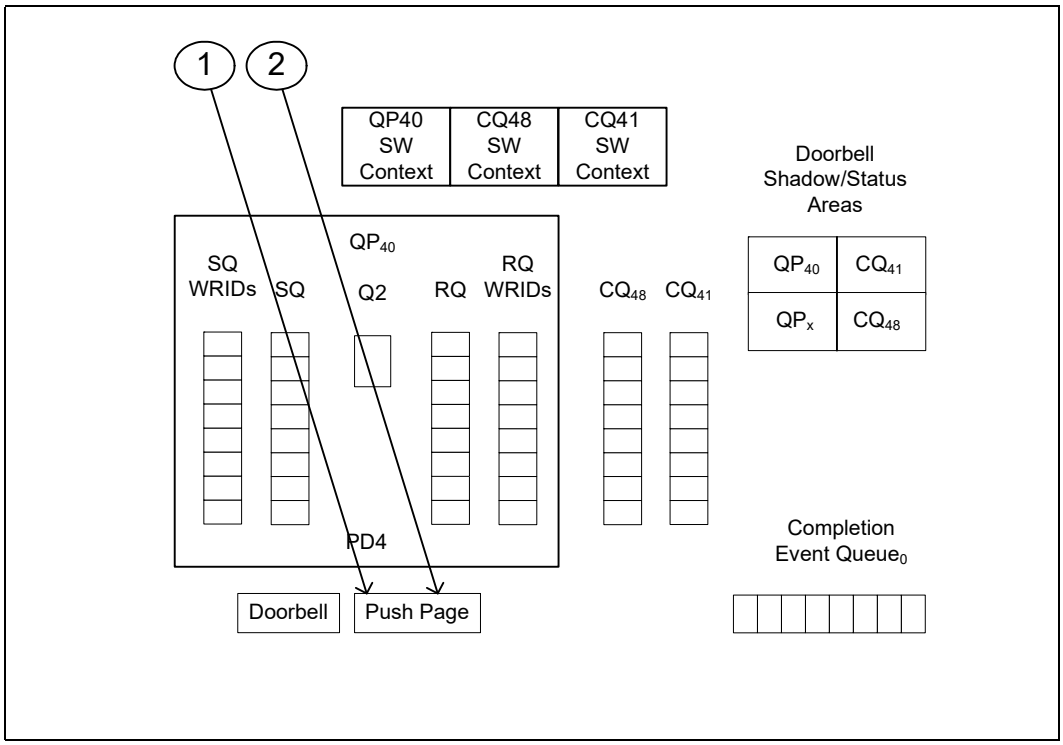
Figure 38-122.Split Push Page Detail



The push doorbell area is located at first 128 byte region of the push page. Push doorbell area for PFs contains 31 instances of the PFPE_WQEALLOC register. PFPE_WQEALLOC instances are continuously located within the push doorbell area starting at offset 4 bytes into the 128 byte region. Push doorbell area for VFs contains 31 instances of the VFPE_WQEALLOC register. VFPE_WQEALLOC instances are continuously located within the push doorbell area starting at offset 4 bytes into the 128 byte region. All QPs associated with a given push page use the same push doorbell area. Each instance of the PFPE_WQEALLOC and VFPE_WQEALLOC corresponds to the respective push SQE. The push SQE index matches an index of the respective PFPE_WQEALLOC or VFPE_WQEALLOC registers. When GLPE_PSHCFG.PSHCFG_DB_SPLIT is set, the WQEALLOC instances are located in odd numbered push pages and the push SQEs are located in even numbered push pages. Additionally, the stride of the doorbell array to 128 bytes instead of the packed array of doorbells shown in Figure 38-121. In other words, each odd numbered push page has WQEALLOC instances at offsets 0, 128, 256, ... instead of 0, 4, 8, ... in order to allow for write-combine avoidance on uncached writes to the push doorbells.

Figure 38-123 shows the mechanism for pushing a WQE to the 10 GbE controller.

Figure 38-123.PE Operation: SQ Push Mode



Push mode operation builds on the typical SQ operation shown in Figure 38-119. The following two steps replace step 4 from the procedure defined in Section 38.36.2.1.7.

1. Instead of the usual doorbell page write, a copy of the SQE is written to one of the push page SQEs from Figure 38-121. This page is configured with write combine attributes for this step. If less than 128 bytes WQEs are written to the 128-byte push WQE and the WQE did not start on a 128-byte boundary within the SQ, then the WQE data from the SQ must be written to the matching offset in the push WQE from the start of the most recent 128-byte boundary in the SQ. For example, if WQE index 5 from the SQ happens to be 32 bytes long and needs to be pushed, 32 bytes need to be written 32 bytes past the start of the Push WQE instead of at offset 0 in the push WQE. This is because WQE index 5 start 32 bytes past the previous 128-byte boundary in the SQ. Two additional requirements place on the writing of the push WQE are:
   a. 32-bit access are the minimum size data that is allowed to be written.
   b. Data must be rounded up to the next 32-byte length when written to the push WQE.
2. After writing the SQE to the push page SQE, the push page doorbell is written to indicate that the push is complete. Note that the doorbell written for non-push operations is not written for push operations.

Step 7 from the procedure described in Section 38.36.2.1.7 is typically skipped for push mode operations unless the 10 GbE controller is low on resources, the TC associated with the push page has insufficient bandwidth allocation to complete the operation. Push mode operations are also ignored if the TC associated with the push page is flow controlled. In the cases where the 10 GbE controller ignores the SQE pushed in Step 2, the WQE is re-fetched normally and processing continues. The only side effects of the dropped push operation is wasted bandwidth on the PCI bus and wasted CPU cycles.

In order to keep the wasted bandwidth and CPU cycles to a minimum, the next CQE generated after the dropped push operation indicates that the drop occurred and software should stop issuing push operations until it reaches an empty SQ condition.

SQ Suspend and Resume

Software is provided with CQP Commands allowing a suspend and resume operation of the SQ. Commands are defined in Section 38.36.3.5.8 and Section 38.36.3.5.9, respectively. Those commands are primarily used to migrate QP from one QS to another QS. This migration might be triggered by a system configuration change. For example, a change in number of TCs configured for the particular physical port, which in turn requires reassignment of QPs to the QSs associated with the TCs.

Software is required to reassign all QPs associated with particular QS prior posting an AQ Command that might result in QS removal. Such reassignment of the QP to the new QS should be performed using Suspend and Resume CQP commands, Section 38.36.3.5.8 and Section 38.36.3.5.9, respectively. Both Suspend and Resume CQP commands are asynchronous requests, and their completion does not indicate completion of a requested operation. Completion of a resume operation is never reported to software. Completion of a suspend operation is reported using AE_QP_SUSPEND_COMPLETE. Software is allowed to request multiple suspend and resume operations for different QPs accelerating transfer of multiple QPs from one QS to another.

RQ

As shown in Figure 38-118, RQs are organized as circular array of RQ Elements (RQEs). The host memory used for RQs can be physically contiguous or dis-contiguous. Each RQE is 32 bytes in size and can have up to 6 additional fragments for a total of 7. Unlike SQs, RQs all have fixed size Work Requests (WR) even if the application uses a smaller number of fragments that could be used to represent a WR. The minimum size of a RQ is four maximum sized WQEs for proper operations. For example, this means that if a WR can ever use 7 fragments, then the RQ must be 512 bytes. This is necessary to allow out of order placement of inbound RDMA send operations for iWARP QPs. All RQEs must start on an offset that is a multiple of RQ WQE size and an RQE along with its associated additional fragments must be contained in a single 128 bytes RQ read size buffer. This leads to the same RQ buffering requirements per WR as those listed for the SQ in Table 38-444.

Figure 38-124.PE RQ

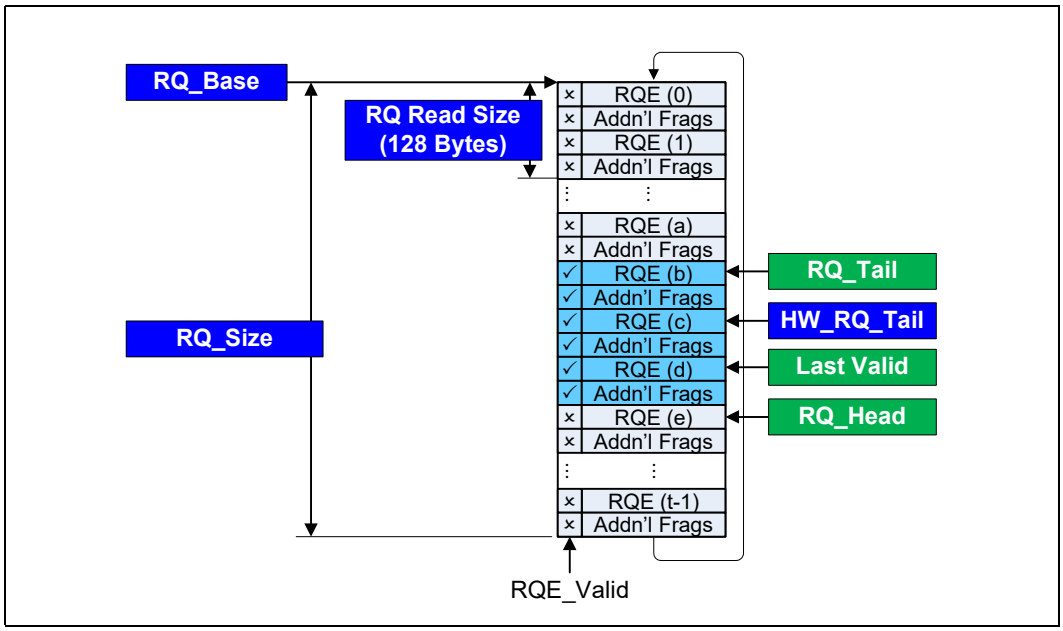
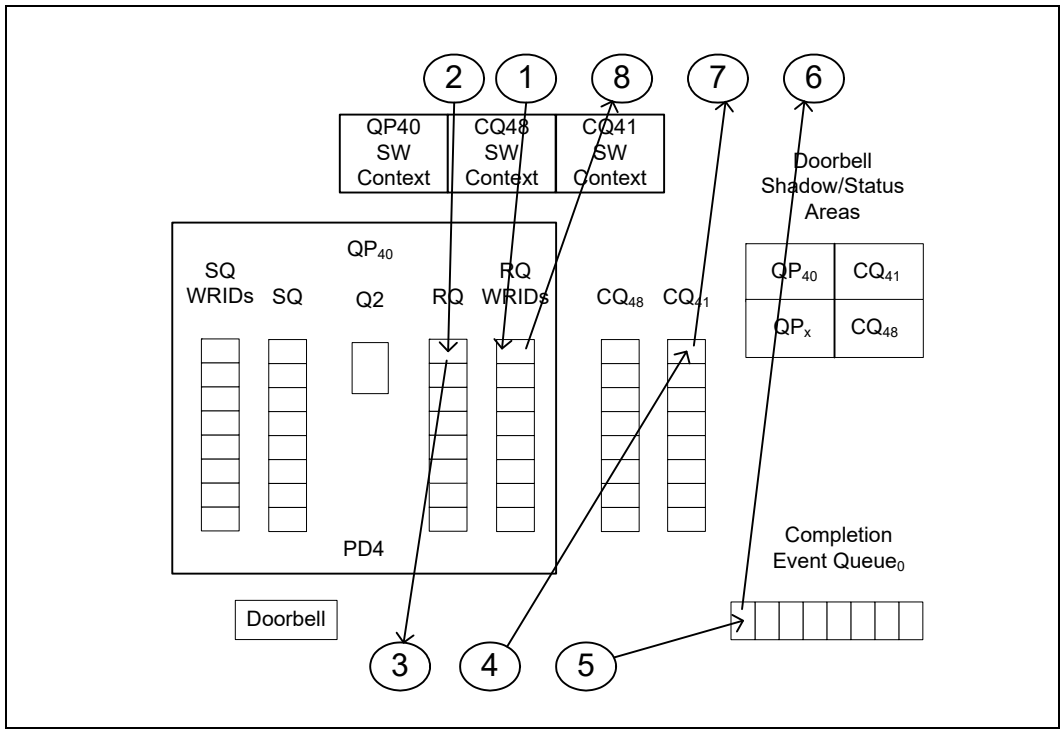


Figure 38-124 shows a typical RQ where software is tracking RQ_head and RQ_Tail and the 10 GbE controller tracks HW_RQ_tail in QP context. Software must size the RQ large enough to support every RQE with the maximum number of additional fragments. All RQEs are shown using two additional fragments and therefore takes 64 bytes. NOP RQEs can be added if necessary to pad RQEs out when necessary to satisfy the RQE alignment requirements. The 32-byte boundary requirements for RQEs means that there is no difference in RQE size between a WR with 2 fragments and a WR with 3 fragments.

Figure 38-125.PE Operation: RQ



The initialization required to bring a QP to an operation state was generally described in Section 38.36.2.1.7. Once this initialization has been performed, the steps in Figure 38-125 show how software and the 10 GbE controller interact for RQ operations.

1. Software saves the Work Request ID (WRID) passed in on the post receive verbs call.
2. Software builds a RQE using the fields that passed in on the post receive verbs call.
   a. The valid bit in the RQE must only be set once all fields are valid since the 10 GbE controller might read RQEs at any time. Note that the valid bit is a generational valid bit, which means that the first (and all subsequent odd) iterations through the RQ, 1 means valid, on all even iterations through the RQ, 0 indicates that the RQE is valid.
3. Once an operation targeting a QPs RQ is received, the 10 GbE controller reads QP context to determine the location of the RQE to be read.
   a. The RQE is read from host memory and the valid bit is checked. If the RQE is invalid, the packet associated with RQE is dropped or the QP is put into the error state and an AE is generated.
4. The 10 GbE controller reads the 128 byte RQ read block that contains the RQE indicated by the QP context value of HW_RQ_Tail
   a. The 10 GbE controller potentially processes all valid RQEs contained in the 128 byte RQ read block if more than one inbound operation targeted this QP's RQ. For this example it is assumed that a single inbound operation has been received.
   b. The 10 GbE controller processes RQE 0 which likely includes writing data from host memory and sending TCP acknowledgments.
   c. The 10 GbE controller increments the HW_RQ_Tail.

5. The 10 GbE controller writes to the CQ associated with the RQ.
6. The 10 GbE controller optionally writes to the CEQ if the CQ was written and generates an interrupt.
7. Software then fields the interrupt if one was generated and reads the CEQ to determine the CQ that generated the event (an application might be polling the CQ direction in which case steps 6 and 7 are skipped). Typically the CEQE contains the most significant 63 bits of the virtual address of CQ context to make CE processing efficient.
8. Software then reads the CQ to determine the QP and WQ that generated the work completion, the RQE index of the WR is reported in the CQE along with a 64-bit pointer that is typically set to the software QP context address at QP creation time.
9. Software then reads the RQ WRID array for the RQE index from the completion, the resulting WRID is returned to the application as an indication that the original WR has completed.

38.36.2.2 iWARP State Management

iWARP QP state is controlled by the application to a large extent. The verbs interface enables the application to manage QP transitions and provides rules for what the RNIC Interface (RI) must enforce. This section is not intended to repeat all information from the verbs specification. It is intended to be an overview of the expected usage model to be used with the 10 GbE controller. In the end, it is a combination of hardware support from the 10 GbE controller and supporting software that provides a verbs-compliant interface. Table 38-445 lists the operations performed by the 10 GbE controller based on the iWARP QP state.

Table 38-445.iWARP QP State Behavior

QP State	QP Activities
Non-existent	• Doorbell rings are dropped silently. • No transmit (because doorbell rings are dropped). • No receive processing (no hash entry).
Idle	• SQ/RQ WQEs can be posted. • No SQ WQEs are processed. • No receive processing (no hash entry).
RTS	• Normal SQ/RQ/CQ/CEQ processing (hash entry added on transition to RTS by firmware).
Closing	• No new SQ work is expected, but if a door bell rings while in this state, Product Name schedules work and reports an AE BAD_CLOSE (note this is not LLP_BAD_CLOSE) and transitions to an error state. • No new RQ work is expected; however, if a packet arrives with data it is treated as an error (BAD_CLOSE) and the QP transitions to error state.
Terminate	• No RQ processing (discarding received data). • No new SQ WQEs processed. • SQ WQEs might be retransmitted. • Terminate/FIN might be transmitted.
Error	• No SQ WQEs processed. • No RQ WQEs processed. • Host software eventually requests CQP to remove the hash entry.

QP state transitions and the associated software and hardware/firmware actions driven from software interactions with the application are listed in Table 38-446.

Table 38-446.iWARP QP State Transition (Sheet 1 of 2)

Initial and Next QP State	Software Actions During State Transition	Hardware Actions During State Transition
Nonexistent ≥ Idle (CreateQP)	• Initialize the HMC backing pages necessary to support the QP. • Allocate/Initialize all software tracking structures for the PD, CQ, QP. • Assign QP to a PD. • Create CQ. • Allocate SQ and RQ. • Allocate terminate message buffer (Q2). • Allocate the doorbell shadow area. • Perform parameter checks specified by the verbs specification. • Issue CreateQP WQE to CQP.	• Update QP context.
Idle ≥ Idle	• Perform parameter checks specified by the verbs specification. • Issue Modify QP to CQP, next state = Idle.	• QP state parameters are updated.
Idle ≥ RTS	• Create a TCP socket and ensure that it is in the established state. • Allocate and initialize a local MAC table entry. • If a streaming mode message is desired, ensure that it has been placed on the SQ immediately following any unacknowledged TCP send data. • Perform parameter checks specified by the verbs specification. • Issue Modify QP to CQP, next state = RTS.	• Set the TCP state to established and iWARP state to RTS. • Insert the QP's hash table entry to receive accelerated data. • Start scheduling work for the QP.
Idle ≥ Error	• Perform parameter checks specified by the verbs specification. • Issue modify QP WQE to CQP with the next state set to error, reset connection (if connection not already reset) and remove quad hash (if quad hash entry has not been previously removed) by setting appropriate bits in the bits set. • If any WQEs are still pending, issue flush WQEs operation to CQP after the Modify QP operation completes, if the QP has been setup with a last streaming mode message, the first WQE on the SQ is the streaming mode message, not an iWARP message, a unique completion context should be put in the LSSM WQE in order to correctly distinguish it from iWARP WQEs. • Handle poll for completion requests for flushed WQEs, the CQ has one CQE for each WQ that had pending WQEs, software must report all pending WQEs as flushed once it gets the single flushed CQE (per WQ with pending WQEs), the number of WQEs flushed can be determined from software's head and tail for the SQ and RQ at the point in time that the poll for completion call is made that returns a flushed CQE.	• Set iWARP state to error. • When the flush WQEs operation is issued and any WQEs are outstanding on the SQ and/or RQ as requested by host software, complete the first pending WQE for each WQ (flushed completion status).
RTS ≥ Closing	• Perform parameter checks specified by the verbs specification. • Stop processing new PostSQ/RQ requests. • Ensure that no outstanding SQ or outbound RDMA read responses are pending (this is really up to the application). • Start a timer to put a bound on how long a connection is allowed to stay in the closing state. • Submit a Modify QP WQE to CQP the next state = Closing. • If the completion error is a bad close, generate an AE, QP state is already error • If any RQ WQEs are still pending, issue Flush WQEs operation to CQP.	• If there is pending SQ or Q1 work, generate a completion error (bad close), software needs to translate into an AE. • Else, update iWARP state to closing. • When the flush WQEs operation is issued and any RQ WQEs are outstanding, complete the first pending WQE for each WQ (flushed completion status).

Table 38-446.iWARP QP State Transition (Sheet 2 of 2)

Initial and Next QP State	Software Actions During State Transition	Hardware Actions During State Transition
RTS ≥ RTS	• Perform parameter checks specified by the verbs specification. • Ensure that the number of currently pending outbound RDMA read requests is less than or equal to the new ORD value. • Ensure the only parameter that is changed is ORD size, it is only allowed to shrink or remain the same.	• Update the ORD value.
RTS ≥ Terminate or Terminate ≥ Terminate	• Perform parameter checks specified by the verbs specification for the RTS -> Terminate case. • Set terminate action (usually send FIN and terminate). • Start a timer to put a bound on how long the QP is allowed to stay in the terminate state in the case of an abortive tear down. • If completion error indicates tcp_state = error: • a. issue Modify QP WQE to CQP with next state = error. • Generate LLP connection reset AE. • If terminate sent AE is received, issue an additional Modify QP with next state = Terminate and terminate action set to 3 (do not sent FIN or terminate). • If completion code indicates LLP closed then a the TCP state has transitioned to closed or time wait so an LLP close complete AE should be generated by software.	• If the TPC state = Closed, generate completion error indicating tcp_state was closed (indicates a reset was received). • Update iWARP state to terminate. • Once all outstanding TCP segments have been acknowledged, send the terminate message and/or FIN. • Generate a terminate sent AE.
RTS ≥ Error or Closing ≥ Error or Terminate ≥ Error	• Perform parameter checks specified by the verbs specification. • See IDLE ≥ Error case for CQP and completion processing. • Block until the RST sent AE has been seen (probably want to start a timer, and upload QP context if it expires to see what has happened). • If any WQEs are still pending, issue flush WQEs operation to CQP after the Modify QP operation completes, if the QP has been setup with a last streaming mode message, the first WQE on the SQ is the streaming mode message, not an iWARP message, a unique completion context should be put in the LSSM WQE in order to correctly distinguish it from iWARP WQEs.	• If host software requested the connection to be reset, send a TCP RST segment. • Remove the QP's HTE if the Remove_Hash_Entry field is set. • Update iWARP state to error. • Wait for all packets to clear the pipeline. • Clean up QP context (prevent future scheduling). • Generate a RST sent AE. • Set the TCP state to closed.
Error ≥ Idle	• Software issues discard and create QP operations to CQP.	
Error ≥ Error	• Submit a Modify QP WQE to CQP with next state = Error and the reset_connection bit set. • If any WQEs are still pending, issue flush WQEs operation to CQP after the Modify QP operation completes, if the QP has been setup with a last streaming mode message, the first WQE on the SQ is the streaming mode message, not an iWARP message, a unique completion context should be put in the LSSM WQE in order to correctly distinguish it from iWARP WQEs.	• See RTS ≥ error.

QP state transitions and the associated software and hardware/firmware actions driven from wire interactions with remote peer are listed in Table 38-447.

Table 38-447.iWARP QP State Transition Driven from the Wire

Row #	Wire Activity	Initial QP State (TCP, iWARP)	10 GbE Controller	Software
0	FIN Received	EST, RTS	Set the TCP state to close wait (note that iWARP state is not changed). If no work was outstanding on the SQ or Q1: Issue AE_LLP_FIN_RECEIVED. Else Issue AE_RDMAP_ROE_BAD_LLP_CLOSE. Send ACK to the FIN. Continue processing SQ WQEs.	Stop processing new post RQ operations if not already stopped. Issue appropriate AEs. If system software does not automatically generate closing, start the closing process based on the received event by issuing Modify QP with the next state set to closing.
1	FIN Received	FW1, Closing	Send ACK to the FIN. If the ACK receive with the inbound FIN ACKs the transmitted FIN: Set the TCP state to Time Wait. Issue AE_LLP_CLOSE_COMPLETE. Else Set the TCP state to Closing (note that iWARP state is not changed).	If the AE_LLP_CLOSE_COMPLETE is received: Cancel the pending disconnect fail safe timer. Stop processing new post RQ operations if not already stopped. SQ operation processing should have been stopped on the Modify QP with the next state set to closing. Issue appropriate AEs. Issue Flush WQE CQP operation. The application or system software starts the QP cleanup process.
2	FIN Received	FW2, Closing	Send ACK to the FIN. Set the TCP state to time wait (note that iWARP state is not changed). Issue AE_LLP_CLOSE_COMPLETE.	See row 1 software actions.
3	FIN Received	EST, Terminate	See Row 0 10 GbE Controller actions.	See row 1 software actions.
4	FIN Received	FW1, Terminate	See Row 1 10 GbE Controller. actions.	See row 1 software actions.
5	FIN Received	FW2, Terminate	See Row 2 10 GbE Controller actions.	See row 1 software actions.
7	ACK Received	FW1, Closing or Terminate	Set the TCP state to FW2.	N/A
8	ACK Received	FW2, Closing or Terminate	Send ACK to the FIN. Set the TCP state to time wait (note that iWARP state is not changed). Issue AE_LLP_CLOSE_COMPLETE.	See row 1 software actions.
9	ACK Received	Closing, Closing or Terminate	Set the TCP state to time wait (note that iWARP state is not changed). Issue AE_LLP_CLOSE_COMPLETE.	See row 1 Software actions.

38.36.2.3 Partial FPDU Support

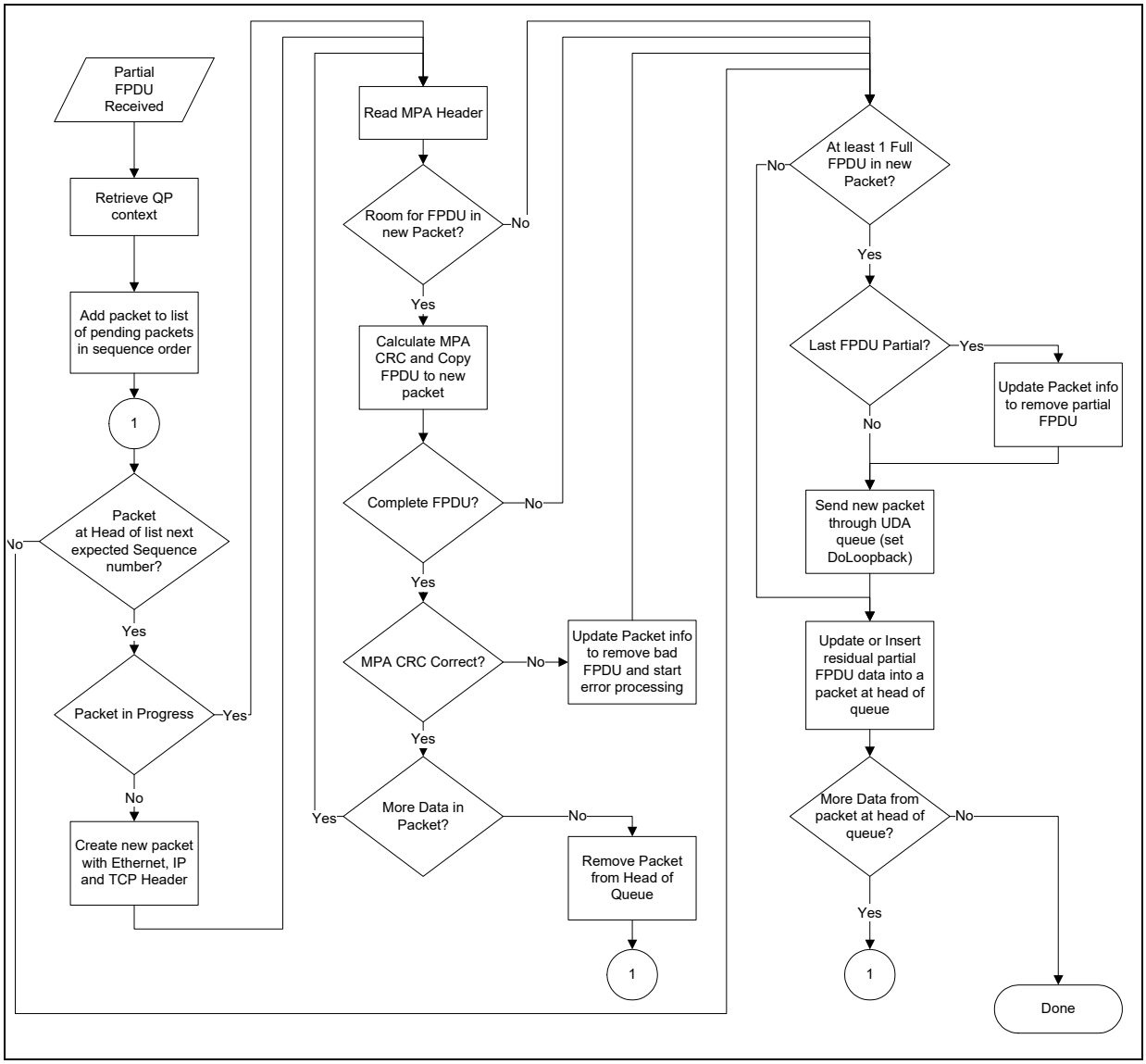
The 10 GbE controller supports partial iWARP FPDUs using software assist. High performance iWARP implementations pay attention to the underlying network MTU and form iWARP FPDUs that do not span Ethernet segments. In the case where software iWARP implementations (or other RNICs) form iWARP FPDUs that span Ethernet segments, the 10 GbE controller reports these Ethernet packets to the UDA queues. Additionally, the PE writes the first_partial_sequence_number field in the Q2 area of the associated QP context. See Section 38.36.4.1 for the definition of the Q2 area. Once a partial FPDU has been received, all subsequent packets for that QP are also forwarded to the UDA queue until software has been able to process the packets and send them back to the 10 GbE controller. Host software must process the packets and break them into full FPDUs and send them back to the 10 GbE controller through a UDA SQ with the *SWPE* bit set. The 10 GbE controller then processes the packets as if they

had come from the wire as full FPDUs. Once all the packets have been sent back to the 10 GbE controller, the QP is automatically set back to normal operation and no further packets are sent to the UDA queues until a new partial FPDU is received. Each time the Q2 area has an updated expected sequence number, the Sequence_Update_Toggle field is toggled. Software must keep track of the last value that it has seen in the Q2 area. If the value does not match, software should update its expected sequence number.

Note: Software must perform the MPA CRC check on all packets that are sent back through to the PE using the *SWPE* bit in addition to reforming full iWARP FPDUs. The 10 GbE controller does not support checking the MPA CRC in hardware on this pass-through chip.

The high-level software algorithm used to handle partial FPDUs is shown in Figure 38-126.

Figure 38-126.Partial FPDU Software Algorithm



38.36.2.4 CEQ Entry Format

A CEQ is a circular ring of CEQEs in host memory. CEQs can be virtually or physically contiguous and are managed using the control QP (see Section 38.36.3.4.2).

Table 38-448.PE CEQ Entry Format

Byte Offset	[Bit Range]Field Name	
0	[63]CEQE_Valid	[62:0]CQ_Context_Value

CEQE_Valid (1)

The CEQE_Valid bit for CEQE is a bit that indicates that a CEQE is ready to be processed by software. The polarity of the valid bit changes each time the CEQ wraps from the last entry back to the first entry. This change in polarity reduces software overhead by avoiding the need to clear the *Valid* bit once software has processed a valid CEQE. Software is responsible to clear (set to 0b) all memory in a CEQ initially at CEQ creation. The first iteration (and subsequent odd numbered iterations) through the CEQ, the 10 GbE controller sets the *Valid* bit to 1b when it writes a new CEQE. For the second iteration (and all even numbered iterations) through the CEQ, the 10 GbE controller sets the *Valid* bit to 0b when it writes a new CEQE.

CQ_Context_Value (63)

CQ_Context_Value is the value of CQ_Context_Value specified at CQ creation (see Section 38.36.3.3.3). Typically this value is the low 63 bits of a pointer to a software CQ object to enable software to quickly process new CEs.

38.36.2.5 AEQ Entry Format

An AEQ is a circular ring of Asynchronous Event Queue Entries (AEQEs) in host memory. AEQs can be virtually or physically contiguous and are managed using the Control QP (see Section 38.36.3.5).

Table 38-449.PE AEQ Entry Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	Completion_Context_Value		
8	[63]	AEQE_Valid	[49:34]	AE_Code
	[62:61]	Q2_data_written	[33]	AEQE_Overflow
	[60:57]	TCP_State	[32]	Reserved
	[56:54]	iWARP_State	[31:18]	WQ_Desc_Index
	[53:50]	AE_Source	[17:0]	QP_CQ_ID

Completion_Context_Value (64)

Completion_Context_Value is a software supplied token that can be used to efficiently locate the necessary software resources necessary to process an AE. The token that is reported can be the QP_Completion_Context from the QP that caused the AE or can be the CQ_Context_Value supplied during CQ creation depending upon the AE_ID and AE_Source fields of the AE.

WQ_Desc_Index (14)

WQ_Desc_Index value is valid only if the source of the event is a WQ (SQ or RQ) and the AE_Code indicates that the WQ_Desc_Index field is valid. See Section 38.36.2.6 for more details on how to determine if WQ_Desc_Index is valid.

QP_CQ_ID (18)

This field carries either the QP ID that is associated with the AE or the CQ ID that is associated with the AE. The AE_Source field (defined later) or the AE_ID field are used to determine if the AE is associated with a QP or a CQ.

AE_Code (16)

AE_Code is the AE code the caused the AE. See Section 38.36.2.6 for more details.

AE_Source (4)

AE_Source defines the source of the AE. The following table lists the values and the related source of the problem.

AE_Source	Description
0000b	Reserved.
00x1b	The AE is associated with the RQ of a QP. QP_CQ_ID is the QP ID. Completion_Context_Value is set to the value of QP_Completion_Context from the associated QP and WQ_Desc_Index is valid.
xx10b	The AE is associated with a CQ. QP_CQ_ID is the CQ ID and Completion_Context_Value is the CQ_Context_Value specified at CQ creation. WQ_Desc_Index not valid.
01x1b	QP_CQ_ID is the QP ID. Completion_Context_Value is set to the value of QP_Completion_Context from the associated QP and WQ_Desc_Index is valid. The event is associated with the SQ of a QP.
10x1b	The AE is associated with an inbound RDMA write or inbound RDMA read response operation related to a QP. QP_CQ_ID is the QP ID. Completion_Context_Value is set to the value of QP_Completion_Context from the associated QP and WQ_Desc_Index is not valid.
11x1b	The AE is associated with an outbound RDMA read response operation related to a QP. This value can also be returned is other cases in which the WQ_Desc_Index value cannot be retrieved by the 10 GbE controller. Completion_Context_Value is set to the value of QP_Completion_Context from the associated QP and WQ_Desc_Index is not valid.

Note that some AE_ID values uniquely define the source of the AE. For these events, AE_Source should be ignored. See Section 38.36.2.6 for more details.

iWARP_State (3)

iWARP_State reflects the iWARP state of the connection at the time the AE occurred according to the 10 GbE controller hardware.

iWARP_State	Description
0	Nonexistent
1	Idle
2	Ready To Send (RTS)
3	Closing
4	Reserved
5	Terminate
6	Error
7	Reserved

TCP_State (4)

TCP_State reflects the TCP state of the connection at the time the AE occurred according to the 10 GbE controller hardware.

TCP_State	Description
0	Nonexistent
1	Closed
2	Listen

TCP_State	Description
3	Syn_Sent
4	Syn_Received
5	Established
6	Close_Wait
7	Fin_Wait_1
8	Closing
9	Last_Ack
10	Fin_Wait_2
11	Time_Wait
12-15	Reserved

AEQE_Valid (1)

The AEQE_Valid bit for AEQE is a bit that indicates that a AEQE is ready to be processed. The polarity of the valid bit changes each time the AEQ wraps from the last entry back to the first entry. This change in polarity reduces software overhead by avoiding the need to clear the *Valid* bit once software has processed a valid AEQE. Software is responsible to clear (set to 0b) all memory in a AEQ initially at AEQ creation. The first iteration (and subsequent odd numbered iterations) through the AEQ, the 10 GbE controller sets the *Valid* bit to 1b when it writes a new AEQE. For the second iteration (and all even numbered iterations) through the AEQ, the 10 GbE controller sets the *Valid* bit to a 0b when it writes an new AEQE.

AEQE_Overflow (1)

The AEQE_Overflow bit for AEQE is a bit that indicates that a AEQE has been completely filled and that there might have been a loss of subsequent AEQEs. No further AEQEs are generated until additional space is allocated to the AEQ using the PFPE_AEQALLOC or VFPE_AEQALLOC register.

Q2_data_written (2)

Q2_data_written provides status regarding error information that is reported for errors detected for iWARP connections when an AE is generated. The Q2 data area is configured as part of QP setup. See Table 38.36.3.3.2 for more information. This values for this field are:

0 = No data associated with this AE has been written to Q2.

1 = Data written to Q2 starts at the Ethernet header of the offending packet.

2 = Data written to Q2 starts at the MPA header of the offending PDU.

3 = Reserved.

38.36.2.6 AE Codes

Table 38-450 lists the values that can be returned in the AE_ID field of an AEQ entry. The column titled Source defines if the source of the AE is pre-determined based on the AE code or the AE_Source field of the AE field must be used to determine the source. Note that the source of the AE effects the Completion_Context_Value, WQ_Desc_Index, and QP_CQ_ID fields in the AE queue entry defined in section Table 38.36.2.5. The WQ_Desc_Index field is not valid for AE codes that have a source designated as QP or CQ.

Table 38-450.PE AE codes (AE_ID) (Sheet 1 of 6)

Asynchronous Event Code	Value	Description	iWARP State After AE	Source
AE_AMP_UNALLOCATED_STAG	0x0102	This AE is generated when an memory protection error has been detected by the 10 GbE controller. This error indicates that attempt was made to access a memory region or memory window that is in an un-allocated state.	Terminate	AE_Source
AE_AMP_INVALID_STAG	0x0103	This AE is generated when an memory protection error has been detected by the 10 GbE controller. This error indicates that attempt was made to access a memory region or memory window that is in an invalid state.	Terminate	AE_Source
AE_AMP_BAD_QP	0x0104	This AE is generated when an memory protection error has been detected by the 10 GbE controller. This error indicates that attempt was made to access a memory window from a different QP than the QP to which the memory window was bound.	Terminate	AE_Source
AE_AMP_BAD_PD	0x0105	This AE is generated when an memory protection error has been detected by the 10 GbE controller. This error indicates that attempt was made to access a memory region or memory window from a QP with a different PD than the PD associated memory region or window.	Terminate	AE_Source
AE_AMP_BAD_STAG_KEY	0x0106	This AE is generated when an memory protection error has been detected by the 10 GbE controller. This error indicates that attempt was made to access a memory region or memory window when the key portion of the STag did not match the key associated with the memory region or window.	Terminate	AE_Source
AE_AMP_BAD_STAG_INDEX	0x0107	This AE is generated when an memory protection error has been detected by the 10 GbE controller. This error indicates that attempt was made to access a memory region or memory window with an STag index larger than the largest index allowed for the PCI function.	Terminate	AE_Source
AE_AMP_BOUNDS_VIOLATION	0x0108	This AE is generated when an memory protection error has been detected by the 10 GbE controller. This error indicates that attempt was made to access memory outside of the memory area defined by the memory region or memory window.	Terminate	AE_Source
AE_AMP_RIGHTS_VIOLATION	0x0109	This AE is generated when an memory protection error has been detected by the 10 GbE controller. This error indicates that attempt was made to access a memory region or memory window and a rights violation occurred.	Terminate	AE_Source
AE_AMP_TO_WRAP	0x010A	This AE is generated when an memory protection error has been detected by the 10 GbE controller. This error indicates that attempt was made to access a memory region or memory window and the tagged-offset plus the ULPDU length caused the address generated for the memory region or window to wrap.	Terminate	AE_Source
AE_AMP_FASTREG_VALID_STAG	0x010C	This AE is generated when an memory protection error has been detected by the 10 GbE controller. This error indicates that attempt was made to issue a registration operation to a valid memory region.	Terminate	AE_Source

Table 38-450.PE AE codes (AE_ID) (Sheet 2 of 6)

Asynchronous Event Code	Value	Description	iWARP State After AE	Source
AE_AMP_FASTREG_MW_STAG	0x010D	This AE is generated when an memory protection error has been detected by the 10 GbE controller. This error indicates that attempt was made to issue a registration operation to a memory window.	Terminate	AE_Source
AE_AMP_FASTREG_INVALID_RIGHTS	0x010E	This AE is generated when an memory protection error has been detected by the 10 GbE controller. This error indicates that a rights mismatch was detected during a registration operation.	Terminate	AE_Source
AE_AMP_FASTREG_INVALID_LENGTH	0x0110	This AE is generated when an memory protection error has been detected by the 10 GbE controller. This error indicates that attempt was made to issue a registration operation with an invalid length.	Terminate	AE_Source
AE_AMP_INVALIDATE_SHARED	0x0111	This AE is generated when an memory protection error has been detected by the 10 GbE controller. This error indicates that attempt was made to invalidate a shared memory region.	Terminate	AE_Source
AE_AMP_INVALIDATE_NO_REMOTE_ ACCESS_RIGHTS	0x0112	This AE is generated when an memory protection error has been detected by the 10 GbE controller. This error indicates that attempt was made to remotely invalidate a memory region or memory window that did not have remote access rights specified.	Terminate	AE_Source
AE_AMP_INVALIDATE_MR_WITH_BOU ND_WINDOWS	0x0113	This AE is generated when an memory protection error has been detected by the 10 GbE controller. This error indicates that attempt was made to invalidate a memory region that still had one or more memory windows bound to it.	Terminate	AE_Source
AE_AMP_MWBIND_VALID_STAG	0x0114	This AE is generated when an memory protection error has been detected by the 10 GbE controller. This error indicates that the STag used on a bind operation was a memory window that was already valid.	Terminate	AE_Source
AE_AMP_MWBIND_OF_MR_STAG	0x0115	This AE is generated when an memory protection error has been detected by the 10 GbE controller. This error indicates that the STag used on a bind operation was a memory region instead of a memory window.	Terminate	AE_Source
AE_AMP_MWBIND_TO_ZERO_BASED_ STAG	0x0116	This AE is generated when an memory protection error has been detected by the 10 GbE controller. This error indicates that an attempt was made to bind a memory window to a zero-based memory region.	Terminate	AE_Source
AE_AMP_MWBIND_TO_MW_STAG	0x0117	This AE is generated when an memory protection error has been detected by the 10 GbE controller. This error indicates that the STag specified for the memory window to be bound to was a memory window instead of a memory region.	Terminate	AE_Source
AE_AMP_MWBIND_INVALID_RIGHTS	0x0118	This AE is generated when an memory protection error has been detected by the 10 GbE controller. This error indicates that a rights violation was detected during a memory window bind operation.	Terminate	AE_Source

Table 38-450.PE AE codes (AE_ID) (Sheet 3 of 6)

Asynchronous Event Code	Value	Description	iWARP State After AE	Source
AE_AMP_MWBIND_INVALID_BOUNDS	0x0119	This AE is generated when an memory protection error has been detected by the 10 GbE controller. This error indicates that a bounds violation was detected during a memory window bind operation.	Terminate	AE_Source
AE_AMP_MWBIND_TO_INVALID_ PARENT	0x011A	This AE is generated when an memory protection error has been detected by the 10 GbE controller. This error indicates that the memory region that a memory window attempted to bind to was invalid.	Terminate	AE_Source
AE_AMP_MWBIND_BIND_DISABLED	0x011B	This AE is generated when an memory protection error has been detected by the 10 GbE controller. This error indicates that a memory window Bind operation was attempted on a memory region that had bind support disabled.	Terminate	AE_Source
AE_PRIV_OPERATION_DENIED	0x011C	This AE is generated when a privileged operation is attempted on a non-privileged QP, this includes bind operations without bind enabled, fast register without fast register enabled, and STag zero without privilege is not enabled.	Terminate	QP
AE_AMP_MWBIND_WRONG_TYPE	0X120	A memory window bind specified a non-matching window type.	Terminate/ Error	QP
AE_AMP_FASTREG_PBLE_MISMATCH	0X121	Invalid VM fast register request to change a physical MR to a virtually mapped MR or vice versa.	Terminate/ Error	QP
AE_UDA_XMIT_DGRAM_TOO_LONG	0x0132	This AE is generated by the 10 GbE controller when a total length of packet exceeds MSS configured for QP.	Terminate	QP
AE_UDA_XMIT_DGRAM_TOO_SHORT	0x0134	This AE is generated by the 10 GbE controller when a total length of packet is smaller than 16 bytes.	Terminate	QP
AE_BAD_CLOSE	0x0201	This AE is generated when in iWARP state = closing and an iWARP PDU is received. If the 10 GbE controller detects SQ or Q1 work during a Modify QP RTS ≥ closing then a completion error is returned to the Modify QP request instead of an AE.	Error	QP
AE_RDMAP_ROE_BAD_LLP_CLOSE	0x0202	This AE indicates that there was either SQ or Q1 work pending when a FIN was received or data was received with the FIN.	Terminate~~- or Error~~	QP
AE_CQ_OPERATION_ERROR	0x0203	Product Name attempted to generate a CQ entry on an empty CQ that has overflow detection enabled.	N/A	AE_Source
AE_RDMA_READ_WHILE_ORD_ZERO	0x0205	This AE is generated when an RDMA read request is issued to a QP that has the Outbound RDMA Read Queue Depth (ORD) set to 0b.	Terminate	QP
AE_STAG_ZERO_INVALID	0x0206	Detected on an inbound RDMA write, RDMA read response, or RDMA read request (either source or sink). Outbound check is also performed for SQ and RQ operations when a QP is not privileged. Inbound 0 byte RDMA read and RDMA write operations are not indicated as errors since no host data is accessed.	Terminate	QP
AE_IB_RREQ_AND_Q1_FULL	0x0207	This AE is generated if an inbound RDMA read is received and the target QP's inbound RDMA RQ (Q1) is full.	Terminate	QP

Table 38-450.PE AE codes (AE_ID) (Sheet 4 of 6)

Asynchronous Event Code	Value	Description	iWARP State After AE	Source
AE_WQE_UNEXPECTED_OPCODE	0x020A	This AE is generated if QP encounters a WQE with an invalid OP code on a SQ.	Transmit Suspended	QP
AE_WQE_INVALID_PARAMETER	0x020B	This AE is generated if QP encounters a WQE with invalid parameters.	Terminate	AE_Source
AE_WQE_LSMM_TOO_LONG	0x0220	This AE is generated if the Last streaming mode message specified on the Modify QP operation from the iWARP IDLE state to RTS state is longer than the MSS.	Terminate	AE_Source
AE_DDP_INVALID_MSN_GAP_IN_MSN	0x0301	The AE is generated with a gap is detected in the MSN used to index the RQ or Q1 when data is in order with respect to TCP sequence space.	Terminate	AE_Source
AE_DDP_UBE_DDP_MESSAGE_TOO_LONG_FOR_AVAILABLE_BUFFER	0x0303	This AE is generated on inbound iWARP PDUs if the message offset plus the size of the PDU data is larger than the number of bytes in the RQ WQE targeted by the PDU.	Terminate	AE_Source
AE_DDP_UBE_INVALID_DDP_VERSION	0x0304	This AE is generated when an inbound iWARP PDU has an incorrect DDP version number.	Terminate	QP
AE_DDP_UBE_INVALID_MO	0x0305	This AE is generated if the message offset is larger than the number of bytes in the RQ WQE targeted by the PDU on inbound iWARP PDUs.	Terminate	QP
AE_DDP_UBE_INVALID_MSN_NO_BUFFER_AVAILABLE	0x0306	This AE is generated if and inbound send operation targets an unallocated RQ WQE and the 10 GbE controller is configured to drop the connection for this condition.	Terminate	AE_Source
AE_DDP_UBE_INVALID_QN	0x0307	This AE is generated when an inbound untagged iWARP PDU has an incorrect queue number.	Terminate	QP
AE_DDP_NO_L_BIT	0x0308	Detected when there is no L bit on an inbound RDMA read request.	Terminate	QP
AE_RDMAP_ROE_INVALID_RDMAP_VERSION	0x0311	This AE is generated when an inbound iWARP PDU has an incorrect RDMAP version number.	Terminate	QP
AE_RDMAP_ROE_UNEXPECTED_OPCODE	0x0312	This AE is generated when an inbound iWARP PDU has an incorrect opcode number.	Terminate	QP
AE_ROE_INVALID_RDMA_READ_REQUEST	0x0313	This AE is generated when an inbound RDMA read iWARP PDU is received but the QP is not enabled for inbound RDMA read support.	Terminate	QP
AE_ROE_INVALID_RDMA_WRITE_OR_READ_RESP	0x0314	This AE is generated when an inbound RDMA read response or RDMA write iWARP PDU is received but the QP is not enabled for inbound RDMA read responses or RDMA writes.	Terminate	QP
AE_INVALID_ARP_ENTRY	0x0401	This AE is reported when a connection attempts to transmit a packet using an invalid ARP entry. Host software will either enable the QP for transmitting after updating the ARP entry or upload/discard the connection because the ARP entry is no longer valid.	Transmit Suspended	QP
AE_INVALID_TCP_OPTION_RCVD	0x0402	This AE is generated when the 10 GbE controller encounters an unsupported TCP option and the 10 GbE controller is not configured to ignore unsupported TCP options via the ignore_tcp_uns_options QP context bit. The packet with invalid TCP options are dropped if this AE is generated.	No change	QP
AE_STALE_ARP_ENTRY	0x0403	This AE is reported by the first connection that attempts to transmit a packet using a stale ARP entry.	No change	QP

Table 38-450.PE AE codes (AE_ID) (Sheet 5 of 6)

Asynchronous Event Code	Value	Description	iWARP State After AE	Source
AE_INVALID_MAC_ENTRY	0x0405	This AE indicates that an invalid MAC table entry was referenced from a QP transmit operation.	Terminate	AE_Source
AE_LLP_CLOSE_COMPLETE	0x0501	This AE indicates that the TCP state of an iWARP QP has transitioned to either closed or time wait in a graceful fashion.	No change	QP
AE_LLP_CONNECTION_RESET	0x0502	This AE indicates that a TCP packet with the *RST* bit set has been received on a TCP connection that is associated.	Error	QP
AE_LLP_FIN_RECEIVED	0x0503	This AE indicates that a TCP packet with the FIN bit set has been received without data and the iWARP state is RTS (implies TCP state is established), host software is responsible for issuing a Modify QP ≥ closing to complete the QP transition to the closing the state.	No change	QP
AE_LLP_RECEIVED_MPA_CRC_ERROR	0x0505	This AE is reported when an MPA CRC error is detected.	Terminate	QP
AE_LLP_SEGMENT_TOO_SMALL	0x0507	This AE is reported when an iWARP segment is received that is too small to contain a DDP or RDMAP header.	Terminate	QP
AE_LLP_SYN_RECEIVED	0x0508	This AE indicates that a TCP packet with the SYN bit set is received on a connection that is associated with an iWARP connection.	Error	QP
AE_LLP_TERMINATE_RECEIVED	0x0509	This AE indicates that a terminate message has been received for an iWARP QP. The received terminate data is placed in the Q2 area associated with the QP in host memory.	Terminate	QP
AE_LLP_TOO_MANY_RETRIES	0x050A	This AE is used to report too many retransmission retries.	Transmit Suspended	QP
AE_LLP_TOO_MANY_KEEPALIVE_ RETRIES	0x050B	This AE is used to report too many keep alive attempts.	Transmit Suspended	QP
AE_LLP_DOUBT_REACHABILITY	0x050C	This AE is used to report that it has reached the threshold for the given connection. This is used in dead gateway detection or black holes.	No change	QP
AE_RESOURCE_EXHAUSTION	0x0520	This AE is used to report that a QP attempted to allocate Q1 or XMIT FIFO resource and no resource was available.	No change	AE_Source
AE_RESET_SENT	0x0601	This AE is generated when the 10 GbE controller sends a reset after host software has requested the connection to be reset via a Modify QP operation.	No change	QP
AE_TERMINATE_SENT	0x0602	This AE is generated when the 10 GbE controller sends a terminate message after host software has requested a terminate to be sent on a connection via a Modify QP operation.	No change	QP
AE_RESET_NOT_SENT	0x0603	This AE is generated when the 10 GbE controller did not send a reset after host software has requested the connection to be reset via a Modify QP operation. The reset will not be sent due to TCP state already having been transitioned to time wait or closed.	No change	QP
AE_LCE_QP_CATASTROPHIC	0x0700	This AE is generated by the 10 GbE controller when it receives an error on a hardware transaction.	Terminate	QP

Table 38-450.PE AE codes (AE_ID) (Sheet 6 of 6)

Asynchronous Event Code	Value	Description	iWARP State After AE	Source
AE_LCE_FUNCTION_CATASTROPHIC	0x0701	This AE is generated by the 10 GbE controller when it receives an error on a hardware transaction that impacts the entire PCI function. QP_CQ_ID is not reliable for this AE and should be ignored.	Terminate	N/A
AE_LCE_CQ_CATASTROPHIC	0x0702	This AE is generated by the 10 GbE controller when it receives a local catastrophic error on a hardware transaction that impacts a CQ.	Terminate	CQ
AE_QP_SUSPEND_COMPLETE	0x0900	This AE is generated upon completion of the requested QP Suspend operation	Transmit Suspended	QP

38.36.2.7 Steering Tag (STag) and Processing Hint Support for PE Traffic (TPH)

Refer to the section for information about how to enable TLP processing hint support.

The following table lists how the STag and processing hints are generated and how TPH operation is enabled for different types of DMA traffic associated with the PE.

Table 38-451.STag and Processing Hint Programming

Traffic Access	STag Location	PH Value[1]	Enable
CQ Element Writes	CQ Context TPH_value	GL_TPH_CTRL.Data_PH	CQ context TPH_en
CQ Doorbell Shadow Area Reads	CQ Context TPH_value	GL_TPH_CTRL.Data_PH	CQ context TPH_en
CEQ Element Writes	CEQ Context TPH_value	GL_TPH_CTRL.Data_PH	CEQ context TPH_en
SQ WQ Element Reads	QP Context SQ_TPH_value	GL_TPH_CTRL.Data_PH	QP context SQ_TPH_en
SQ Work Doorbell Shadow Area Reads	QP Context SQ_TPH_value	GL_TPH_CTRL.Data_PH	QP context SQ_TPH_en
RQ WQ Element Reads	QP Context RQ_TPH_value	GL_TPH_CTRL.Data_PH	QP context RQ_TPH_en
Data Payload Writes	QP Context RQ_TPH_value	GL_TPH_CTRL.Data_PH	QP context RCV_TPH_en
Data Payload Reads	QP Context SQ_TPH_value	GL_TPH_CTRL.Data_PH	QP context XMIT_TPH_en
UDA Packet Header Read	QP Context SQ_TPH_value	GL_TPH_CTRL.Data_PH	QP context SQ_HDR_TPH_en
UDA Packet Header Write	QP Context RQ_TPH_value	GL_TPH_CTRL.Data_PH	QP context RQ_HDR_TPH_en

Notes:

1. Default is 10b (target).

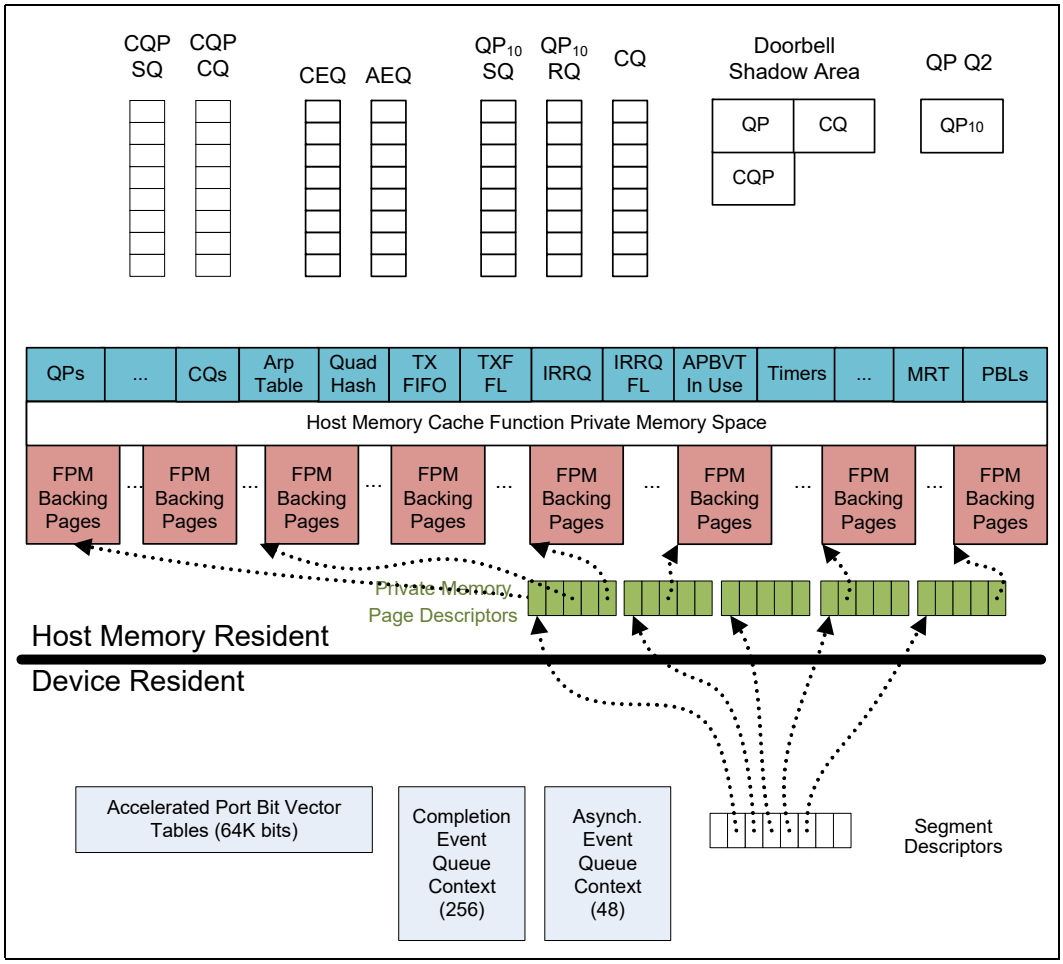
The 10 GbE controller has TPH values for CQ and CEQ in each of the various contexts. When a read is done (such as reading the CQ shadow area) the 10 GbE controller updates the context with the current TPH hint that is returned in the completion (CQ:TPH_value). If the enable bit is set in the particular context, then the 10 GbE controller should provide the TPH information on all subsequent PCI transactions associated with CQ or CEQ transactions. There is no TPH value used on AEQE writes because they are so infrequent that it is unlikely to ever get them on the correct CPU.

QP context has two values for TPH, SQ_TPH and RQ_TPH. They are independent of each other because the SQ and RQ could be on different processors in PE accelerations. They also have separate enable bits.

38.36.3 Resource Management

PE resources are managed through a combination of the HMC FPM space and Control QP operations (described in the following sections). Figure 38-127 shows the PE related data structures that are located in host memory and that reside on-die for the 10 GbE controller. Before any QP or CQ operation is attempted with CQP, the associated HMC pages must be initialized and allocated. Once all FPM page descriptor pages and backing pages have been properly allocated, initialized, and allocated to the 10 GbE controller, CQP can be used to create/modify/discard various objects necessary for PE operations. Note that Figure 38-127 only depicts memory that is visible to the 10 GbE controller. Many other software allocated data structures are allocated to track progress of QPs, CQs, etc., as shown in previous sections (Section 38.36.2.1). All of the host memory shown in Figure 38-127 must be pinned memory. QP and CQ memory including the doorbell shadow areas might be allocated from kernel or user-space virtual memory; however, all other memory shown must be allocated from the kernel in order to provide the proper level of protection between different processes accessing the 10 GbE controller concurrently. Note that any reset that affects a single PCI function also resets the HMC and require reprogramming of all HMC resources. Global resets also reset the segment descriptor programming that require re-selecting the HMC resource profile if the default profile was not in use.

Figure 38-127.10 GbE Controller and Host Memory Data Structures



Since FPM page descriptors and backing pages can be allocated on the fly in order to reduce the kernel drivers memory footprint but certain objects must be allocated at driver initialization. Table 38-452 lists the dependencies between various PE objects.

Table 38-452.PE Objects Dependencies

PE HMC Object	HMC Object Page Population Algorithm	Object Dependencies	Non-HMC Object Dependences
QP Context	Dynamic	ARP Table Entries Quad Hash Entries TX FIFO Entries APBVT In-Use Table CQ Physical Buffer List Entries if Virtually Mapped	VSI Local MAC Address L2 Tag Configuration
CQ Context	Dynamic	Physical Buffer List Entries if Virtually Mapped	
ARP Table Entries	Dynamic	None	
Quad Hash Entries	Static	None	
TX FIFO Entry Free List	Static	None	
TX FIFO Entries	Static	TX FIFO Entry Free List	
Inbound Read Request Queue Entry Free List	Static	None	
Inbound Read Request Queue Entries	Static	IRRQ Entry Free List	
Accelerated Port Bit Vector Table In-Use Table	Static	None	
Memory Registration Table	Dynamic	None	
Physical Buffer List Entries	Dynamic	None	

All PE objects marked as static in Table 38-452 must be allocated at driver initialization time. Entries marked as dynamic may be allocated and de-allocated on an as needed basis. Objects shown as dependencies must be populated and allocated via CQP (along with the HMC FPM memory for the object itself) prior to issuing a CQP operation to manipulate the PE object. More information regarding the object dependencies is available in the individual CQP operations in following sections.

Note that PF resets clears the PE state including any HMC configuration settings for the PF and all associated VFs. VF resets clear all PE state and HMC configuration settings only for the VF if the VF is enabled for PE accelerations.

Dynamic objects of the same type might share a cache line. However, a dynamic object must not share any cache lines with another object that has a different type.

38.36.3.1 PE Initialization

Table 38-453 lists the high-level steps required to initialize a PE. Table 38-454 and Table 38-455 list the high-level steps required to bring an RDMA or UDA QP to the operational state for the 10 GbE controller. This list is not meant to be exhaustive but more as a guide to show one ordered list of a typical initialization flow. It is assumed that the LAN initialization of the VSI has enabled PE operation and has the LAN queues operational for the PF.

Table 38-453.PE Resource Initialization

Step	Resource	Responsible Software Component	Action(s)	Notes
1	Determine if the PE is enabled	Product Name driver.	Read the GLGEN_PE_ENA.PE_ENA CSR bit.	
2	Determine the IP and MAC address for the associated LAN queue pool	Product Name driver.	Each PE interface is associated with a specific LAN interface, The mechanism to find the interface information is operating system specific.	
3	Enable RDMA/UDA for the corresponding VSI	Product Name driver	Each PE interface is associated with a specific VSI. Set PETCP_ENA, PEUUDP_ENA and PEMUDP_ENA bits in VSIQFCTL VSI register.	
4	Enable Local VSI Loopback	Product Name driver.	PE enables communication between QPs associated with the same VSI. Set ALLOWLOOPBACK and ALLOWLOCALLOOPBACK bits in VSISWID register.	This step is not required. The PE uses SWPE to perform loopback.
5	Setup the PE portion of the HMC	Product Name driver.	Wait for PE firmware to be initialized by polling GLPE_CPUSTATUS0 is set to 0x80. First the HMC base and size registers must be programmed for this PF. Subsequently appropriate backing pages must be initialized.	Refer to the details on initializing the HMC.
6	Acquire one or more MSI-X vectors for the PE	Product Name driver.	MSI-X vectors can be shared between LAN and protocol operation or can be partitioned to enable independent operation	See the previous section on LAN initialization for more information on MSI-X vector initialization
7	Initialize the PE control QP	Product Name driver.	Allocate memory for the CQP SQ and CQP context and write to the PECCQPHIGH and PECCQPLOW registers.	See Section 38.36.3.2.1 for more details.
8	Initialize the PE CQ 0	Product Name driver.	Allocate memory for the CQ and issue a create CQ operation to CQP.	See Section 38.36.3.3.3 for details on the Create CQ CQP operation.
9	Initialize the PE CEQ 0	Product Name driver.	Allocate memory for the CEQ and issue a create CEQ operation to CQP, The CEQ is associated with a Product Name interrupt.	See Section 38.36.3.4.2 for details on the create CEQ CQP operation.
10	Initialize the PE AEQ	Product Name driver.	Allocate memory for the AEQ and issue a create CEQ operation to CQP,.	See Section 38.36.3.5 for details on the create AEQ CQP operation.
11	Allocate a local MAC address table entry for the PE	Product Name driver.	Request and initialize a PE MAC address table entry and initialize the entry.	See Section 38.36.3.3.6 and Section 38.36.3.3.7 for the allocate and manage local MAC address table entry CQP operations.

The following table lists steps required to bring RDMA QP to the operational state. The majority of steps are similar to initialization of RDMA QP. PE initialization steps are common for RDMA and UDA traffic and should be done only once.

Table 38-454.RDMA Resource Initialization

Step	Resource	Responsible Software Component	Action(s)	Notes
1	Create a Protection Domain	Product Name driver on behalf of a verbs application.	Reserve a doorbell page from the PE doorbell pages in the Product Name BAR.	See Section 38.36.2.1.6 for more information on doorbell pages.
2	Allocate a CQ	Product Name driver on behalf of a verbs application.	Allocate memory for the CQ and issue a create CQ operation to CQP, The CQ will be associated with the CEQ from step 7.	See Section 38.36.3.3.3 for details on the create CQ CQP operation.
3	Allocate a QP	Product Name driver on behalf of a verbs application.	Allocate memory for the QPs SQ and RQ and issue a Create QP operation to CQP.	See Section 38.36.3.3.2 for details on the create QP CQP operation.
4	Allocate Memory Regions	Product Name driver on behalf of a verbs application.	Allocate physical buffer list entry ranges for the page list that backs the memory provided by the application and then issue register memory region CQP operation.	See Section 38.36.3.3.4 for more information on the register memory region operation and also Section 38.36.2.1.4 for an overview of the 10 GbE controller's memory registration capabilities.
5	Post Work Requests to the QPs RQ	Product Name driver on behalf of a verbs application.	Fill in a RQ work request with the information supplied by the application and submit the work.	See Section 38.36.2.1.4 for more information on RQ operation.
6	Post Work Requests to the QPs RQ	Product Name driver on behalf of a verbs application.	Fill in a SQ work request with the information supplied by the application and submit the work.	See Section 38.36.4.5.1 for information about the RQ work request format and Section 38.36.2.1.7 for more information on RQ operation.
7	Process Interrupts and the CEQ (optional)	Product Name driver.	Once an interrupt occurs for a CEQ, the CQ can be determined from the CEQ Entry. Events are generated to that application to let the application know that an event for a specific CQ has been received.	See Section 38.36.2.1.2 for more information on processing CEQ entries.
8	Process the CQ (optional)	Product Name driver on behalf of a verbs application.	One the application has either receive a completion event or is constantly polling the CQ, CQ entries are processed by the application.	See Section 38.36.2.1.3 for more information on processing CQ entries.

The following table lists steps required to bring UDA QP to the operational state. PE initialization steps are common for RDMA and UDA traffic and should be done only once.

Table 38-455.UDA Resource Initialization (Sheet 1 of 2)

Step	Resource	Responsible Software Component	Action(s)	Notes
1	Create a Protection Domain	Product Name driver on behalf of a verbs application.	Reserve a doorbell page from the PE doorbell pages in the Product Name BAR.	See Section 38.36.2.1.6 for more information on doorbell pages.
2	Allocate a CQ	Product Name driver on behalf of a verbs application.	Allocate memory for the CQ and issue a create CQ operation to CQP. The CQ is associated with the CEQ from step 7.	See Section 38.36.3.3.3 for details on the create CQ CQP operation.
3	Allocate a QP	Product Name driver on behalf of a verbs application.	Allocate memory for the QPs SQ and RQ and issue a create QP operation to CQP.	See Section 38.36.3.3.2 for details on the create QP CQP operation.
4	Allocate Memory Regions	Product Name driver on behalf of a verbs application.	Allocate PBL entry ranges for the page list that backs the memory provided by the application and then issue register memory region CQP operation.	See Section 38.36.3.3.4 for more information on the register memory region operation and alsoSection 38.36.2.1.4 for an overview of the 10 GbE controller's memory registration capabilities.
5	Allocate Address Vector (optional)	Product Name driver on behalf of a verbs application.	For each destination Node issue a create address vector operation to CQP. Allocated address vector is used to generate Ethernet/IP headers.	See Section 38.36.3.5.3 for more information on the create address vector operation as well as an overview of the 10 GbE controller's address vector capabilities.
6	Bind to Local Port (optional)	Product Name driver on behalf of a verbs application.	For each local port allocated for UDP traffic issue a management quad hash entry operation to CQP. This operation will enable forwarding of unicast packet to the QP.	See Section 38.36.3.5.7 for more information on the manage quad hash operation.
7	Post Work Requests to the QPs RQ	Product Name driver on behalf of a verbs application.	Fill in a RQ work request with the information supplied by the application and submit the work.	See Section 38.36.2.1.4 for more information on RQ operation.
8	Post Work Requests to the QPs RQ	Product Name driver on behalf of a verbs application.	Fill in a SQ work request with the information supplied by the application and submit the work.	See Section 38.36.4.5.1 for information about the RQ work request format and Section 38.36.2.1.7 for more information on RQ operation.
9	Process Interrupts and the CEQ (optional)	Product Name driver.	Once an interrupt occurs for a CEQ, the CQ can be determined from the CEQ Entry. Events are generated to that application to let the application know that an event for a specific CQ has been received	See Section 38.36.2.1.2 for more information on processing CEQ entries.

Table 38-455.UDA Resource Initialization (Sheet 2 of 2)

Step	Resource	Responsible Software Component	Action(s)	Notes
10	Process the CQ (optional)	Product Name driver on behalf of a verbs application.	One the application has either receive a completion event or is constantly polling the CQ, CQ entries are processed by the application.	See Section 38.36.2.1.3 for more information on processing CQ entries.

38.36.3.2 Control CQP Operation

CQP operations are modeled after verbs SQ operations as far as the work submission process. Some differences exist for CQP when compared to a typical PE QP:

- CQP QP context is located on-die instead of in the HMC.
- The doorbell register used for work submission is PFPE_CQPDB instead of PFPE_WQEALLOC.
- PFPE_CQPDB take SQ head as input and there is no QP ID specified since each PE enabled PCI function has a dedicated CQP.
- There is no doorbell shadow area associated with CQP.
- Hardware tail can be read from the PFPE_CQPTAIL, but in general software can just track SQ tail based on completion processing from CQ0.

The operations defined for CQP are listed in Table 38-459. CQ0 is the CQ associated with the CQP for each function. All other CQs are available for accelerated PE use.

38.36.3.2.1 Creating the CQP

CQP is created by first allocating memory for CQP context shown in Section 38.36.3.2.3 and initializing context. The HMC resource profile is also able to be selected during the CQP creation process. Note that HMC resource profile is selected during first CQP creation on a device and is then locked. Subsequent attempts to change the HMC resource profile are ignored until all CQPs created on the device have been destroyed. The resource profile that has been selected is reported in the PFPE_CCQPSTATUS register after CQP has been successfully created. Additionally the SQ memory and FPM configuration buffer must also be allocated. This memory must be pinned and the physical address must be determined. Once software has filled in the CQP context properly, the physical address of CQP context is written to the PFPE_CCQPHIGH and PFPE_CCQPLOW registers to notify the 10 GbE controller of the create CQP operation. Both registers must be written and the PFPE_CCQPLOW register must be written last. Product Name then fetches the context and create CQP for the current PCI function. Once CQP has be successfully created, the CCQP_DONE bit in the PFPE_CCQPSTATUS register is set to 1b. If an error was encountered while attempting to create CQP, the CCQP_ERR bit is set instead of the CCQP_DONE bit. The major and minor error codes are available in the PFPE_CQPERRCODES register. The next six operations necessary to get CQP to a functional state are to:

1. Query FPM values.
2. Commit FPM values.
3. Initialize the HMC resources for CQ0.
4. Create CQ 0.
5. Create CEQ.
6. Create the AEQ.

The operations must be done in this order or unexpected results might be observed. In addition to the CQ0 HMC backing pages initialized in step 3, if either the CQ or CEQ need to be virtually mapped, the PBLE HMC object backing pages must also be populated before creating the CQ and/or CEQ. Protocol enabled VFs use the VFPE_* versions of the registers previously mentioned. These steps involve issuing standard CQP operations with the following exceptions:

- PFPE_CQPTAIL must be polled until the create CQ operation has been submitted to CQP and only a single CQP operation must be submitted at a time to CQP while polling CQP Tail. Any errors encountered via CQP is reported via the CQP_ERR bit in the PFPE_CQPTAIL register. The major and minor error codes are available in the PFPE_CQPERRCODES register.
- CQ0 can be polled for the completion of the create CQ operation. No events can be generated since there is no CEQ created at the point when CQ0 is created.

One additional CQP operation must be issued anytime after step 3 but before any iWARP QPs can be created. The static HMC pages allocated operation must be submitted after all static HMC objects have been allocated and configured for the PCI function. See Table 38-452 for the list of statically configured HMC objects.

38.36.3.2.2 Destroying the CQP

Once all PE resources have been destroyed and deallocated, the control QP related resources much also be destroyed. The flow is the following:

1. Issue a CQP operation to discard the AEQ.
2. Issue a CQP operation to discard the CEQ(s).
3. Poll CQ0 to verify these operations completed.
4. Discard CQP by writing the PFPE_CCQPHIGH and PFPE_CCQPLOW registers with 0b.
5. Poll the PFPE_CCQPSTATUS register until it changes to 0b.
6. Discard the CQ0 HMC resources to ensure that no further accesses to host memory is performed.

Once CQP has been destroyed, CQ0 will also have been destroyed as well.

38.36.3.2.3 CQP Context

CQP context describes the SQ for the control QP.

Table 38-456.CQP Context Format

Byte Offset	[Bit Range]Field Name			
0	[63:40] [39:32] [31:24] [23:12]	Reserved num_CEQs_per_VF StructVersion Reserved	[11:08] [7:2] [1] [0]	SQ_Size Reserved Disable_FPDU_Packing Enable_Fine_Grained_Timers
8	[63:9]	SQ_base	[8:0]	Reserved
16	[63:38] [37:32]	RSVD PEEnabledVfCount	[31:3] [2:0]	RSVD HMCProfileType
24	[63:0]	QP_Completion_Context		
32	[63:0]	Reserved		
40	[63:0]	Reserved		
48	[63:0]	Reserved		
56	[63:0]	Reserved		

StructVersion (8)

Used to specify changes in CQP context format. The only value defined is 0.

Enable_Fine_Grained_Timers (1)

This bit enables fine grained TCP timer mode for the PCI function associated with the CQP to be created. Fine grained TCP timer mode changes the PE TCP timers from normal values (2 minutes to ~100 ms) to a more aggressive set of values (~1s to ~100 µs). Fine grained TCP timer mode is useful for environments with low packet congestion and/or low packet loss.

SQ_Size (4)

This field encodes the maximum number of work requests for each WQ. The encoding of the SQ sizes are $4*2^{SQ\_Size}$ in terms of 32-byte quanta of memory. Since each CQP WQE takes 64 bytes (as if it consumed an additional fragment descriptor) only the following settings are valid for CQP (all other settings are reserved and might cause unexpected results):

1 = 4 WQEs
2 = 8 WQEs
3 = 16 WQEs
4 = 32 WQEs
5 = 64 WQEs
6 = 128 WQEs
7 = 256 WQEs
8 = 512 WQEs
9 = 1024 WQEs
10 = 2048 WQEs

Software can only submit N-1 WQEs to a WQ without processing completions for the WQ without exposing the possibility of a WQ overflow. WQ overflow results in indeterministic behavior for the affected WQ.

SQ_base (55)

SQ_base is the most significant bits of the physical address of the SQ for the control QP. The WQ base address must be evenly divisible by 512 for proper behavior.

HMCProfileType (3)

This field specifies the HMC resource profile to be activated.

0: Reserved

1: Default

2: SR-IOV VF Primary

3: SR-IOV Even Distribution

7: Reserved

Specifying a reserved HMC resource profile causes the current HMC resource profile to remain active. This field is ignored for VFs.

PEEnabledVfCount (6)

Only valid when HMCProfileType is one of the SR-IOV profile types. Specifies the number of PE enabled VFs to be allocated. Values larger than 32 are rounded down to 32. A value of zero forces the default profile to be selected. This field is ignored for VFs.

QP_Completion_Context

QP_Completion_Context is reported in every CQ Entry related to the CQP.

Disable_FPDU_Packing (1)

num_CEQs_per_VF (8)

This specifies the max CEQs needed for each VF. The purpose of this field is to limit the number of CEQs a VF uses in the even distribution profile in order to leave enough CEQs for the PF. A value of zero preserves the behavior of evenly allocating the CEQs across all functions. This field is valid only if SRIOV is enabled and the balanced profile is in use.

38.36.3.2.4 CQP Error Codes

CQP can return a number of error and status code the operations that it performs. Table 38-457 lists these error and status codes that can be returned in the CQ entry listed in Table 38-458.

Table 38-457.CQP Error Codes (Sheet 1 of 2)

Major Error Code	Minor Error Code	Completion Reason	Description
0x0000	0xKK00	STag Valid	The STag queried is in the valid state, KK is the STag key.
0x0000	0x0001	STag Invalid	The STag queried is in the invalid state.
0x0000	0x0002	RQ WQE Flushed	A RQ WQE was flushed as a result of the flush WQEs operation.
0x0000	0x0003	SQ WQE Flushed	A SQ WQE was flushed as a result of the flush WQEs operation.
0x0000	0x0004	SQ and RQ WQEs Flushed	Both a RQ WQE and a SQ WQE were flushed as a result of the flush WQEs operation.
0x0000	0x0005	Suspend Pending	A suspend QP WQE was issued referencing a busy QP. An AE_QP_SUSPEND_COMPLETE AE is generated when the QP has been successfully suspended.
0xFFFF	0x3000	Packet Count Error	A packet count issue has been detected on a QP that is either being destroyed or uploaded. The state of the QP is indeterminate once this completion code has been observed.
0xFFFF	0x3001	Scratchpad Flush Error	CQP has timed out waiting for TEP to flush scratch pads for a QP that is getting destroyed or uploaded. This error has never been observed and is not expected.
0xFFFF	0x4000	Memory Window Bound	The discard QP operation failed because there are still MWs bound to the QP.
0xFFFF	0x6000	Insufficient Resources (General)	There is an insufficient amount of the resources needed to complete this operation.
0xFFFF	0x6001	Insufficient FLM Resources	A create QP was attempted but there were no available transmit of Q1 FIFO entries.
0xFFFF	0x6002	Insufficient Doorbell Resources	Returned by commit FPM (if requested number of QPs or CQs is greater than count allotted in partition registers).
0xFFFF	0x7000	Bad IP/MAC Table Index	The MAC_IP table index given is either invalid or is not owned by this function.
0xFFFF	0x8000	QPID Too Big	The QPID specified on the operation is too large for the on board memory configuration.
0xFFFF	0x8002	Invalid State	The operation requested is not valid for this resource based upon its current state.
0xFFFF	0x8003	Invalid Next QP State	The next state specified on the Modify QP operation is not valid based on the QP's current state.
0xFFFF	0x8004	Invalid Queue Size	The SQ, RQ, or Q1 is too large or to small.
0xFFFF	0x8005	Invalid QP Type	The operation is not valid for this type of QP.

Table 38-457.CQP Error Codes (Sheet 2 of 2)

Major Error Code	Minor Error Code	Completion Reason	Description
0xFFFF	0x8006	No WQE Pending	A flush operation was attempted but there was not a WQE pending.
0xFFFF	0x8007	Bad Close	A Modify QP from RTS->Closing was attempted while there was SQ and/or Q1 activity in progress.
0xFFFF	0x8009	LLP Closed	Returned by Modify QP (next iWARP state = Terminate if the TCP state is already closed or time wait).
0xFFFF	0x800A	Reset Not Sent	Returned by Modify QP (if the TCP state is already closed).
0xFFFF	0x800C	SD Index Out of Range	An attempt has been made to update a VF's page descriptor using an SD that does not reference a PBLE object.
0xFFFF	0x800D	PD Index Out of Range	An attempt has been made to update a VF's page descriptor using a SD and PD index that does not reference a PBLE object
0xFFFF	0x800E	PD Page Boundary Exceeded	An attempt has been made to update a VF's page descriptor but the request has crossed from one PD page to the next PD page.
0xFFFF	0x800F	SD Boundary Exceeded	Returned by commit FPM (if requested number of HMC objects crosses SD boundary for that function).
0xFFFF	0x8010	Invalid Function Type	The operation is not valid for the PCI function type that attempted to issue the operation.
0xFFFF	0x8011	FLM Not Initialized	Returned by create QP if an attempt to create a QP has been attempted before the static resources allocated WQE has been issued.
0xFFFF	0x8012	Invalid ECN Codepoint	An attempt was made to create a QP with an invalid initial ECN code-point value.
0xFFFF	0x8013	Invalid VF Number	A PF attempted to access resources associated with a VF that it does not own.
0xFFFF	0x8014	Invalid MRTE Index	An attempt was made to access a reserved STag.
0xFFFF	0x8015	Invalid Terminate Message Size	A terminate was requested with a Term_length value that is too small.
0xFFFF	0x8016	Bad Send MSS value	The value given for SndMSS is invalid.
0xFFFF	0x80EE	Unsupported Opcode	The opcode given is unsupported.
0xFFFF	0x80EF	WQE Not Valid	The WQE fetched due to a CQP doorbell ring is not valid.
0xFFFF	0xFFFF	Error	CQP encountered an error during the processing of this WQE.

38.36.3.2.5 CQP CQE Format

The CQP CQ entry is listed in Table 38-458. Every work request processed by the control QP returns a completion on CQ 0 (CQ_ID 0). If CQ0 is created with the Avoid_Memory_Conflicts bit set, then an additional 32 bytes of padding is added to each CQE for a total of 64 bytes per CQE. This doubles the size of the CQ in host memory.

Table 38-458.CQP CQ Entry Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	RSVD		
8	[63:0]	QP_Completion_Context		
16	[63:32]	RSVD	[31:0]	Operation_Return_Value
24	[63] [62] [61:56] [55]	CQE_Valid SQ RSVD Error	[54:46] [45:32] [31:16] [15:0]	RSVD WQ_Desc_Index Major Error Code Minor Error Code

Op (6)

This field reports the opcode from the operation associated with the CQE.

QP_Completion_Context (64)

This field is transferred to the CQE from CQP QP context listed in Table 38-456.

WQ_Desc_Index (14)

WQ are sliced up into 32-byte descriptor quanta. Every WQE must start with a 32-byte descriptor on a 32-byte boundary. WQ_Desc_Index reports the 32-byte quanta index of the WQE associated with the completion.

QP_ID (18)

QP associated with the completed message.

Operation_Return_Value (32)

This field is filled in by CiP with operation dependent return values such as the reachability time stamp or the MAC and IP table Index.

CQE_Valid (1)

The CQE_Valid bit for CQE is a bit that indicates that a CQE is ready to be processed. The polarity of the valid bit changes each time the CQ wraps from the last entry back to the first entry. This change in polarity reduces software overhead by avoiding the need to clear the valid bit once software has processed a valid CQE. Software is responsible to clear (set to 0b) all memory in a CQ initially at CQ creation. The first iteration (and subsequent odd numbered iterations) through the CQ, the 10 GbE controller sets the Valid bit to 1b when it writes a new CQE. For the second iteration (and all even numbered iterations) through the CQ, the 10 GbE controller sets the valid bit to a 0b when it writes an new CQE.

Error (1)

The *Error* bit indicates if there was a problem with the operation specified by the WQE associated with the WR that is associated with completion.

0 = No error is being reported with this completion.

1 = An error occurred when processing the WQE associated with this CQE and that the *Error Code* field is valid

SQ (1)

The SQ bits is used to determine if the completion is associated with a SQ (SQ) or a RQ (RQ).

0 = RQ

1 = SQ

Major error code (16) and minor error code (16)

The *Error Code* fields are valid if the *Error* bit is set or if the operation submitted was query STag. See Table 38-457 for the values that are returned for CQP operations.

38.36.3.3 CQP SQ Descriptor Formats

The following WQE formats are used in conjunction with the Control QP (CQP) to manage internal PE structures that the 10 GbE controller uses to communicate with host software. Operations that are supported for CQPs are the following.

Table 38-459.CQP Operations

Operation Code	Operation Name	Page	Operation Code	Operation Name	Page
0x00	Create QP	3196	0x17	RSVD	
0x01	Modify QP	3196	0x18	Discard CEQ	3214
0x02	Discard QP	3196	0x19	Create AEQ	3216
0x03	Create CQ	3200	0x1A	RSVD	
0x04	Modify CQ	3200	0x1B	Discard AEQ	3216
0x05	Discard CQ	3200	0x1C		
0x06			0x1D		
0x07			0x1E		
0x08			0x1F	Update PE SDs	3217
0x09	Allocate STag	3203	0x20	Query FPM Values	3218
0x0A	Register MR	3203	0x21	Commit FPM Values	3221
0x0B	Query STag	3206	0x22	Flush WQEs	3222
0x0C	Register Shared MR	3203	0x23	Manage APBV Table Entry	3224
0x0D	Deallocate STag	3203	0x24	NOP	3224
0x0E	Manage Local MAC Table	3208	0x25	Manage Quad Hash Table Entry	3225
0x0F	Manage ARP Cache		0x26		
0x10	Manage VF PBLE Backing Pages	3210	0x27		
0x11	Manage Push Page	3211	0x28		
0x12	RSVD		0x29	Suspend QP	3228
0x13	Upload context	3212	0x2A	Resume QP	3228
0x14	Allocate Local MAC Entry	3207	0x2B	Static HMC Pages Allocated	3229
0x15	Manage HMC PM Function Table	3214	0x2C	RSVD	
0x16	Create CEQ	3214	0x2D	RSVD	
			0x2C - 0x3F	RSVD	

38.36.3.3.1 Common CQP Descriptor Format Fields

The basic CQP WQE is a 64 byte structure that is broken up into 64 bit (8-byte) words. The placement of these fields within the WQE are common among all CQP WQEs and also iWARP WQEs. Table 38-460 lists the basic structure of a CQP WQE including the common fields. The definition of fields marked as operation code dependent vary from operation-to-operation and are detailed in subsequent sections.

Table 38-460.CQP Common WQE Fields

Byte Offset	[Bit Range]Field Name		
0	[63:0]	Operation Code Dependent	
8	[63:0]	Operation Code Dependent	
16	[63:0]	Operation Code Dependent	
24	[63] [62:41] [40:38]	WQE_Valid Operation Code Dependent RSVD (AdditionalFragmentCount)	[37:32] OP [31:0] Operation Code Dependent
32	[63:0]	Operation Code Dependent	
40	[63:0]	Operation Code Dependent	
48	[63:0]	Operation Code Dependent	
56	[63:0]	Operation Code Dependent	

OP (6)

CQP operation code for create QP, modify QP, or discard QP. See Table 38-459 for the specific values.

WQE_Valid (1)

The WQE_Valid bit for WQ Entries (WQE) is a bit that indicates that a WQE is ready to be processed by the 10 GbE controller. The polarity of the valid bit changes each time the WQ wraps from the last entry back to the first entry. This change in polarity reduces software overhead associated with the need to clear a valid bit and also to enable the 10 GbE controller to read ahead in the WQ to reduce the need for doorbell rings. See Section 38.36.2.1.5 for more information on submitting work to a QP with the 10 GbE controller. Software is responsible to clear (set to 0b) all memory in a WQ initially at QP creation. The first iteration (and subsequent odd numbered iterations) through the WQ, software sets the Valid bit to 1b when it writes a new WQE. For the second iteration (and all even numbered iterations) through the WQ, software sets the valid bit to a 0b when it writes an new WQE.

38.36.3.3.2 Create/Modify/Discard QP Descriptor Format

This WQE format is used by host software to manage QPs. The various valid bits in this format are designed to reduce the load on CQP for determining what has changed. If a valid bit is set for a particular field, then CQP updates the value in the 10 GbE controller's context, else it does not. The only valid operation for QP type CQP is the discard operation. See Section 38.36.3.2.1 for more details on created CQP.

Note that software is responsible for populating the appropriate HMC objects and have having created objects for the HMC and through CQP that the QP depends on.

Table 38-461.CQP Create/Modify/Discard QP WQE Format (Sheet 1 of 2)

Byte Offset	[Bit Range]Field Name			
0	[63:0]	RSVD		
8	[63:52] [51:48] [47:46]	RSVD Term_Length[1] RSVD	[45:32] [31:00]	RSVD RSVD
16	[63:0]	QP Context Address[1]		

Table 38-461.CQP Create/Modify/Discard QP WQE Format (Sheet 2 of 2)

Byte Offset	[Bit Range]Field Name			
24	[63]	WQE_Valid	[46]	Force_Loopback[3]
	[62:60]	Next iWARP State[1]	[45]	Virtual_WQs[3]
	[59]	ARP Table Index Valid[1]	[44]	Cached_Variables_Valid[1]
	[58]	Reset Connection[1]	[43]	TOE_Context_Valid[3]
	[57:56]	Terminate_Actions[1]	[42]	ORD_Valid[1]
	[55]	Remove_Hash_Entry	[41]	RSVD
	[54]	Ignore MW Bound[2]	[40:38]	RSVD
	[53]	RSVD	(AdditionalFragmentCount)	
	[52]	RSVD	[37:32]	OP
	[51]	RSVD	[31:18]	Reserved
	[50:48]	QP_Type	[17:0]	QP_ID
	[47]	CQ_Numbers_Valid[1]		
32	[63:0]	RSVD		
40	[63:7]	Doorbell Shadow Address[3]	[6:0]	RSVD
48	[63:0]	RSVD		
56	[63:0]	RSVD		

Notes:

1. Not used for discard QP operations.
2. Only used for discard QP operations.
3. Valid for create and modify QP operations as long as the iWARP state is not already RTS, error, or terminate.

OP (6) and WQE_Valid

See Section 38.36.3.3.1.

ARP table index valid (1)

If set (1b) ARP cache index has been updated in the QP context. Else (0b) CQP must ignore ARP *Cache Index* field in QP context.

TOE_Context_Valid (1)

If set (1b) the TOE portion of QP context is valid, Else (0b) ignore the TOE portion of QP context. This bit is useful if an iWARP QP is created that does not yet have a TOE connection associated with it. Host software can create the SQ/RQ etc, and then later issue a modify QP to update the TOE context, updating TOE context is only allowed prior to setting the QP iWARP state to RTS for iWARP QPs and prior to setting the QP TOE state to established for TOE QPs.

ORD_Valid (1)

If set (1b) ORD has been updated in QP context. Else (0b) CQP must ignore ORD field in QP context.

Virtual_WQs (1)

If set (1b) the SQ and RQ for this QP are virtually mapped, if clear the SQ and RQ are physically mapped. When a QP is using virtual WQs, the RQ and SQ base address in QP context point to the first HMC FPM space PBLE index for the page table for each WQ. Host software is responsible for populating the PBLEs with the page list associated with each WQ before making the QP active (Modify QP with next state of RTS). Both the SQ and the RQ buffers must be aligned to a multiple of 4 KB in host memory when Virtual_WQs is set. This field is only valid for QPs in the IDLE state or during the IDLE to RTS transition.

Remove_Hash_Entry (1)

Only valid with modify and discard QP opcodes. If set (1b) then the hash table entry associated with this QP is removed from the hash list. This bit has no effect when clear (0b).

CQ_Numbers_Valid (1)

If set (1b) the values for RxCmpQueueNum and TxCmpQueueNum in QP context is used to update internal context, else RxCmpQueueNum and TxCmpQueueNum in QP context are ignored.

Force_Loopback (1)

If set (1b) the traffic for the QP is internally looped back within the VSI associated with the QP. If clear (0b), the traffic is switched normally unless the source and destination IP addresses and MAC addresses match. This field is used to enable loopback traffic for different IP addresses with a single VSI.

QP_Type (3)

QP_Type indicates the type of QP for CQP to create. The valid QP types are CQP (0, only valid for DestroyQP), iWarp (1), or UDA (2).

MSS_Change (1)

If set (1b) the MSS value for the connection has changed. The 10 GbE controller starts using the new MSS value (found in the New MSS Value) immediately which might temporarily cause partial FPDUs to be generated for any RDMA messages that need re-transmission. Only valid for Modify QP operations on QPs with the TCP state = ESTABLISHED.

New MSS value (14)

If MSS_Change is set (1b) this MSS value is used for subsequent outbound packets on this QP. Note that reduction in the MSS might cause fragmentation for retransmitted packets. Only valid for Modify QP operations on QPs with the TCP state = ESTABLISHED.

Ignore MW bound (1) - This field is ignored

This field is only meaningful for discard QP for iWARP QPs. If set (1b) the QP is destroyed even if memory windows are still bound to the QP. If clear (0b) then the discard fails with a major/minor code of 0xFFFF/0x4000 when memory windows are still bound to the QP.

Terminate_Actions (2)

Only valid for Modify QP for iWARP QPs with Next iWARP State set to terminate. Values for this field are:

0 = Send both Terminate and FIN

1 = Send Terminate Only

2 = Send FIN only

3 = Do not send Terminate or FIN

Term_Length (4)

Only valid with modify opcodes when the next iWARP state is set to 5 (terminate) and Terminate_Actions is set to 0b or 1b. This field is ignored in all other cases. This field specifies the number of 4-byte Dwords to transfer from the Q2 outbound terminate data area when a terminate message is requested. The Q2 area starts with a Dword specifying the *Terminate Control* field as specified in the RDMAP RFC. A maximum of 13 Dwords is transmitted (terminate control Dword plus 48 bytes of data for an RDMA read header). Values greater than 13 results in 13 Dwords being transmitted.

Cached_Variables_Valid (1)

Only valid with modify opcodes, If set (1b) then only the cached variables associated with this QP is updated on the modify operation. This bit has no effect when clear (0b). See Table 38-493 for the list of variable that are cached. Each cached variable is marked with a table footnote.

Next iWARP state (3)

Only valid on Modify QP opcodes for iWARP QPs. The iWARP state definitions are:

0 = NON EXISTANT

1 = IDLE

2 = RTS

3 = CLOSING

4= reserved

5 = TERMINATE (iWARP only)

6 = ERROR

7 = Reserved

Reset connection (1)

Only valid with Modify QP opcodes. If set (1b) then a TCP reset is sent out if the TCP state for the QP has already been ESTABLISHED is not TIME_WAIT or CLOSED. If the QP TCP state is already TIME_WAIT or CLOSED, an AE_RESET_NOT_SENT AE is generated.

QP_ID (18)

QP_ID identifies the QP that is to be acted upon by the 10 GbE controller.

QP context address (64)

A physically mapped pointer in host memory that contains QP context. The format of the QP context structures are define in Section 38.36.4.2, Section 38.36.3.2.3. This buffer must be aligned to a multiple of 4 bytes.

Doorbell shadow address (58)

A physically mapped pointer in host memory that hardware writes to when forward progress has been made on SQ work. The format of the doorbell shadow area is listed in Table 38-463. This buffer must be aligned to a multiple of cache-line size bytes.

Modify QP processing performed by CQP for the 10 GbE controller is not designed to be verbs compliant from the point of view of which iWARP state transitions are allowed. The 10 GbE controller enables invalid state transitions to enable host hardware to make state transitions to ERROR or TERMINATE autonomously in order to prevent subsequent work from being processed, but still enable host software the ability to provide verbs compliant behavior to the application.

Additionally, the 10 GbE controller is not providing support for the transition back to IDLE. It is expected that host software destroys QPs and recreate them when the transition from non-IDLE to IDLE is desired. Table 38-462 describes the transitions allowed by the 10 GbE controller.

Table 38-462.Product Name iWARP QP State Transitions

Current iWARP State	Next iWARP State				
	Idle	RTS	Closing	Terminate	Error
Idle	Yes	Yes	No	No	Yes
RTS	No	Yes	Yes	Yes	Yes
Closing	No	No	No	No	Yes
Terminate	No	No	No	Yes	Yes
Error	No (No error reported)	No	No	No	Yes

Table 38-463.QP Doorbell Shadow Area

Byte Offset	[Bit Range]Field Name			
0	[63:14]	RSVD	[13:0]	HW_SQ_Tail
8	[63:0]	RSVD		
16	[63:0]	RSVD		
24	[63:0]	RSVD		
32	[63:0]	RSVD		
40	[63:0]	RSVD		
48	[63:0]	RSVD		
56	[63:0]	RSVD		

For information on the doorbell ringing algorithm and usage of the QP doorbell shadow area, see Section 38.36.2.1.7.

38.36.3.3.3 Create/Modify/Discard CQ Descriptor Format

Host software uses the following structure to manage CQs through the control QP. CQs are comprised of a packed array of CQEs, which are WQ type specific that are managed as a circular queue. They can be either virtually or physically contiguous memory buffers. Further discussion of CQ operation is found in Section 38.36.2.1.3.

Table 38-464.CQP Create/Modify/Discard CQ WQE Format (Sheet 1 of 2)

Byte Offset	[Bit Range]Field Name			
0	[63:20]	RSVD	[19:00]	cq_size[1]
8	[63]	RSVD	[62:0]	CQ_Context_Value
16	[63:18]	RSVD	[17:00]	cq_shadow_read_threshold[2]
24	[63] [62] [61] [60] [59:50] [49] [48] [47] [46]	WQE_Valid RSVD Avoid_Memory_Conflicts[2] TPH_en RSVD CEQ_ID_Valid enable_ceqe_mask[2] Virtually Mapped[1] Check_Overflow[2]	[45:44] [43] [42:38] [37:32] [31] [30:24] [23:17] [16:0]	Leaf_PBL_Size[1] CQ_Resize RSVD OP RSVD CEQ_ID RSVD CQ_ID
32	[63:8]	Physical_Buffer_Address[1]	[7:0]	RSVD
40	[63:6]	Doorbell Shadow Address	[5:0]	RSVD

Table 38-464.CQP Create/Modify/Discard CQ WQE Format (Sheet 2 of 2)

Byte Offset	[Bit Range]Field Name			
48	[63:28]	RSVD	[27:0]	first_pm_pbl_index[1]
56	[63:8]	RSVD	[7:0]	TPH_Value

Notes:

1. Valid on modify operations with the CQ_Resize bit set or on create operations.
2. Only valid on create operations.

OP (6) and WQE_Valid

See Section 38.36.3.3.1.

enable_ceqe_mask (1)

If clear (0b), the 10 GbE controller potentially generates multiple CEQEs per CQ if a request for completion notification is generated by software before a CEQE is consumed by software. This behavior can lead to CEQ overflows. If set (1b), the 10 GbE controller only allows one CEQE to be generated per CQ. Subsequent CEQE generation is enabled by writing the CQ ACK register.

CEQ_ID (7)

CEQ_ID specifies the CEQ to associated with the CQ specified by CQ_ID. This field is only meaningful when CEQ_ID_Valid is set (1b).

CEQ_ID_Valid (1)

If set (1b), the 10 GbE controller updates the CEQ for the CQ; else the CEQ_ID field is ignored. Changing the CEQ_ID for an active CQ is allowed to enable software to redistribute CQ event processing between different processors. Each CEQ for a given function must be assigned to independent MSI-X vectors in order to take advantage of this capability.

CQ_Resize (1)

If set (1b), CQ_Resize a resize operation is being requested; else (0b) ignore CQ_Resize. If a resize operation is performed, the cq_resize_count CQ context variable (known as hw_cq_select in the description of CQ resizing) is incremented by the 10 GbE controller. See the description of CQ resizing in Section 38.36.2.1.3 for more information on the CQ resizing operation.

Avoid_Memory_Conflicts (1)

If set (1b), the size of each CQE is padded with 0's to a total of 64 bytes and the size of the CQ doubles; else (0b) each CQE is 32 bytes in size.

TPH_en (1)

If set (1b), TPH is enabled for this resource. If clear (0b), TPH is not used for this resource.

TPH_value (8)

If TPH_en is set (1b), TPH STag is initialized with TPH_value. If TPH_en is clear (0b), this field is ignored.

Leaf_PBL_Size (2)

See Section , "Address Translation and Protection Overview" for more details on usage of two level PBLs.

0 = Reserved

1 = Variable (one level)

2 = 256 bytes (two level)

3 = 4 KB (two level)

Check_Overflow (1)

If set (1b), the 10 GbE controller checks for CQ overflow conditions; else it is host software's responsibility to ensure that a CQ cannot overflow.

Virtually mapped (1)

If clear (0b), Physical_Buffer_Address is the physical address of the physically contiguous CQ ring buffer; else the CQ is virtually mapped and first_pm_pbl_index is the index of the PBLE HMC object to use for mapping the virtually contiguous CQ. The CQ buffer must be aligned to a multiple of 4 KB in host memory when virtually mapped is set.

first_pm_pbl_index (28)

Points to the first HMC FPM space PBLE index for the page table for the CQ. Host software is responsible for populating the PBLEs with the page list associated with the CQ before issuing the create CQ operation and the page table is not allowed to be changed.

Physical_Buffer_Address (56)

This field specifies the physical buffer address associated with the queue in host memory if virtually mapped is set to zero. If virtually mapped is set to one, this field is ignored by the 10 GbE controller. This address must be aligned to a 256-byte boundary.

cq_size (20)

The number of CQEs allowed on the CQ. Note that the actual number of concurrent CQEs that might be allocated to the CQ is cq_size -1. 0 and 1 are invalid sizes for a CQ.

cq_shadow_read_threshold (18)

Controls when the CQ shadow area is read to in order to update hardware's copy of CQE_Index. When the CQ drops below the number for CQEs specified by cq_shadow_read_threshold, the CQ shadow area is read to find out which CQEs have been processed by software. The value of 0 indicates that the CQ shadow area is only read when the CQ has no more available entries to use for a new CQE that needs to be generated.

CQ_ID (17)

Identifies the CQ index associated with the current CQP operation.

CQ_Context_Value (63)

This value is returned in the CEQE when an event is generated for the CQ specified by CQ_ID. This value is intended to be used by host software to quickly locate host software's CQ context (virtual pointer to CQ context).

Doorbell shadow address (58)

A physically mapped pointer in host memory that hardware reads to determine the CQEs that are owned by hardware. The format of the doorbell shadow area is listed in Table 38-465. This buffer must be aligned to a multiple of cache-line size bytes or 64 bytes, whichever is greater.

Table 38-465.CQ Doorbell Shadow Area

Byte Offset	[Bit Range]Field Name			
0	[63:20]	RSVD	[19:0]	CQE_Index
8	[63:0]	RSVD		
16	[63:0]	RSVD		
24	[63:0]	RSVD		
32	[63:18] [17:16] [15]	RSVD arm_seq_num arm_next_se	[14] [13:0]	arm_next sw_cq_select
40	[63:0]	RSVD		
48	[63:0]	RSVD		
56	[63:0]	RSVD		

CQE_Index (20)

Indicates the next CQE that software are polling through verbs. This field is used to update the hardware's version of context to detect CQ overrun conditions.

arm_next (1)

Indicates that software wants a CEQE generated for the next CQ entry posted to the CQ.

arm_next_se (1)

Indicates that software wants a CEQE generated for the next CQ entry posted to the CQ that is associated with a solicited event operation.

arm_seq_num (2)

Arm Sequence Number (arm_seq_num) is a valued incremented by software each time an arm request is issued. It is expected that no more than two arm requests are issued per CE received. If more than 3 arm requests are issued without waiting for a CE, arm request might be lost.

sw_cq_select (14)

This field is used for CQ resize operations. See Section 38.36.2.1.3 for more details on the usage of this field.

38.36.3.3.4 Allocate/Register/Register Shared/De-Allocate STag Descriptor Format

This WQE format is used by host software to manage memory regions and windows. See **Address Translation and Protection Overview** for more details on memory registration concepts with the 10 GbE controller.

Table 38-466.CQP Allocate/Register/Register Shared/De-Allocate STag WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0] Buffer Offset	Virtual Address or First		
8	[63] [62:48]	RSVD PD_ID	[47:46] [45:00]	RSVD STag Length
16	[63:54] [53:32]	RSVD Parent_STag_Index	[31:08] [07:00]	Driver_Key_STag_Index Consumer_Key

Table 38-466.CQP Allocate/Register/Register Shared/De-Allocate STag WQE Format

Byte Offset	[Bit Range]Field Name			
24	[63] [62:61] [60] [59] RSVD [57:54] [53]	WQE_Valid RSVD use_hmc_fcn_index[1] VA_Based_TO RSVD RemoteAccessEnabled	[52:48] [47] [46] [45:44] [43] [42:38] [37:32] [31:0]	Access_Rights RSVD Host_Page_Size Leaf_PBL_Size Memory_Region RSVD OP RSVD
32	[63:0]	Physical_Buffer_Address		
40	[63:6]	RSVD	[5:0]	hmc_fcn_index
48	[63:28]	RSVD	[27:0]	first_pm_pbl_index
56	[63:0]	RSVD		

Notes:

1. This field is only valid for PF operation and will be treated as 0 for VFs.

OP (6) and WQE_Valid

See Section 38.36.3.3.1.

Virtual address or first buffer offset (64)

Indicates the base VA for this region/window for VA-based entries and indicates the first buffer offset for zero-based entries. For VA-based entries, the least significant bits (12 bits for entries based on 4 KB pages, 21 bits for entries based on 2 MB pages) of the VA indicate the offset in bytes from the beginning of the first page where the entry begins.

Physical_Buffer_Address (64)

This field specifies the physical buffer address associated with the virtual address if Leaf_PBL_Size is set to 0b. If Leaf_PBL_Size is not set to zero, this field is ignored by the 10 GbE controller. This address must be aligned to the page size specified in Host_Page_Size. Any offset into the page will be ignored by the 10 GbE controller.

PD_ID (15)

Indicates the protection domain ID associated with the memory region or window.

STag length (46)

Length of the memory region or memory windows specified by the STag index specified by Driver_Key_Stag_Index. A value of 0b indicates that the length is not checked. This is typically used for systems that want to address all of RAM with a single STag. The maximum size that can be registered with a single STag with 4 KB pages and 1 level PBLs is 1TB (2^{28} 4KB pages).

Parent_STag_Index (22)

Index of the STag associated with the memory window (Memory_Region = 0) or linked memory region (OP = Register Shared MR).

Driver_Key_STag_Index (24)

Index and *Driver Key* fields of the STag associated with the memory window (Memory_Region = 0b) or linked memory region (OP = register shared MR). The 10 GbE controller supports a variable size STag index. This means that the number of bits used for *Driver Key* and for STag_Index are dependent on the maximum number of STag supported for a given PCI function. For example, if a PCI function read the

FPMPEMRSZ field and found that the maximum number of MRTEs was 64 KB, the lower 16 bits of this field would be the STag *Index* and the upper 8 bits would be a driver key that the driver can randomize to make guessing the MRTE layout more difficult to guess. STag index 4 M-1 (4,194,303) is reserved for hardware use. The completion returns an error if this reserved STag index is specified.

Consumer key (8)

Consumer key is the least significant 8-bit portion of the STag. This field is supplied by the user application or the driver.

use_hmc_fcn_index (1)

When set to 0b, the HMC function index is determined by the PCI function associated with the PCI function that issues the CQP operation. When set to 1b, hmc_fcn_index can be set to a HMC function index that is a VF associated with the PF associated with the CQP instance used to issue the command. Only valid when issued to a PF CQP instance. Ignored (treated as if set to 0) for VF CQP instances.

hmc_fcn_index (6)

Only valid when issued to a PF CQP instance and use_hmc_fcn_index is set to 1b. Ignored otherwise.

VA_Based_TO (1)

VA_Based_TO specifies if the STag is zero-based or Virtual Address (VA) based. Zero-based STags carry only the first buffer offset in the VA or *First Buffer Offset* field. VA-based STags carry the full base VA including first buffer offset in the VA or *First Buffer Offset* field. The STag is VA based if VA_Based_TO is set (1b); else the STag is zero based.

RemoteAccessEnabled (1)

If set, the entry is enabled for remote access. See the remote access flag in verbs for more details.

Access_Rights (5)

Indicates the rights assigned to this STag. The values for this field are 1 (enable local read), 2 (enable local write), 4 (enable remote read), 8 (enable remote write), and 16 (enable window bind).

Host_Page_Size (1)

Host_Page_Size specifies the page size of the backing pages for the STag. The values for this field are:

0b = 4 KB pages

1b = 2 MB pages

Leaf_PBL_Size (2)

The 10 GbE controller support physically contiguous STags and two forms of virtually contiguous STags. Physically contiguous STag do not require any PBLs and store physical address of the first page of the STag directly with the STag (No leaf PBL). Virtually contiguous STags that can be represented with a single HMC virtually contiguous address range require a single level PBL of Variable size. Virtually contiguous STags that are large (or in cases where the HMC address space for PBLs becomes fragmented) might require two level PBLs. In this case, the 10 GbE controller needs to know the length of the leaf PBLs in order to properly manage access to the PBLs. The valid settings for Leaf_PBL_Size are the following:

0 = No leaf PBL

1 = Variable (one level)

2 = 256 bytes (two level)

3 = 4 KB (two level)

Memory_Region (1)

If Memory_Region is set (1b) then the entry describes a memory region; If cleared (0b) the entry describes a memory window.

first_pm_pbl_index (64)

first_pm_pbl_index designates the HMC base address for the PBLs for this STag.

Table 38-467.WQE Fields Valid for Allocate/Register/Register Shared/De-Allocate STAG

WQE Field	Allocate	Register	Register Shared	Deallocate	Comments
Memory_Region	Valid	Valid	Not Valid	Valid	
VA_based_TO	Not Valid	Valid	Valid	Not Valid	
Leaf_PBL_Size	Valid if MR=1	Valid	Not Valid	Not Valid	
Access_Rights	Not Valid	Valid	Valid	Not Valid	
RemoteAccessEnabled	Valid	Valid	Valid	Not Valid	
PD_ID	Valid	Valid	Valid	Valid	
Driver Key STAG Index	Valid	Valid	Valid	Valid	
Parent STAG Index	Not Valid	Not Valid	Valid	Not Valid	
Consumer_Key	Not Valid	Valid	Valid	Not Valid	Host software is expected to validate the consumer key on a de-allocate and on register shared.
Base_VA or First_Buffer_Offset	Not Valid	Valid	Valid	Not Valid	
Physical_Buffer_Address	Not Valid	Valid	Not Valid	Not Valid	
first_pm_pbl_index	Valid	Valid	Not Valid	Not Valid	
STag Length	Valid if MR=1	Valid	Not Valid	Not Valid	
Host_Page_Size	Valid	Valid	Not Valid	Not Valid	
use_hmc_fcn_index	Valid	Valid	Valid	Valid	This field is ignored (assumed to be 0b) for VFs.
hmc_fcn_index	Valid	Valid	Valid	Valid	Only used when use_hmc_fcn_index is set to 1b.

Table 38-467 lists the WQE fields that are valid for the STag management WQEs. Fields that are marked as not valid are ignored by the 10 GbE controller.

38.36.3.3.5 Query STag Descriptor Format

WQE returns the STag value and current state of an STag (valid or invalid) and the consumer key in the completion for the WQE. See Table 38-457 for information on CQP completion codes that indicate the STag state. The full STag value is returned in the Operation_Return_Value. Note that the error bit in the CQE should be ignored for query STag operations.

Table 38-468.CQP Query STag WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	RSVD		
8	[63:0]	RSVD		
16	[63:30] [29:08]	RSVD STag_Index	[7:0] RSVD	
24	[63] [62:61] [60] [59:41]	WQE_Valid RSVD use_hmc_fcn_index[1] RSVD	[40:38] RSVD (AdditionalFragmentCount) [37:32] [31:0]	 OP RSVD
32	[63:0]	RSVD		
40	[63:6]	RSVD	[5:0]	hmc_fcn_index[a]
48	[63:0]	RSVD		
56	[63:0]	RSVD		

Notes:

1. This field is only valid for PF operations and is treated as 0b for VFs.

OP (6) and WQE_Valid

See Section 38.36.3.3.1.

STag_Index (22)

STag_Index specifies the index of the STag to be queried.

use_hmc_fcn_index (1)

When set to 0b, the HMC function index is determined by the PCI function associated with the PCI function that issues the CQP operation. When set to 1b, hmc_fcn_index can be set to a HMC function index that is a VF associated with the PF associated with the CQP instance used to issue the command. Only valid when issued to a PF CQP instance. Ignored (treated as if set to 0b) for VF CQP instances.

hmc_fcn_index (6)

Only valid when issued to a PF CQP instance and use_hmc_fcn_index is set to 1b. Ignored otherwise.

38.36.3.3.6 Allocate Local MAC Entry

This WQE format is used to request an entry from the 10 GbE controller's MAC address table. The MAC table entry index is reported in the Operation_Return_Value field of the CQE associated with this WQE if the operation is successful.

Table 38-469.Allocate Local MAC and IP Address Entry WQE Format

Byte Offset	[Bit Range]Field Name		
0	[63:0]	RSVD	
8	[63:0]	RSVD	
16	[63:0]	RSVD	
24	[63] [62:41] [40:38]	WQE_Valid RSVD RSVD (AdditionalFragmentCount)	[37:32] OP [31:0] RSVD
32	[63:0]	RSVD	

Table 38-469.Allocate Local MAC and IP Address Entry WQE Format

Byte Offset	[Bit Range]Field Name	
40	[63:0]	RSVD
48	[63:0]	RSVD
56	[63:0]	RSVD

OP (6) and WQE_Valid

See Section 38.36.3.3.1.

38.36.3.3.7 Manage Local MAC Descriptor Format

This WQE format is used to manage the 10 GbE controller's local MAC table. A MAC table entry must be created before TCP/IP connection setup can be started and before a QP can be transitioned to RTS since the QP context references an local MAC entry.

Note that the PE has a separate MAC address table from the L2 filters. The reason for this is that the MAC address table is somewhat fluid for L2 but fixed for PE. Note that it is expected that CQP firmware will program both the PE MAC table and the L2 MAC IP table filter to either match indexes or will keep a mapping between L2 filters and PE MAC address. Host software is responsible for keeping the MAC addresses in sync between L2 and the PE table but there is no relationship required between the MAC tables required other than both MAC address filters contain the same values. If there is a change in the local MAC address, it is acceptable for a limited amount of time to go by when L2 and PE are using different local addresses since the network has been disrupted during the transition time anyway.

Table 38-470.Manage Local MAC WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	RSVD		
8	[63:0]	RSVD		
16	[63:32]	RSVD	[31:0] RSVD	
24	[63] [62] [61:41]	WQE_Valid Free_Table_Entry RSVD	[40:38] (AdditionalFragmentCount) [37:32] [31:6] [5:0]	RSVD OP RSVD MAC_Table_Index
32	[63:48] [47:40] [39:32] [31:24]	RSVD Local_MAC_Address[5] Local_MAC_Address[4] Local_MAC_Address[3]	[23:16] [15:8] [7:0]	Local_MAC_Address[2] Local_MAC_Address[1] Local_MAC_Address[0]
40	[63:0]	RSVD		
48	[63:0]	RSVD		
56	[63:0]	RSVD		

OP (6) and WQE_Valid

See Section 38.36.3.3.1.

MAC_Table_Index (6)

MAC_IP_Table_Index specifies the index of the local IP table to be manipulated. IP table entries might be shared between members of a PE team.

Free_Table_Entry (1)

When this bit is set (1b), the table entry specified by MAC_IP_Table_Index is freed and available for allocation through the allocate local MAC and IP address table entry CQP operation. When this bit is set, the values for MAC and IP address are ignored. When this bit is clear (0b), this operation configures the MAC and IP addresses for the table entry specified by MAC_IP_Table_Index.

Local_MAC_Address[5-0] (8 bits each)

Specifies the MAC address associated with the local IP address(es) specified in Local_IPv6_IP_Address and Local_IPv4_Address. Index 0 is the LSB and byte 5 is the MSB.

38.36.3.3.8 Manage ARP Table Descriptor Format

This WQE format is used to manage the 10 GbE controller's ARP table. An ARP table entry must be created before a QP can be transitioned to RTS or established since the QP context references an ARP table entry.

Table 38-471.Manage ARP Table WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	RSVD		
8	[63:32]	RSVD	[31:0]	Reachability_Max
16	[63:48]	RSVD	[47:0]	MAC_Address
24	[63] [62:45] [44] [43] [42]	WQE_Valid RSVD Query Permanent Entry_Valid	[41] [40:38] [37:32] [31:16] [15:0]	RSVD RSVD (AdditionalFragmentCount) OP RSVD ARP_Entry_Index
32	[63:0]	RSVD		
40	[63:0]	RSVD		
48	[63:0]	RSVD		
56	[63:0]	RSVD		

OP (6) and WQE_Valid

See Section 38.36.3.3.1.

ARP_Entry_Index (16)

ARP_Entry_Index specifies the index of the ARP table entry to be manipulated.

Permanent (1)

When this bit is set (1b), the ARP entry is marked as permanent and is not aged; when this bit is clear (0b), the entry is created in the reachable state and is aged.

Query (1)

When this bit set (1b), the ARP entry is not written instead the entry is read and the reachability timestamp from the ARP entry is returned in the Operation_Return_Value field of the CQE. When this bit is clear (0b), the ARP entry is written according to the specified parameters.

Entry valid (1)

When this bit is set (1b), the ARP entry is set to the valid state; else the ARP entry is set to the invalid state.

Note that the lower 32 bits of the AE generated when an ARP entry is stale contain the timestamp value from the ARP table entry.

Reachability_Max (32 bit):

The maximum number of microseconds that should be allowed to expire before generating a doubt neighbor reachability AEQE. Doubt neighbor reachability AEs indicate that there has been TCP/IP transmits outstanding for Reachability_Max without receiving any inbound traffic from the associated neighbor. In order to receive subsequent doubt neighbor reachability AEs, a manage ARP table WQE must be re-submitted to CQP with query clear (0b) and all original information supplied when the ARP entry was last updated.

MAC address (48 bits):

MAC address to be placed in the ARP cache table.

38.36.3.3.9 Manage VF PBLE Backing Pages Descriptor Format

The manage VF PBLE backing pages descriptor is used to populate and depopulate VF related HMC page descriptor content. A VF that allocates the PBLE HMC object backing pages can request CQP to copy the page list associated with the backing pages to the PF driver allocated page descriptor pages. A VF might also request CQP to invalidate VF related page descriptor entries. This operation reduces the number of VF to PF messages required to populate the VF related HMC PBLE object backing pages.

Table 38-472.CQP Manage VF PBLE Backing Pages WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	RSVD		
8	[63:0]	RSVD		
16	[63:44] [43:32] [31:25]	RSVD SD_Index RSVD	[24:16] [15:10] [9:0]	First_PD_Index RSVD PD_Entry_Count
24	[63] [62] [61:41] [40:38]	WQE_Valid Invalidate_PD_Entries RSVD RSVD (AdditionalFragmentCount)	[37:32] [31:0]	OP RSVD
32	[63:3]	PD_Pagelist_Physical_Buffer_Address	[2:0]	RSVD
40	[63:0]	RSVD		
48	[63:0]	RSVD		
56	[63:0]	RSVD		

OP (6) and WQE_Valid

See Section 38.36.3.3.1.

Invalidate_PD_Entries (1)

When clear (0b), VF PD entries is copied from VF address space to the PF allocated PDs. When set (1b), the PF allocated PDs are invalidated.

PD_Entry_Count (10)

This field indicates the number of PD entries to be populated or invalidated. Valid values for this field are 1 through 512. Other values generate completion errors for the CQP operation. PD_Entry_Count + First_PD_Index must not exceed 512 or a completion error is generated.

First_PD_Index (9)

This field indicates the starting index of the PD page. SD_Index identifies the PD page and First_PD_Index provides the offset into the PD page that is the destination of the CQP operation. If the PD entries associated with the SD_Index and First_PD_Index is not related to a PBLE object, an completion error is generated by CQP for this operation.

SD_Index (12)

This field indicates the SD index that identifies the PD page. The SD entries are in no way modified. If the SD is not related to a PBLE object, then CQP returns a completion error.

PD_Pagelist_Physical_Buffer_Address (61)

This field indicates the 8-byte aligned physical address of the packing page list to be copied to the PD page if Invalidate_PD_Entries is clear. This address is a guest physical address. Host software is responsible to properly set the fields of the PD page list entries previous to issuing the request to CQP.

38.36.3.3.10 Manage Push Page Descriptor Format

The manage push page descriptor is used to assign a push page from BAR0 to a VSI and TC. For an allocate operation, the push page index assigned by CQP is returned in the Operation_Return_Value of the CQE associated with the WQE. For VFs, the push page index is function relative. When GLPE_PSHCFG.PSHCFG_DB_SPLIT is set, two pages (one for doorbells and one for push WQEs) are represented by push page index instead of one.

Table 38-473.CQP Manage Push Page WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	RSVD		
8	[63:0]	RSVD		
16	[63:10]	RSVD	[09:0]	QS_Handle
24	[63] [62] [61:41] [40:38]	WQE_Valid Free_Page RSVD RSVD (AdditionalFragmentCount)	[37:32] [31:10] [9:0]	OP RSVD Push_Page_Index
32	[63:0]	RSVD		
40	[63:0]	RSVD		
48	[63:0]	RSVD		
56	[63:0]	RSVD		

OP (6) and WQE_Valid

See Section 38.36.3.3.1.

Free_Page (1)

When clear (0b) a push page is allocated for the VSI and TC specified by QS_Handle. When set (1b), the push page specified by Push_Page_Index is freed.

Push_Page_Index (10)

Only valid when free page is set (1b). Identifies the push page to be freed. For VFs, the push page index is function relative.

QS_Handle (10)

Identifies the QS handle associated with the push page.

38.36.3.4 Upload Context Descriptor Format

Upload context is used by host software to freeze a QPs state and upload the context to the host.

Table 38-474.CQP Upload Context WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	RSVD		
8	[63:0]	RSVD		
16	[63:0]	QP_Context_Address		
24	[63] [62] [61] [60:52] [51:48]	WQE_Valid Freeze_QP RSVD RSVD QP_Type	[47:41] [40:38] [37:32] [31:18] [17:0]	RSVD RSVD (AdditionalFragmentCount) OP RSVD QP_ID
32	[63:0]	RSVD		
40	[63:0]	RSVD		
48	[63:0]	RSVD		
56	[63:0]	RSVD		

OP (6) and WQE_Valid

See Section 38.36.3.3.1.

QP_ID (18)

Identifies the QP number be uploaded to host memory.

QP_Type (3)

iWarp=1, UDA=2,all other values are reserved.

Freeze_QP (1)

When set, the QP is frozen before the context is retrieved, else the QP continues processing normally after the snapshot of the context has been returned.

Raw_Format

When set, the QP context is reported in its raw form. This bit should only be set for debug context operations.

QP Context Address

A physically mapped pointer in host memory that contains uploaded context structure format listed in the following table. This buffer must be aligned to a multiple of 4 bytes.

Table 38-475.Uploaded Context Structure Format

Byte Offset	[Bit Range]Field Name			
0	[63:56] [55:48]	RSVD kalive_timer_probes	[47:8] [7:4] [3:0]	RSVD iwarp_state TCP_state
8	[63:32]	timestamp_age	[31:0]	timestamp_recent
16	[63:32]	snd_wnd	[31:0]	snd_nxt
24	[63:32]	rcv_wnd	[31:0]	rcv_nxt
32	[63:32]	snd_una	[31:0]	snd_max
40	[63:32]	rtt_var	[31:0]	srtt
48	[63:32]	cwnd	[31:0]	ss_thresh
56	[63:32]	snd_wl2	[31:0]	snd_wl1
64	[63:54] [53:48] [47:41]	RSVD rexmit_count RSVD	[40:32] [31:0]	RSVD max_snd_window
72	[63:46] [45:32] [31]	RSVD hw_sq_tail q1_wa	[30:27] [26:24] [23:16] [15:0]	RSVD dupacks probe_cnt RSVD
80	[63:14]	RSVD	[13:0]	hw_rq_tail
88	[63:0]	RSVD		
96	[63:0]	RSVD		
104	[63:0]	RSVD		
112	[63:0]	RSVD		
120	[63:0]	RSVD		

hw_sq_tail

This is the 32-byte WQE quanta index of the first unacknowledged WQE for the SQ. There is not a CQE generated for this WQE.

hw_rq_tail

This is the 32-byte WQE quanta index of the first incomplete WQE for the RQ. There is not a CQE generated for this WQE.

q1_wa

This value is set (1b) if Q1 (or inbound RDMA read queue) work is pending when the context was uploaded. The value is clear (0b) if no Q1 work was pending when the context was uploaded.

kalive_timer_probes

The number of retransmits that have been sent (RFC 1122).

Note: See Section 38.36.4.2 for more on context variable definitions.

38.36.3.4.1 Manage HMC PM Function Table

The manage HMC PM function table is used to allocate and free HMC private memory functions for VFs associated with a PF. This WQE is only supported for PF CQPs. For allocate operations (Free_PM_FCN=0b), the HMC function index is reported in the Operation_Return_Value field of the CQE associated with this WQE if the operation is successful.

Table 38-476.CQP Manage HMC PM Function Table WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	RSVD		
8	[63:0]	RSVD		
16	[63:0]	RSVD		
24	[63] [62] [61:38]	WQE_Valid Free_PM_FCN RSVD	[37:32] [31:7] [6:0]	OP RSVD VF_Index
32	[63:0]	RSVD		
40	[63:0]	RSVD		
48	[63:0]	RSVD		
56	[63:0]	RSVD		

OP (6) and WQE_Valid

See Section 38.36.3.3.1.

Free_PM_FCN (1)

If clear (0b), the 10 GbE controller attempts to allocate a PM function for the VF specified in VF_Index. If set (1b), then the 10 GbE controller frees the PM function specified in VF_index.

VF_Index (7)

VF_Index indicates the PCI VF index to be used for the allocation or free request.

38.36.3.4.2 Create/Discard CEQ Descriptor Format

Host software uses the following structure to manage CEQs through the control QP. CEQs are comprised of a packed array of CEQ entries (see Section 38.36.2.4) that are managed as a circular queue. They can be either virtually or physically contiguous memory buffers. When MSI-X is enabled, each CEQ might be assigned to independent MSI-X vectors to enable distribution of completion event processing across multiple CPUs. Application's process and MSI-X vector assignment capabilities and mechanisms are specific to each operating system.

Table 38-477.CQP Create/Discard CEQ WQE Format (Sheet 1 of 2)

Byte Offset	[Bit Range]Field Name			
0	[63:0]	RSVD		
8	[63:0]	RSVD		
16	[63:18]	RSVD	[17:0]	ceq_size

Table 38-477.CQP Create/Discard CEQ WQE Format (Sheet 2 of 2)

Byte Offset	[Bit Range]Field Name			
24	[63] [62:61] [60] [59:48] [47] [46] [45:44]	WQE_Valid RSVD TPH_en RSVD Virtually Mapped RSVD Leaf_PBL_Size	[43:41] [40:38] (AdditionalFragmentCount) [37:32] [31:16] [15:7] [6:0]	RSVD RSVD OP RSVD RSVD CEQ_ID
32	[63:8]	Physical_Buffer_Address	[7:0]	RSVD
40	[63:0]	RSVD		
48	[63:28]	RSVD	[27:0]	first_pm_pbl_index
56	[63:8]	RSVD	[7:0]	TPH_Value

OP (6) and WQE_Valid

See Section 38.36.3.3.1.

CEQ_ID (7)

CEQ_ID specifies the CEQ to manipulated.

TPH_en (1)

If set (1b), TPH is enabled for this resource. If clear (0b), THP is not used for this resource.

TPH_value (8)

If TPH_en is set (1b), TPH STag is initialized with TPH_value. If TPH_en is clear (0b), this field is ignored.

first_pm_pbl_index (28)

Points to the first HMC FPM space PBLE index for the page table. Host software is responsible for populating the PBLEs with the page list associated with the object before using the object.

Leaf_PBL_Size (2)

See Section , “Address Translation and Protection Overview” for more details on usage of two level PBLs.

0 = Reserved

1 = Variable (one level)

2 = 256 bytes (two level)

3 = 4 KB (two level)

Virtually mapped (1)

If clear (0b), Physical_Buffer_Address is the physical address of the physically contiguous CEQ ring buffer; else the CEQ is virtually mapped and the first_pm_pbl_index is the index of the PBLE HMC object of the PBL to use for mapping the virtually contiguous CEQ. The CEQ buffer must be aligned to a multiple of 4 KB in host memory when virtually mapped is set.

Physical_Buffer_Address (56)

This field specifies the physical buffer address associated with the queue in host memory if virtually mapped is set to zero. If virtually mapped is set to one, this field is ignored by the 10 GbE controller. This address must be aligned to a 256-byte boundary.

ceq_size (18)

The number of CEQEs allowed on the CEQ. Note that the actual number of concurrent CEQEs that can be allocated to the CEQ is ceq_size -1. 0 and 1 are invalid sizes for a CEQ.

38.36.3.5 Create/Discard AEQ Descriptor Format

Host software uses the following structure to manage AEQs through the control QP. AEQs are comprised of a packed array of AEQ entries (see Section 38.36.2.5) that are managed as a circular queue. They can be either virtually or physically contiguous memory buffers.

Table 38-478.CQP Create/Discard AEQ WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	RSVD		
8	[63:0]	RSVD		
16	[63:19]	RSVD	[18:0]	AEQE Count
24	[63] [62:48] [47] [46]	WQE_Valid RSVD Virtually Mapped RSVD	[45:44] [43:41] [40:38] (AdditionalFragmentCount) [37:32] [31:0]	Leaf_PBL_Size RSVD RSVD OP RSVD
32	[63:8]	Physical_Buffer_Address	[7:0]	RSVD
40	[63:0]	RSVD		
48	[63:28]	RSVD	[27:0]	first_pm_pbl_index
56	[63:0]	RSVD		

OP (6) and WQE_Valid

See Section 38.36.3.3.1.

first_pm_pbl_index (28)

Points to the first HMC FPM space PBLE index for the page table. Host software is responsible for populating the PBLEs with the page list associated with the object before using the object.

Leaf_PBL_Size (2)

See Section , “Address Translation and Protection Overview” for more details on usage of two level PBLs.

0 = Reserved

1 = Variable (one level)

2 = 256 bytes (two level)

3 = 4 KB (two level)

Virtually mapped (1)

If clear (0b), Physical_Buffer_Address is the physical address of the physically contiguous AEQ ring buffer; else the AEQ is virtually mapped and first_pm_pbl_index is the index of the PBLE HMC object of the PBL to use for mapping the virtually contiguous AEQ. The AEQ buffer must be aligned to a multiple of 4 KB in host memory when virtually mapped is set.

Physical_Buffer_Address (56)

This field specifies the physical buffer address associated with the queue in host memory if virtually mapped is set to zero. If virtually mapped is set to one, this field is ignored by the 10 GbE controller. This address must be aligned to a 256-byte boundary.

AEQE Count (19)

The number of AEQEs allowed on the AEQ. Note that the actual number of concurrent AEQEs that can be allocated to the AEQ is AEQE Count -1. 0 and 1 are invalid sizes for an AEQ.

38.36.3.5.1 Update PE SDs Descriptor Format

This WQE is used to program HMC segment descriptors (SDs) associated with PE enabled PCI functions. CQP ensures that PFs only access HMC functions that belong to the PF at the time of issues the CQP request. PFs can access its own SDs or SDs of protocol enabled VFs that belong to the PF. Accesses to other HMC functions are denied and generate a completion error. This operation is restricted to PF CQP operations.

Table 38-479.CQP Update PE SDs WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:32]	RSVD	[31:0]	SDCMD0
8	[63:32]	SDDATAHIGH0	[31:0]	SDDATALOW0
16	[63:7]	SD_Buffer_Address	[6] [5:0]	RSVD HMC_FCN_ID (PF only)
24	[63] [62:41] [40:38] (AdditionalFragmentCount)	WQE_Valid RSVD RSVD	[37:32] [31:8] [7] [6:4] [3:0]	OP Reserved SKIP_SD_ENTRY_0 Reserved SD_ENTRY_COUNT
32	[63] [62:0]	SD_ENTRY_VALID1 RSVD	[31:0]	SDCMD1
40	[63:32]	SDDATAHIGH1	[31:0]	SDDATALOW1
48	[63] [62:0]	SD_ENTRY_VALID2 RSVD	[31:0]	SDCMD2
56	[63:32]	SDDATAHIGH2	[31:0]	SDDATALOW2

OP (6) and WQE_Valid

See Section 38.36.3.3.1.

HMC_FCN_ID (6)

This field specifies the HMC function ID for the FPM settings to be queried. For PFs, the HMC_FCN_ID is strictly the PF index. For VFs, the HMC function ID was returned from the manage HMC PM function CQP operation.

SD_CMDn (32)

The SD_CMD field definition is the same as the PFHMC_SDCMD register.

SD_DATA_LOWn (32)

The SD_DATA_LOW field definition is the same as the PFHMC_SDDATALOW register.

SD_DATA_HIGHn (32)

The SD_DATA_HIGH field definition is the same as the PFHMC_SDDATAHIGH register.

SD_ENTRY_VALIDn (1)

Indicates of the SDCMD and DATA information is used to update additional SD entries.

SKIP_SD_ENTRY_0 (1)

Indicates SD_CMD0, SD_DATA_LOW/HIGH0 are to be ignored. This is typically useful only if all the SD updates are specified in the update SD host memory structure instead of this WQE.

SD_Buffer_Address (58)

Most significant address bits of the physical address of the update SD host memory structure (listed in Table 38-480). This structure is used to provide additional SD updates. This host memory is only accessed if SD_ENTRY_COUNT is not equal to 0b. The number of entries used from the update SD host memory structure is indicated by SD_ENTRY_COUNT.

SD_ENTRY_COUNT (4)

Indicates the number of contiguous (starting with SDCMD3) SD updates found in the update SD host memory structure.

Table 38-480.Update SD Host Memory Structure

Byte Offset	[Bit Range]Field Name			
0	[63:32]	RSVD	[31:0]	SDCMD3
8	[63:32]	SDDATAHIGH3	[31:0]	SDDATALOW3
16	[63:32]	RSVD	[31:0]	SDCMD4
24	[63:32]	SDDATAHIGH4	[31:0]	SDDATALOW4
32	[63:32]	RSVD	[31:0]	SDCMD5
40	[63:32]	SDDATAHIGH5	[31:0]	SDDATALOW5
48	[63:32]	RSVD	[31:0]	SDCMD6
56	[63:32]	SDDATAHIGH6	[31:0]	SDDATALOW6
64	[63:32]	RSVD	[31:0]	SDCMD7
72	[63:32]	SDDATAHIGH7	[31:0]	SDDATALOW7
80	[63:32]	RSVD	[31:0]	SDCMD8
88	[63:32]	SDDATAHIGH8	[31:0]	SDDATALOW8
96	[63:32]	RSVD	[31:0]	SDCMD9
104	[63:32]	SDDATAHIGH9	[31:0]	SDDATALOW9
112	[63:32]	RSVD	[31:0]	SDCMD10
120	[63:32]	SDDATAHIGH10	[31:0]	SDDATALOW10

38.36.3.5.2 Query FPM Values Descriptor Format

This WQE is used to query the FPM configuration for the PE portion of the HMC objects. The query FPM values command is used to trigger firmware to return the FPM base registers based on the previous software settings or default values.

Table 38-481.CQP Query FPM Values WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	RSVD		
8	[63:0]	RSVD		
16	[63:6]	RSVD	[5:0]	HMC_FCN_ID (PF only)
24	[63] [62:41] [40:38]	WQE_Valid RSVD RSVD (AdditionalFragmentCount)	[37:32] [31:0]	OP Reserved
32	[63:2]	Physical_Buffer_Address	[1:0]	RSVD
40	[63:0]	RSVD		
48	[63:0]	RSVD		
56	[63:0]	RSVD		

OP (6) and WQE_Valid

See Section 38.36.3.3.1.

HMC_FCN_ID (6)

This field specifies the HMC function ID for the FPM settings to be queried. For PFs, the HMC_FCN_ID is strictly the PF index. For VFs, the HMC function ID was returned from the manage HMC PM function CQP operation. If a CQP instance associated with a VF submits this operation, this field is ignored and the HMC_FCN_ID is determined by firmware.

Physical_Buffer_Address (62)

This field specifies the physical address of the buffer that firmware fills with the object maximum counts and the object size values.

Table 38-482.FPM Query Configuration Structure Format (Sheet 1 of 2)

Byte Offset	[Bit Range]Field Name			
0	[63:45] [44:32]	RSVD max_pe_sds	[31:14] [13:0]	RSVD first_pe_sd_index
8	[63:32]	GLHMC_PEQPOBJSZ	[31:19] [18:0]	RSVD max_qps
16	[63:32]	GLHMC_PECQOBJSZ	[31:18] [17:0]	RSVD max_cqs
24	[63:32]		[31:0]	
32	[63:32]	GLHMC_PEHTEOBJSZ	[31:0]	GLHMC_PEHTMAX
40	[63:32]	GLHMC_PEARPOBJSZ	[31:0]	GLHMC_PEARPMAX
48	[63:32]	GLHMC_PEMROBJSZ	[31:0]	GLHMC_PEMRMAX
56	[63:32]	GLHMC_PEXFOBJSZ	[31:0]	GLHMC_PEXFMAX
64	[63:32]	XFBLOCKSIZE	[31:0]	GLHMC_PEXFFLMAX
72	[63:32]	GLHMC_PEQ1OBJSZ	[31:0]	GLHMC_PEQ1MAX
80	[63:32]	Q1BLOCKSIZE	[31:0]	GLHMC_PEQ1FLMAX
88	[63:32]	GLHMC_PETIMEROBJSZ	[31:0]	GLHMC_PETIMERMAX
96	[63:32]		[31:0]	

Table 38-482.FPM Query Configuration Structure Format (Sheet 2 of 2)

Byte Offset	[Bit Range]Field Name			
104	[63:32]		[31:0]	
112	[63:32]	RSVD	[31:0]	GLHMC_PEPBLMAX
120	[63:48] [47:32] [31:20]	RSVD TIMERBUCKETCNT RSVD	[19:16] [15:8] [7:0]	HTMULTIPLIER RSVD max_ceqs

first_pe_sd_index (14)

This field specifies the function relative starting HMC segment descriptor index for PE resources.

max_pe_sds (13)

This field specifies the maximum number of PE SDs allowed for the HMC function.

max_qps (19)

This field specifies the maximum number of PE QPs allowed for the HMC function.

max_cqs (18)

This field specifies the maximum number of PE CQs allowed for the HMC function.

max_ceqs (8)

This field specifies the maximum number of PE CEQs allowed for the HMC function.

XFBLOCKSIZE (32)

This field specifies the number of transmit FIFO entries per transmit FIFO free list entry. This field is used to determine how much space is consumed for transmit FIFO free list entries. The number of transmit FIFO free list entries is calculated with the equation number of transmit FIFO entries / XFBLOCKSIZE.

Q1BLOCKSIZE (32)

This field specifies the number of Q1 FIFO entries per Q1 FIFO free list entry. This field is used to determine how much space is consumed for Q1 free list entries. The number of Q1 free list entries is calculated with the equation number of Q1 entries / Q1BLOCKSIZE.

HTMULTIPLIER (4)

This field specifies the number of hash filter bucket entries per QP. This field is used to determine how much space is consumed for hash filter entries. The number of hash entries is calculated with the equation (round_up_512(number of QPs) rounded up to the next power of two) * HTMULTIPLIER.

TIMERBUCKETCNT (16)

This field specifies the number of timer buckets. This field is used to determine how much space is consumed for timers. The number of timer entries is calculated with the equation (round_up_512(number of QPs)/512 + 1) * TIMERBUCKETCNT.

The remaining fields are register values from the HMC function specified by HMC_FCN_ID or the VF index.

38.36.3.5.3 Commit FPM Values Descriptor Format

This WQE is used to commit the FPM configuration for the PE portion of the HMC objects. The commit FPM values command is used to trigger firmware to calculate the FPM base registers based on the software settings.

Table 38-483.CQP commit FPM Values WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	RSVD		
8	[63:0]	RSVD		
16	[63:6]	RSVD	[5:0]	HMC_FCN_ID (PF only)
24	[63] [62:41] [40:38]	WQE_Valid RSVD RSVD (AdditionalFragmentCount)	[37:32] [31:0]	OP Reserved
32	[63:2]	Physical_Buffer_Address	[1:0]	RSVD
40	[63:0]	RSVD		
48	[63:0]	RSVD		
56	[63:0]	RSVD		

OP (6) and WQE_Valid

See Section 38.36.3.3.1.

HMC_FCN_ID (6)

This field specifies the HMC function ID for the FPM settings to be programmed. For PFs, the HMC_FCN_ID is strictly the PF index. For VFs, the HMC function ID was returned from the manage HMC PM function CQP operation. If a CQP instance associated with a VF submits this operation, this field is ignored and the HMC_FCN_ID is determined by firmware.

Physical_Buffer_Address (62)

This field specifies the most significant bits of the physical address of the buffer that firmware uses to configure the object maximum counts and the object size values. The buffer described must be aligned to a four byte aligned boundary.

Table 38-484.FPM Commit Configuration Structure Format (Sheet 1 of 2)

Byte Offset	[Bit Range]Field Name			
0	[63:32]	GLHMC_PEQPBASE	[31:19] [18:0]	RSVD GLHMC_PEQPCNT
8	[63:32]	GLHMC_PECQBASE	[31:18] [17:0]	RSVD GLHMC_PECQCNT
16	[63:32]		[31:0]	
24	[63:32]	GLHMC_PEHTEBASE	[31:0]	GLHMC_PEHTCNT[1]
32	[63:32]	GLHMC_PEARPBASE	[31:0]	GLHMC_PEARPCNT
40	[63:32]	GLHMC_APBVTINUSEBASE	[31:0]	RSVD
48	[63:32]	GLHMC_PEMRBASE	[31:0]	GLHMC_PEMRCNT
56	[63:32]	GLHMC_PEXFBASE	[31:0]	GLHMC_PEXFCNT
64	[63:32]	GLHMC_PEXFFLBASE	[31:0]	GLHMC_PEXFFLCNT[1]

Table 38-484.FPM Commit Configuration Structure Format (Sheet 2 of 2)

Byte Offset	[Bit Range]Field Name			
72	[63:32]	GLHMC_PEQ1BASE	[31:0]	GLHMC_PEQ1CNT
80	[63:32]	GLHMC_PEQ1FLBASE	[31:0]	GLHMC_PEQ1FLCNT[1]
88	[63:32]	GLHMC_PETIMERBASE	[31:0]	GLHMC_PETIMERCNT[1]
96	[63:32]		[31:0]	
104	[63:32]		[31:0]	
112	[63:32]	GLHMC_PEPBLBASE	[31:0]	GLHMC_PEPBLCNT
120	[63:0]	RSVD		

Notes:

1. These fields are calculated by CQP and returned on completion. On submission, these fields are ignored.

These fields are register values from the HMC function specified by HMC_FCN_ID or the VF index. On request submission, the GLHMC_{obj}CNT fields must be filled in the software. Upon completion of the request, all fields are filled in with the updated values that were actually committed.

38.36.3.5.4 Flush WQEs Descriptor Format

This WQE is used to flush pending WQEs to a CQ. The initial usage for the WQE is to support Winsock Direct socket handoff from one process to another. This WQE is now also used for all WQE flushing. Host software uses this after the quad hash has been deleted for a QP in order to comply with verbs requirements. Note that only a single unprocessed WQE is flushed from the WQ(s) specified by the *FlushRQ*/*FlushSQ* bits. Host software must generate additional completions if more than one outstanding WQE is pending on a WQ that has been flushed.

This WQE can be used to flush pending transmit and receive WQEs for UDA and iWARP QPs. For UDA all filters forwarding to the specified UDA QP must be disabled, prior to flushing WQEs.

Table 38-485.CQP Flush WQEs WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	RSVD		
8	[63:20] [19:16]	RSVD AE_Source	[15:0]	AE_Code
16	[63:48] [47:32]	SQ_Major_Code SQ_Minor_Code	[31:16] [15:0]	RQ_Major_Code RQ_Minor_Code
24	[63] [62] [61] [60] [59]	WQE_Valid FlushRQ FlushSQ UserFlushCode GenerateAE	[58:41] [40:38] (AdditionalFragmentCount) [37:32] [31:18] [17:0]	RSVD RSVD OP Reserved QP_ID
32	[63:0]	RSVD		
40	[63:0]	RSVD		
48	[63:0]	RSVD		
56	[63:0]	RSVD		

OP (6) and WQE_Valid

See Section 38.36.3.3.1.

FlushRQ (1)

0b = Do not flush any pending WQEs from the RQ.

1b = Generate a completion for the first pending RQ with a flushed return code.

FlushSQ (1)

0b = Do not flush any pending WQEs from the SQ.

1b = Generate a completion for the first pending SQ with a flushed return code.

UserFlushCode (1)

UserFlushCode indicates that the WQEs flushed should carry the values specified by SQ/RQ major and minor codes.

0b = Any flushed WQEs carry the flushed completion major and minor codes

1b = Flushed WQEs for the SQ carry SQ_Major_Code and SQ_Minor_Codes, flushed WQEs on the RQ carry RQ_Major_Code and RQ_Minor_Code.

SQ_Major_Code (16)

Valid only when UserFlushCode=1b. SQ_Major_Code is reported in *Major Error Code* field of the CQE for the first flushed WQE for the SQ of the QP specified by QP_ID.

SQ_Minor_Code (16)

Valid only when UserFlushCode=1b. SQ_Minor_Code is reported in *Minor Error Code* field of the CQE for the first flushed WQE for the SQ of the QP specified by QP_ID.

RQ_Major_Code (16)

Valid only when UserFlushCode=1b. RQ_Major_Code is reported in *Major Error Code* field of the CQE for the first flushed WQE for the RQ of the QP specified by QP_ID.

RQ_Minor_Code (16)

Valid only when UserFlushCode=1. RQ_Minor_Code is reported in *Minor Error Code* field of the CQE for the first flushed WQE for the RQ of the QP specified by QP_ID.

GenerateAE (1)

GenerateAE indicates that an AE should be generated after the CQEs have been generated for the flushed WQEs.

AE_Code (16)

Valid only when GenerateAE=1. See Table 38-450 for the values that should be used for this field.

AE_Source (4)

Valid only when GenerateAE=1. See Section 38.36.2.5 for the values that should be used for this field.

QP_ID

Identifies the QP number targeted for the flush operation. This must be an accelerated QP (UDA or iWARP).

38.36.3.5.5 Manage Accelerated Port Bit Vector (APBV)

This WQE sets or clear the bit in the APBV table for the host NIC, IP address and TCP or UDP port number specified in the WQE. The APBV table is used to filter TCP segments belonging to accelerated iWARP connections, TCP segments, UDP multicast and unicast fragments and datagrams belonging to accelerated UDA QPs. Host software also uses the APBV table to flag inbound TCP packets used for connection setup of accelerated connections.

Table 38-486.CQP Manage Accelerated Port Table WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	RSVD		
8	[63:0]	RSVD		
16	[63:16]	RSVD	[15:0]	Local TCP/UDP Port
24	[63] [62] [61:41] [40:38]	WQE_Valid Add Port RSVD RSVD (AdditionalFragmentCount)	[37:32] [31:0]	OP RSVD
32	[63:0]	RSVD		
40	[63:0]	RSVD		
48	[63:0]	RSVD		
56	[63:0]	RSVD		

OP (6) and WQE_Valid

See Section 38.36.3.3.1.

Add Port (1)

When this bit is set to 0b, the port is deleted from the table, when this bit is set to 1b, the port is added to the table.

Local TCP/UDP port (16)

The local (destination port for inbound packets) TCP/IP or UDP/IP port that is used for accelerated iWARP connections and accelerated UDA UDP unicast and multicast first fragments and datagrams, and UDA TCP streams. Any inbound TCP packet and UDP datagram or UDP/IP first fragment who's destination IP address matches that IP address table and destination TCP/UDP port matches Local TCP/UDP port is filtered through quad hash table. If missed, TCP packet and UDP datagram/fragment is passed to the host NIC, and TCP packet has the apbvt_hit bit set in the CQE.

38.36.3.5.6 NOP Descriptor Format

This WQE can be used to generate a CQE that can be used to trigger subsequent processing in a completion handler or as a synchronization mechanism to indicate when all previously issued CQP requests have completed.

Table 38-487.CQP NOP WQE Format (Sheet 1 of 2)

Byte Offset	[Bit Range]Field Name	
0	[63:0]	RSVD
8	[63:0]	RSVD
16	[63:0]	RSVD

Table 38-487.CQP NOP WQE Format (Sheet 2 of 2)

Byte Offset	[Bit Range]Field Name			
24	[63] [62:41] [40:38]	WQE_Valid RSVD RSVD (AdditionalFragmentCount)	[37:32] [31:0]	OP RSVD
32	[63:0]	RSVD		
40	[63:0]	RSVD		
48	[63:0]	RSVD		
56	[63:0]	RSVD		

OP (6) and WQE_Valid

See Section 38.36.3.3.1.

38.36.3.5.7 Manage Quad Hash Table Descriptor Format

The quad hash table entry can be configured to forward packets to the user-space direct access QP. Multiple entries of the quad hash table can refer to the same QP or multicast group. This CQP operation should be used to program quad hash tables associated with the user-space direct access QPs only.

The same quad hash table is intended to be shared by iWARP QPs and UDA QPs and UDA multicast groups. The quad hash table must be properly sized to allow a good hit rate. It also must be properly sized with respect to the maximum number of entries that are intended to be allocated. One approach is to limit the total number of resources that can be allowed to use quad hash for the given function. This would enable the best use of the shared table. In any case, since allocation of entries in the quad hash table is controlled by PCI function driver, sizing of the table, and allocation policy should be controlled there.

UDA TCP traffic should be using quad hash tables associated with respective PCI function (PF or VF). The manage quad hash table CQP command is allowed to be used on CQP belonging to PF only.

The quad hash table index is calculated by hardware. All hash functions include destination MAC address and VLAN (if valid) in addition to the description provided later. Hash function inputs and a kind of hash function depends on a hash table entry type:

- TCP SYN (ACK Clear) — For the TCP packets with the *SYN* bit set, a hash function should be used as an input a destination IP address and destination TCP port. A hash table entry should match both destination IP address and destination TCP port, and should be of TCP SYN (ACK clear) entry type. Matching entry should carry a destination QP number.
- Established TCP — For the TCP packets with the *SYN* bit clear or the *SYN* bit set and ACK bit set, the hash function should be used as an input source and destination IP addresses, and source and destination TCP ports. The hash table should match both source and destination IP addresses, and source and destination TCP ports. Matching entry should carry a destination QP number.

Table 38-488.CQP Manage Quad Hash Table WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:48] [47:40] [39:32] [31:24]	RSVD Dest_MAC_Address[5] Dest_MAC_Address[4] Dest_MAC_Address[3]	[23:16] Dest_MAC_Address[2] [15:8] Dest_MAC_Address[1] [7:0] Dest_MAC_Address[0]	
8	[63:50] [49:32]	RSVD QPN	[31:16] [15:0]	src_port dest_port
16	[63:44] [43:32]	RSVD VLAN_ID	[31:10] [09:0]	RSVD QS_Handle
24	[63] [62:61] [60] [59] [58:45] [44:42] [41] [40:38]	WQE_Valid ManageEntry IPv4_Valid VLAN_Valid RSVD EntryType RSVD RSVD (AdditionalFragmentCount)	[37:32] [31:0]	OP RSVD
32	[63:32]	Src_IP_Address_2	[31:00]	Src_IP_Address_3
40	[63:32]	Src_IP_Address_0	[31:00]	Src_IP_Address_1
48	[63:32]	Dest_IP_Address_2	[31:00]	Dest_IP_Address_3
56	[63:32]	Dest_IP_Address_0	[31:00]	Dest_IP_Address_1

OP (6) and WQE_Valid

See Section 38.36.3.3.1.

QS_Handle (10)

Identifies the queue set handle associated with the hash entry.

Dest IP address 0 (32)

This field specifies bits 127 through 96 of the IPv6 IP addresses. Reserved when IPv4 is set (1b). The MSB of this field (bits 31:24) is the first byte on the wire for this field.

Dest IP address 1 (32)

This field specifies bits 95 through 64 of the IPv6 IP addresses. Reserved when IPv4 is set (1b).

Dest IP address 2 (32)

This field specifies bits 63 through 32 of the IPv6 IP addresses. Reserved when IPv4 is set (1b).

Dest IP address 3 (32)

This field specifies the 32-bit IPv4 (see RFC 791) or the least significant 32-bits of the IPv6 IP address. The LSB of this field (bits 7:0) is the last byte on the wire for this field.

Src IP address 0 (32)

This field specifies bits 127 through 96 of the IPv6 IP addresses. Reserved when IPv4 is set (1b). The MSB of this field (bits 31:24) is the first byte on the wire for this field.

Src IP address 1 (32)

This field specifies bits 95 through 64 of the IPv6 IP addresses. Reserved when IPv4 is set (1b).

Src IP address 2 (32)

This field specifies bits 63 through 32 of the IPv6 IP addresses. Reserved when IPv4 is set (1b).

Src IP address 3 (32)

This field specifies the 32-bit IPv4 (see RFC 791) or the least significant 32-bits of the IPv6 IP address. The LSB of this field (bits 7:0) is the last byte on the wire for this field.

src_port(16)

Source TCP port.

dest_port (16)

Destination TCP port.

QPN (18)

Index of the QP context associated with the entry.

EntryType (3)

The quad hash table is used as a perfect filter for multiple types of accelerated traffic. Depending on the traffic type different hardware accelerations might apply.

0 = iWARP hardware acceleration — not allowed to be used by this WQE. Allocation of a hash table entry with accelerated iWARP QP should be done using Modify QP WQE.

1 = Userspace TCP Established

2 = Userspace TCP SYN (ACK clear)

IPv4_Valid (1)

When set (1b) indicates that both source and destination IP addresses are IPv4. When clear (0b) indicates that both source and destination IP addresses are IPv6.

ManageEntry (2)

0 = Delete entry from the table

1 = Add entry to the table

2 = Modify entry

3 = Reserved

Dest_MAC_Address[5-0] (8 bits each)

Specifies the MAC address associated with the destination IP address. Index 0 is the LSB and byte 5 is the MSB.

VLAN_ID (12 bits)

Specifies the VLAN ID associated with the hash entry. This field is ignored unless VLAN_Valid is set (1b).

VLAN_Valid (1 bit)

Specifies if the hash entry has a VLAN associated with it. If this field is clear (0b), then the hash entry is not associated with a VLAN. If set (1b), then the VLAN associated with the hash entry is specified in the VLAN ID field.

38.36.3.5.8 Suspend QP Descriptor Format

This WQE can be used to suspend a QP that is to be moved to a different QS that the one to which it is currently assigned. QP suspension is required before changing the QS. This WQE might return suspend pending in which case an AE_QP_SUSPEND_COMPLETE AE is issued when the suspend operation completed.

Table 38-489.CQP Suspend QP WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	RSVD		
8	[63:0]	RSVD		
16	[63:0]	RSVD		
24	[63] [62:41] [40:38]	WQE_Valid RSVD RSVD (AdditionalFragmentCount)	[37:32] [31:18] [17:0]	OP RSVD QP_ID
32	[63:0]	RSVD		
40	[63:0]	RSVD		
48	[63:0]	RSVD		
56	[63:0]	RSVD		

OP (6) and WQE_Valid

See Section 38.36.3.3.1.

QP_ID (18)

Index of the QP to be suspended.

38.36.3.5.9 Resume QP Descriptor Format

This WQE can be used to resume a QP that had been previous suspended in order to be moved to a different QS that the one to which it was assigned. QP suspension is required before changing the QS.

Table 38-490.CQP Resume QP WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	RSVD		
8	[63:0]	RSVD		
16	[63:10]	RSVD	[9:0]	QS_Handle
24	[63] [62:41] [40:38]	WQE_Valid RSVD RSVD (AdditionalFragmentCount)	[37:32] [31:18] [17:0]	OP RSVD QP_ID
32	[63:0]	RSVD		
40	[63:0]	RSVD		
48	[63:0]	RSVD		
56	[63:0]	RSVD		

OP (6) and WQE_Valid

See Section 38.36.3.3.1.

QP_ID (18)

Index of the QP to be suspended.

QS_Handle (10)

This field specifies Tx scheduler QS handle associated with the TC for this QP. For VFs, the QS_Handle is checked to ensure that the VF issuing the CQP command is associated with the QS_Handle.

38.36.3.5.10 Static HMC Resources Allocated Descriptor Format

This WQE must be used after the commit FPM values operation is complete and all of the static HMC resources from Table 38-452 have HMC backing pages allocated and configured. CQP then completes initializing the function for PE operation.

Table 38-491. Static HMC Resources Allocated WQE Format

Byte Offset	[Bit Range]	Field Name		
0	[63:0]	RSVD		
8	[63:0]	RSVD		
16	[63:6]	RSVD	[5:0]	HMC_FCN_ID (PF only)
24	[63] [62:41] [40:38]	WQE_Valid RSVD RSVD (AdditionalFragmentCount)	[37:32] [31:0]	OP RSVD
32	[63:0]	RSVD		
40	[63:0]	RSVD		
48	[63:0]	RSVD		
56	[63:0]	RSVD		

OP (6) and WQE_Valid

See Section 38.36.3.3.1.

HMC_FCN_ID (6)

This field specifies the HMC function ID that software has completed the allocation and configuration of the static HMC backing pages. For PFs, the HMC_FCN_ID is strictly the PF index. For VFs, the HMC function ID was returned from the manage HMC PM function CQP operation. If a CQP instance associated with a VF submits this operation, this field is ignored and the HMC_FCN_ID is determined by firmware.

38.36.4 iWARP Functionality

The operations for RDMA via iWARP are implemented using the context and WQE formats described in the following sections. QP operation, the verbs interface, and system view of the 10 GbE controller are described in Section 38.36.2.1. The CQP operations and HMC structures necessary to bring the 10 GbE controller to a functional state for iWARP are described in Section 38.36.3.

38.36.4.1 iWARP Q2 Area

iWARP QPs have a memory area that is written under error conditions. The format of this area is listed in Table 38-492. This memory area is reference from the Q2_Address variable from the QP context listed in Table 38-493.

Table 38-492.iWARP Q2 Structure Format

Byte Offset	[Bit Range]Field Name			
0 - 55	Outbound_Terminate_Header			
64	[63] Sequence_Update_Toggle [62:32]	RSVD	[31:0]	First_Partial_Sequence_Number
72 -255	Inbound_Q2_Data			

Outbound_Terminate_Header (56 bytes)

This field contains the outbound terminate header used for Modify QP CQP operations when Terminate_Actions is set to send terminate only or send fin and terminate.

First_Partial_Sequence_Number(32 bits)

This field contains the sequence number of the first partial FPDU receive by the 10 GbE controller. See Section 38.36.2.3 for more details on partial FPDU support.

Sequence_Update_Toggle(1 bit)

This field toggles each time the First_Partial_Sequence_Number is updated by the 10 GbE controller. See Section 38.36.2.3 for more details on partial FPDU support.

Inbound_Q2_Data(184 bytes)

This field contains the first 184 bytes of the packet that caused an AE. This could either be an inbound terminate message or the Ethernet packet that had an error.

38.36.4.2 iWARP QP Context Format

iWARP QP context is used by software to initialize or update the 10 GbE controller QP context. Create and Modify QP CQP operations (see Section 38.36.3.3.2) reference this structure.

Table 38-493.iWARP QP Context Structure Format (Sheet 1 of 3)

Byte Offset	[Bit Range]Field Name			
0	[63:62]	iwarp_rdmap_ver	[13:12]	limit
	[61:48]	RSVD	[10]	RSVD
	[47]	Push Mode Enable	[9:8]	RQ_WQE_Size
	[46:42]	RSVD	[7]	timestamp
	[41:32]	Push Page Index	[6]	
	[31]	SQ_TPH_en	[5]	Insert VLAN Tag
	[30]	RQ_TPH_en	[4]	NoNagle[1]
	[29]	XMIT_TPH_en	[3]	IPv4
	[28]	RCV_TPH_en	[2]	RSVD
	[27]	RSVD(SQ_HDR_TPH_en)	[1:0]	iwarp_ddp_ver[2]
	[26]	RSVD(RQ_HDR_TPH_en)		
	[2:20]	RSVD		
	[19]	RSVD		
	[18:16]	dupack_thresh		
	[15]	drop_out_of_order_seg		
	[14]	ECN_enable		
8	[63:0]	SQ_Address		
16	[63:0]	RQ_Address		

Table 38-493.iWARP QP Context Structure Format (Sheet 2 of 3)

Byte Offset	[Bit Range]	Field Name	[Bit Range]	Field Name
24	[63:48] [47:32] [31:24] [22] [23] [21:16]	Dest Port Number Source Port Number Traffic Class or TOS[2] RSVD avoid_stretch_ack Source MAC Address Index	[15:12] [11:8] [7:0]	SQ Size RQ Size Hop Limit or TTL[2]
32	[63:32]	Dest_IP_Address_2	[31:0]	Dest_IP_Address_3
40	[63:32]	Dest_IP_Address_0	[31:0]	Dest_IP_Address_1
48	[63:48] [47:32] [31:30]	ARP Index VLAN Tag[2] RSVD	[29:16] [15:14] [13:0]	snd_mss RSVD rsvd (rcv_mss)
56	[63] [62:48] [47:44] [43:40] [39:36] [35:32]	RSVD pd_index[3] RSVD Snd_wscale RSVD Rcv_wscale	[31:28] [27:24] [23] [22] [21] [20] [19:00]	TCP_state RSVD ignore_tcp_uns_options ignore_tcp_options keepalive wscale Flow Label[2]
64	[63:32] [31: 24]	RSVD Keepalive Interval	[23:16] [15:00]	kalive_timer_max_probes RSVD
72	[63:32]	timestamp_age	[31:0]	timestamp_recent
80	[63:32]	snd_wnd	[31:0]	snd_nxt
88	[63:32]	rcv_wnd[2]	[31:0]	rcv_nxt
96	[63:32]	snd_una	[31:0]	snd_max
104	[63:32]	rtt_var	[31:0]	srtt
112	[63:32]	cwnd	[31:0]	ss_thresh
120	[63:32]	snd_wl2	[31:0]	snd_wl1
128	[63:54] [53:48]	RSVD rexmit_thresh	[47:46] [45:32] [31:0]	RSVD err_RO_ index max_snd_window
136	[63:49] [48:32]	RSVD RxCmpQueueNum	[31:17] [16:0]	RSVD TxCmpQueueNum
144	[63:8]	Q2_Address	[7:0]	RSVD
152	[63:47] [46:32]	RSVD	[8] [7:0]	RSVD last_byte_sent
160	[63:57] [56:48] [47:41] [40:32] [31] [30] [29] [28] [27] [26]	RSVD snd_mrk_offset[3] RSVD rcv_mrk_offset[3] rcv_no_mpa_crc[3] assume_aligned_headers[3] Receive Markers[3] iWARP Mode[3] RSVD RSVD	[25] [24] [23] [22] [21] [20] [19:18] [17:16] [15:7] [6:0]	PrivilegedEnable[3] FastRegisterEnable[3] BindEnable[3] Send Markers[3] rdmard_ok[3] rdmawr_rdresp_ok[3] RSVD IRD_Size[3] RSVD ORD_Size[3]
168	[63:8]	QP Completion Context	[7:0]	RSVD

Table 38-493.iWARP QP Context Structure Format (Sheet 3 of 3)

Byte Offset	[Bit Range]Field Name			
176	[63:50] [49:32] [31:26]	RSVD Exception_UDA_Queue RSVD	[25:16] [15:8] [7:0]	QS_Handle RQ_TPH_value SQ_TPH_value
184	[63:32]	Local_IP_Address_2	[31:0]	Local_IP_Address_3
192	[63:32]	Local_IP_Address_0	[31:0]	Local_IP_Address_1
200-248	[63:0]	RSVD		

Notes:

1. This variable is a cached variable. See Cached_Variables_Valid in Modify QP operations for more details on cached context variables.
2. Only valid for QPs.

Push page index (10)

This field identifies the push page associated with the QP. For VFs, the push page index is function relative.

Push mode enable (1)

This field indicates if push mode is enabled for the QP.

ignore_tcp_options (1)

Specifies that the TCP options should be ignored when processing received TCP headers.

ignore_tcp_uns_options (1)

Specifies that the unsupported TCP options should be ignored when processing received TCP headers.

NoNagle (1)

This field indicates if nagle algorithm is in use on this connection. If this field is set, then the nagle algorithm is disabled on this connection (RFC 896).

PrivilegedEnable (1)

This bit is used to enable privilege mode on STag's. If this bit is set, then a local STag of zero (STag 0) is allowed on this connection and the *TO* field should be treated as a physical address. If this bit is cleared, then STag 0 is NOT allowed for a local STag and an AEQE is generated with privilege error indicated. STag 0 is never allowed in an inbound RDMA packet.

FastRegisterEnable (1)

This bit is used to enable fast register opcodes on privilege mode QP's. If this bit is set, then a fast register is allowed on this connection. If this bit is cleared, then fast register is NOT allowed for this connection and an AEQE is generated with privilege error indicated.

BindEnable (1)

This bit is used to enable memory window bind operations on this QP. If this bit is set, then memory window bind operations are allowed on this connection. If this bit is cleared, then memory window bind operations are NOT allowed on this connection and an AEQE is generated with privilege error indicated.

ReceiveMarkers (1)

This field specifies if we are to receive MPA markers in the receive stream. If set to 0b, then no markers are expected in the receive stream; if set to 1b, then markers are expected every 512 bytes.

SendMarkers (1)

This field specifies if generating MPA markers in the transmit stream. If set to 1b, then markers are expected every 512 bytes; if set to 0b, then markers are not included in the transmit stream.

rdmard_ok (1)

This field enables inbound iWARP RDMA read requests when set. If cleared then this field disables inbound iWARP RDMA read requests and an AEQE is generated.

rdmawr_rdresp_ok (1)

This field enables inbound iWARP RDMA write requests and inbound RDMA read responses when set. When cleared, this field disables inbound iWARP RDMA write responses and inbound iWARP RDMA read requests and an AEQE is generated.

timestamp (1)

This field indicates if time stamp option is in use on this connection. If this field is set, then the time stamp option is enabled on this connection (RFC 1323); if cleared, the time stamp option is not present.

keepalive (1)

This field indicates if the keep alive option is in use on this connection. If this field is set, then the keep-alive option is enabled on this connection (RFC 1122).

wscale (1)

This field indicates if window scale option is in use on this connection. If this field is set, then the window scale option is enabled on this connection in both directions (RFC 1323). The window is scaled by the snd_wscale and rcv_wscale variables described later. If clear then both windows are not scaled.

dupack_thresh (3)

Specifies the number of dupacks received before starting fast retransmit (1-7). A value of 0 is remapped to a default value of 3.

err_RQ_index_valid (1)

Indicates that the err_RQ_index is valid. This setting is allowed only for a modify QP-to-terminate or a modify QP-to-error operation.

err_RQ_index (14)

Software uses err_RQ_index to override the index in the RQ during a FlushWQEs CQP operation. This is only valid if err_RQ_index_valid is on and it is a modify QP-to-terminate or modify QP-to-error operation.

drop_out_of_order_seg (1)

If this bit is set and a segment is received that is out-of-order, TRX drops the segment and sends an acknowledgment as required by TCP. If this bit is zero, the out-of-order segment is processed.

ECN_enable (1)

When this bit is clear (0b), ECN is disabled. When this bit is set (1b), then the TOS field of the IPv4 header or IP TC field of the IPv6 header contains the ECN code point (bits 6 and 7 - least significant bits) and the code point in the IP TC or TOS context field must be set to 01b (normal) or code point 10b (optional).

XMIT_TPH_en (1)

If set (1b), TPH is enabled for data reads associated with this QP. If clear (0b), THP is not used for data reads associated with this QP.

SQ_TPH_en (1)

If set (1b), TPH is enabled for the SQ of this QP. If clear (0b), THP is not used for this resource.

SQ_TPH_value (8)

If SQ_TPH_en is set (1b), TPH STag associated with SQ operations is initialized with SQ_TPH_value. If SQ_TPH_en is clear (0b), this field is ignored.

SQ_Size (4)

This field encodes the maximum size for the WQ. The encoding of the SQ sizes are $4*2^{SQ\_Size}$ in terms of 32-byte quanta of memory. The following value are allowed for SQ_Size:

0 = 128 bytes
1 = 256 bytes
2 = 512 bytes
3 = 1024 bytes
4 = 2048 bytes
5 = 4096 bytes
6 = 8192 bytes
7 = 16384 bytes
8 = 32768 bytes
9 = 65536 bytes
10 = 131072 bytes
11 = 262144 bytes
12 = 524288 bytes

Software can only allocate N-1 WQEs on the SQ. Each WQE is variable in size and can consume up to 128 bytes of memory. For more information see Section 38.36.2.1.7. The minimum size for an SQ is four WQEs of the maximum size that will be used.

RCV_TPH_en (1)

If set (1b), TPH is enabled for data placement associated with this QP. If clear (0b), THP is not be used for data placement associated with this QP.

RQ_TPH_en (1)

If set (1b), TPH is enabled for the RQ of this QP. If clear (0b), THP is not be used for this resource.

RQ_TPH_value (8)

If RQ_TPH_en is set (1b), TPH STag associated with RQ operations is initialized with RQ_TPH_value. If RQ_TPH_en is clear (0b), this field is ignored.

RQ_Size (4)

This field encodes the maximum size for the WQ. The encoding of the RQ sizes are $4*2^{RQ\_Size}$ in terms of 32-byte quanta of memory. The following value are allowed for RQ_Size:

- 0 = 128 bytes
- 1 = 256 bytes
- 2 = 512 bytes
- 3 = 1024 bytes
- 4 = 2048 bytes
- 5 = 4096 bytes
- 6 = 8192 bytes
- 7 = 16384 bytes
- 8 = 32768 bytes
- 9 = 65536 bytes
- 10 = 131072 bytes
- 11 = 262144 bytes
- 12 = 524288 bytes

The actual number or WQEs that can be posted to the RQ is the size of the WQ divided by the WQE size determined from RQ_WQE_Size. Software can only submit N-1 WQEs to a WQ without processing completions for the WQ without exposing the possibility of a WQ overflow. WQ overflow results in indeterministic behavior for the affected WQ. The minimum size for an RQ is four WQEs.

RQ_WQE_Size (2)

Specifies the number of 32 byte chunks of memory included with each RQ WQE. The maximum number of additional fragments allowed for an RQ WQE is 6 for a total of 7 fragments and a maximum WQE size of 128 bytes. Valid values are:

- 0 = 32 bytes per WQE (no additional fragments)
- 1 = 64 bytes per WQE (1 or 2 additional fragments)
- 2 = 128 bytes per WQE (3 to 6 additional fragments)
- 3 = Reserved

Insert VLAN tag (1)

This bit is set to enable VLAN processing on a connection. The tag configured in the VLAN_Tag field is used for all processing. If this bit is clear then there is no VLAN insertion or removal performed by the PE. Additional VLAN and priority setting can be configured through the VSI associated with the QP. The most significant portion of the VLAN tag carry the user specified priority. When this bit is set, received packets destined for the QP with mis-matched VLAN or no VLAN tag is sent to LAN queues. When this bit is clear, received packets destined for the QP with VLAN tags are sent to LAN queues.

Insert L2TAG2 (1)

This bit is set to enable L2TAG2 processing for this connection.

IPv4 (1)

This field indicates if the QP is IPv4 or IPv6. 1b indicates IPv4 and 0b indicates IPv6.

iwarp_ddp_ver (2)

These bits are used to set and check the DV fields of DDP PDUs for this connection.

iwarp_rdmap_ver (2)

These bits are used to set and check the RV fields of RDMAP PDUs for this connection.

RxCmpQueueNum (17)

This field specifies which of the 128 K CQs is used for receive completion notification. The Rx and Tx completions can be mapped to the same CQ or different queues.

TxCmpQueueNum (17)

This field specifies which of the 128 K CQs is used for transmit completion notification.The Rx and Tx completions can be mapped to the same CQ or different queues.

SQ address(64)

If VirtualWQ bit is clear, then this field holds SQ base physical address. It must be aligned to an address divisible by 128 bytes. If VirtualWQ bit is set, then this field specifies the first HMC PBLE index of the 1-level page list for the SQ (first first_pm_pbl_index).

RQ address (64)

If VirtualWQ bit is clear or the QP is associated with a shared RQ, then this field holds RQ base physical address. It must be aligned to an address divisible by 128 bytes. This is the RQ base pointer when this is an iWARP accelerated connection. If VirtualWQ bit is set and the QP is not associated with a shared RQ, then this field specifies the first HMC PBLE index of the 1-level page list for the SQ (first first_pm_pbl_index).

IP TC or TOS (8)

This field specifies the IPv4 type of service bits (RFC 2474). If these bits represent the IPv4 TOS bits, then only the lower 4-bits are valid. These bits are set by software when the connection is created and transmitted in the IP header of all sent datagrams for this connection.

rexmit_thresh (6)

Specifies the number of re-transmissions on this connection that can occur before the AE_LLP_TOO_MANY_RETRIES AE is generated. A value of 0b disables generation of the AE and allows infinite retries.

avoid_stretch_ack (1)

When this bit is set, an ACK is generated ever 2*MSS sequence numbers from cur_ack_seq_num up to current rcv_nxt value. When this bit is off, then ACKs can be coalesced if the 10 GbE controller is busy. In the 10 GbE controller, this works for all cases except when an immediate ACK needs to be generated in which case the ACK sequence number my advance by more than 2*MSS segments. This field is normally clear for iWARP connections.

Source MAC address index (6)

This field specifies an index into one of the 64 possible source MAC addresses. This is used to specify the outbound MAC address in the Ethernet header.

QS_Handle (10)

This field specifies Tx scheduler Queue Set handle associated with the TC for this QP. For VFs, the QS_Handle is checked to ensure that the VF issuing the CQP command is associated with the QS_Handle.

Exception_UDA_Queue (18)

This field specifies UDA queue that receives partial iWARP FPDUs and TCP/IP packets with the *URG* bit set.

Limit (2)

This field specifies a limit value for the number of bytes that increase cwnd for each received acknowledgment - as per RFC3465. If an acknowledgment is for less than limit, then cwnd is advanced by that number of bytes. Generally, this field should be set to 3 for iWARP connections; other values are for interoperation with various TCP offload engines:

0 = cwnd updated by at most 1*SMSS bytes.

1 = cwnd updated by at most 2*SMSS bytes.

2 = cwnd updated by at most 4*SMSS bytes (experimentation).

3 = cwnd updated by the number of bytes Acknowledged (experimentation).

if limit = 3 (cwnd += ACK_SEQ - snd_una)

else cwnd += min(ACK_SEQ - snd_una, SMSS<<limit)

Hop limit or TTL (8)

This field specifies the IPv4 Time-To-Live (TTL) parameter in the IP header (RFC 791). It is initialized by software.

Dest port number (16)

This field specifies the destination TCP port number for the TCP header (RFC 793).

Source port number (16)

This field specifies the source TCP port number for the TCP header (RFC 793).

Dest IP address 0 (32)

This field specifies bits 127 through 96 of the IPv6 IP addresses. Reserved when IPv4 is set (1b). The MSB of this field (bits 31:24) is the first byte on the wire for this field.

Dest IP address 1 (32)

This field specifies bits 95 through 64 of the IPv6 IP addresses. Reserved when IPv4 is set (1b).

Dest IP address 2 (32)

This field specifies bits 63 through 32 of the IPv6 IP addresses. Reserved when IPv4 is set (1b).

Dest IP address 3 (32)

This field specifies the 32-bit IPv4 (see RFC 791) or the least significant 32-bits of the IPv6 IP address. The LSB of this field (bits 7:0) is the last byte on the wire for this field. For IPv4 addresses, the MSB (bits 31:24) of this field is the first byte of the destination IP address on the Ethernet wire.

Local IP address 0 (32)

This field specifies bits 127 through 96 of the IPv6 IP addresses. Reserved when IPv4 is set (1b). The MSB of this field (bits 31:24) is the first byte on the wire for this field.

Local IP address 1 (32)

This field specifies bits 95 through 64 of the IPv6 IP addresses. Reserved when IPv4 is set (1b).

Local IP address 2 (32)

This field specifies bits 63 through 32 of the IPv6 IP addresses. Reserved when IPv4 is set (1b).

Local IP address 3 (32)

This field specifies the 32-bit IPv4 (see RFC 791) or the least significant 32-bits of the IPv6 IP address. The LSB of this field (bits 7:0) is the last byte on the wire for this field. For IPv4 addresses, the MSB (bits 31:24) of this field is the first byte of the destination IP address on the Ethernet wire.

snd_mss (14)

Maximum segment size the sender is allowed to transmit. The 10 GbE controller snd_mss is often set to values smaller than the maximum based on the capabilities of the fabric or connection partner. It can change based on the minimum of the rcv_mss, PathMTU or Ethernet MTU. This field needs to be set to a value less any TCP options so rather than setting the value to 1460 if the time stamp option is enabled it would be set to 1448.

ARP index (16)

Index into the ARP cache to specify the Ethernet MAC address to use for this connection. Initialized by software during connection establishment. Allows access to 65536 entry ARP table.

VLAN tag (16)

Specifies one of the 4096 VLAN tags for this connection, 3-bits of priority and 1-bit canonical format. All bits are valid. The VLAN is in the lower 12-bits of the VLAN *Tag* field. The upper 3 bits are priority.

L2TAG2 (16)

This 16 bit tag will be inserted into outbound packets based on the rules define for LAN if Insert_L2TAG2 is set.

TCP_state (4)

Here are the state definitions (RFC 793):

0 = NON EXISTANT (Software state)
1 = CLOSED
2 = LISTEN (Software state)
3 = SYN_SENT (Software state)
4 = SYN_RECEIVED (Software state)
5 = ESTABLISHED
6 = CLOSE_WAIT
7 = FIN_WAIT_1
8 = CLOSING (Software state)
9 = LAST_ACK (Software state)
10 = FIN_WAIT_2
11 = TIME_WAIT (Software state)
15-12 = Reserved

Flow label (20)

This field specifies the IPv6 *Flow Label* field contents (RFC 2460). This field is set by software when a connection is offloaded and can be changed during operation.

pd_index (15)

Protection domain for this context. This specifies which one of the 32 BK iWARP protection domains this connection context belongs too. There are no reserved index values.

Snd_wscale (4)

This field specifies the shift count used on the advertised window size received from the other end to obtain the real 32-bit advertised window size (AWS). A value of 0 selects no window scaling. A value of 14 supports a maximum window of 1,073,725,440 bytes [(65536 x 16384) - 1) (RFC 1323]. Software initializes this field from the value supplied on the during connection establishment (such as SYN segment). Valid values are: 0-14 supporting windows up to 1 GB -1. A value of 15 is rolled back to a value of 14.

Rcv_wscale (4)

This field specifies the shift count used every time TCP sends a segment to determine the window size to report. The internal 32-bit window size is right shifted by this field to give the value to place in the TCP header. A value of 0 selects no window scaling. A value of 15 supports a maximum window of 1,073,725,440 bytes ((65536 x 16384) - 1) (RFC 1323). Software initializes this field from the value it supplied during connection establishment (like on the SYN segment) based on the receive buffer size. Valid values are: 0-14 supporting a receive window up to 1 GB -1. A value of 15 is rolled back to a value of 14.

Keepalive interval (8 bits):

This field specifies the keep-alive interval in terms of timer ticks of the keep-alive timer.

kalive_timer_max_probes (8)

This field specifies the maximum number of keep-alive probes that are allowed before declaring issuing the AE_LLP_TOO_MANY_KEEPALIVE_RETRIES AE.

timestamp_recent (32)

This field is updated when a segment arrives that includes the expected segment number (RFC 1323). This field is returned in the *Echo Reply* field of a timestamp.

timestamp_age (32)

This field records the value from tcp_now (500 ms timer) the last time ts_recent was copied from a receive segment (RFC 1323). This field is required for TCP to perform PAWS.

snd_nxt (32)

The next sequence number that TCP sends in a transmitted packet (RFC 793).

snd_wnd (32)

This field specifies the senders Advertised Window Size (AWS). This field is the 16-bit receive window shifted by snd_wscale to come up with the AWS (RFC 793). This is a window our partner advertised that limits our transmit operations.

rcv_nxt (32)

This is the next expected receive sequence number. It is also referred to as the left hand receive pointer (RFC 793).

rcv_wnd (32)

This field specifies the receivers AWS. The AWS is flow control imposed by the receiver. This field is used to generate the 16-bit window size reported in the TCP header, shifted by the receive window scale (rcv_wscale). This is the value advertised to the sender (RFC 793).

snd_max (32)

This sequence number points to the maximum send sequence number that has been transmitted. This field sets the upper bound of valid ACK when processing a re-transmission timeout (RFC 793). On connection offload this field is normally set to the same sequence number as snd_nxt unless a retransmission was in progress when the connection was offloaded.

snd_una (32)

This sequence number points to the oldest unacknowledged sequence number for this connection (RFC 793).

srtt (32)

This field is the smoothed round trip time (rtt) << 3 (RFC 793 and RFC 2988) in ms. If this field is set to 0b on connection offload then the hardware calculates the value.

rtt_var (32)

Round trip variation (RTTVAR in RFC 2988) in ms. This field MUST never be set to 0b if srtt is also set to 0b. Make this a non-zero value if srtt is 0b.

ss_thresh (32)

Slow start threshold (RFC 2581).

cwnd (32)

This field is the sender congestion widow. It gets incremented by MSS when an ACK arrives. Software should initialize this field to 1b segment size (in bytes). If a duplicate ACK arrives, this is set to min(cwnd, rcv_wnd) but at least 2 segments. On a transport timeout this is set to 1b segment (RFC 2581).

snd_wl1 (32)

Segment sequence number used for last send window update (RFC 793).

snd_wl2 (32)

Segment acknowledgment used for last send window update (RFC 793).

max_snd_window (32)

Indicates the largest send window advertised by the remote peer. Used in step 3 of the sender side SWS algorithm. If at least ½ of the largest window seen so far is available, then send a segment.

retransmitCount (6)

The number of re-transmits that have been sent when the connection is offloaded (RFC 2581).

Q2 Address (56)

This field points to a buffer in host memory that is used to report a terminate message received on this QP. The field can be located at a fixed offset based on some other pointer (receive base pointer); it only points to a one entry queue to handle errors (terminate requests).

last_byte_sent (8)

This is the last byte transmitted on the wire. It is only used during window probes. During modified iWARP only window probes, the 10 GbE controller transmits the last acknowledged data byte on the wire. This corresponds to the last byte sent because a zero window condition is defined as all bytes acknowledged but no window to continue transmitting. Software should initialize it to the last byte sent before the connect was

offloaded the connection to the hardware, so if a streaming mode message had been sent, it's the last byte sent of the last MPA message sent. If after a SYN/ACK handshake then initialize this field to 0b.

rcv_no_mpa_crc (1)

This field specifies if we are to check MPA CRCs in the receive stream. If set to 1b, then no MPA CRC checks are performed on FPDUs in the receive stream; if set to 0b, then all inbound FPDUs are checked by the 10 GbE controller.

assume_aligned_headers (1)

This field is set by software during connection setup if it knows the other adapter is a the 10 GbE controller that does not generate unaligned headers. This enables the 10 GbE controller to place all segments that arrive if it passes basic header checks even if the packet was received out of order. Markers are not examined.

iWARP mode (1)

This field sets the mode of operation when RQ empty conditions are encountered. When this field is clear (0b) the connections is placed in the terminate state and an AE is generated. When this field is set (1b) the offending packet is dropped and the peer re-transmits the packet in the hope that the empty condition is resolved by the time the re-transmission occurs.

IRD_Size (2)

This field specifies the number of inbound RDMA resources available for this connection (Q1). Valid settings are:

00b = 4 WQE

01b = 16 WQE

10b = 64 WQE

11b = 128 WQE

When advertising resources to the other side only 1/2 the queue is available. For example, if IRD_Size is set to 64 entries, then the ORD size for the connection partner should be set to 32 or less.

ORD_Size (7)

This field specifies the number of outbound RDMA resources available for this connection. Up to 127 outbound RDMA read requests are supported. This field is not encoded because a partner might support a different number of RDMA read requests than on transmit (such as 1, or 10, or 64, etc.).

snd_mrk_offset (9)

This field specifies the offset to the MPA marker in the transmit stream of bytes. Subsequent markers are located every 512 bytes from this location. So modulo math can be used to place the markers in the transmitted byte stream. Software initializes this field upon connection setup.

rcv_mrk_offset (9)

This field specifies the offset to the MPA marker in the receive stream of bytes. Subsequent markers are located every 512 bytes from this location. So modulo math can be used to extract the markers in the receive byte stream. Software initializes this field upon connection setup.

QP_Completion Context (64)

This field is reported in CQEs and also in AEQEs.

Use_Statistics_Instance (1)

This field indicates if the default per PMF statistics are used or if one of the additional RDMA statistics instances are used for this QP. When this field is zero, the default statistics are used. When this field is one, the statistics instance indicated by the Statistics_Instance_Index field is used.

Statistics_Instance_Index (5)

This field specifies which of the additional RDMA statistics indexes are used if Use_Statistics_Instance is one. This field is ignored if Use_Statistics_Instance is zero.

t_high

The high RTT threshold in ms for TIMELY If the current RTT is higher than this threshold then the congestion window is multiplicatively decreased. If zero, the default value of 500 ms is used. This value applies only if TimelyEnable is set (1b).

t_low

The low RTT threshold in ms for TIMELY If the current RTT is less than this threshold then the congestion window is additively increased. If zero, the default value of 50 ms is used. This value applies only if TimelyEnable is set (1b).

38.36.4.3 iWARP QP Completion Codes

iWARP QP errors are typically reported via AEs with the 10 GbE controller. After fielding the AEs, software might issue the flush WQES CQP operation (see Section 38.36.3.5.4) to complete any pending operations. Table 38-494 lists the completion codes reported by the 10 GbE controller in a CQ entry (Table 38-495). Also note that host software can report any completion code necessary using the flush WQEs operation instead of the codes listed in Table 38-494.

Table 38-494.iWARP QP Error Codes

Major Error Code	Minor Error Code	Completion Reason	Description
0x0001	0x0001	WQE Flushed	The WQE has been flushed due to a modify QP state transition.

38.36.4.4 iWARP CQ entry formats

CQ operation for iWARP is described in Section 38.36.2.1.3. CQs are manipulated through CQP operations. See Section 38.36.3.3.3 for further details. Note that an extra 32 bytes of 0's are added to each CQE if the Avoid_Memory_Conflicts is set during the CreateCQ CQP operation.

Table 38-495.iWARP CQ Entry Format Without Immediate Data (Sheet 1 of 2)

Byte Offset	[Bit Range]Field Name			
0	[63:32]	TCP Sequence Number	[31:0]	Payload Length

Table 38-495.iWARP CQ Entry Format Without Immediate Data (Sheet 2 of 2)

Byte Offset	[Bit Range]Field Name			
8	[63:0]	QP Completion Context		
16	[63:50] [49:32]	RSVD QP_ID	[31:0]	Invalidated STag
24	[63] [62] [61:56] [55] [54] [53]	CQE_Valid SQ OP Error Solicited Event (iWARP RQ Only) STag (iWARP RQ Only)	[52] [51] [50:46] [45:32] [31:16] [15:0]	 Push Dropped (iWARP SQ Only) RSVD WQ_Desc_Index Major Error Code Minor Error Code

OP (6)

This field reports the opcode from the operation associated with the CQE.

WQ_Desc_Index (14)

WQ is sliced up into 32-byte descriptor quanta. Every WQE must start with a 32-byte descriptor on a 32-byte boundary. WQ_Desc_Index reports the 32-byte quanta index of the WQE associated with the completion.

CQE_Valid (1)

The CQE_Valid bit for CQE is a bit that indicates that a CQE is ready to be processed. The polarity of the valid bit changes each time the CQ wraps from the last entry back to the first entry. This change in polarity reduces software overhead by avoiding the need to clear the valid bit once software has processed a valid CQE. Software is responsible to clear (set to 0b) all memory in a CQ initially at CQ creation. The first iteration (and subsequent odd numbered iterations) through the CQ, the 10 GbE controller sets the *Valid* bit to 1b when it writes a new CQE. For the second iteration (and all even numbered iterations) through the CQ, the 10 GbE controller sets the valid bit to 0b when it writes an new CQE.

Error (1)

0b = No Error.

1b = An error occurred when processing the WQE associated with this CQE and that the *Error Code* field is valid.

Stag (1)

Indicates if the Invalidated_STag field has valid contents.

0b = Ignore Invalidated_Stag.

1b = An STAG was invalidated and Invalidated_Stag field is valid.

Solicited_Event (1)

Valid for receive only. Value for SQ completions is undefined.

0b = SE bit not set in received packet.

1b = SE bit set in received packet.

SQ (1)

0b = RQ.

1b = SQ.

Push dropped (1)

Valid only when SQ is set.

0b = Push operations are being processed successfully.

1b = A recent push mode operation has been dropped by the 10 GbE controller, software should refrain from submitting additional push mode operations until the SQ has encountered an empty condition.

Payload_Length (31)

Total payload length of the completed message. This field is only valid for RQ WQE completions.

TCP sequence number

For iWARP, this field reports the TCP sequence number associated with the work request specified by the completion.

QP completion context (64)

Completion context pointer. This field is transferred to the CQE from QP context.

QP_ID (18)

QP associated with the completed message.

Invalidated_Stag (32)

Any stag that was invalidated by the completed message. Only valid if the *Stag* bit is also set. If the *Stag* bit is clear, then Invalidated_Stag must be ignored.

Major Error_Code (16)

Valid if error is set (1b). See Table 38-494 for defined values. Software can also report any value via the flush WQEs CQP operation (see Section 38.36.3.5.4).

Minor Error_Code (16)

Valid if error is set (1b). See Table 38-494 for defined values. Software can also report any value via the flush WQEs CQP operation (see Section 38.36.3.5.4).

38.36.4.5 iWARP Descriptor Formats

38.36.4.5.1 iWARP SQ Descriptors

The following WQE formats are used in conjunction with iWARP QPs. Operations that are supported for iWARP QPs are the listed in Table 38-496.

Table 38-496.iWARP QP Operations (Sheet 1 of 2)

Operation Code	Operation Name	Page	Operation Code	Operation Name	Page
0x00	RDMA Write	3249	0x0A	Local Invalidate STag	3252
0x01	RDMA Read	3250	0x0B	RDMA Read with Local Invalidate	3252
0x02			0x0C	NOP	3246
0x03	Send	3247	0x0D		
0x04	Send with Invalidate	3247	0x0E	RSVD	
0x05	Send with Solicited Event	3247	0x0F		
0x06	Send with Solicited Event and Invalidate	3247	0X10		
0x07	RSVD		0x11		
0x08	Memory Window Bind	3251	0x12		
			0x13		

Table 38-496.iWARP QP Operations (Sheet 2 of 2)

Operation Code	Operation Name	Page	Operation Code	Operation Name	Page
			0x14		
			0x15		
0x09	Fast Register memory region	3217	0x12 - 0x3F	RSVD	

Common iWARP SQ Descriptor Format Fields

The basic iWARP WQE is a 32-byte structure that is broken up into 64 bit (8-byte) words. The placement of these fields with WQE are common among all CQP WQEs and also iWARP WQEs. Table 38-497 lists the basic structure of a CQP WQE including the common fields. The definition of fields marked as operation code dependent vary from operation to operation and are detailed in subsequent sections. Fields marked as RSVD (reserved) must be set to 0b or undesired behavior related to the specific QP associated with the WQE might occur. Additionally, iWARP WQEs can optionally include additional descriptors to enable larger WQEs to be created. The additional descriptor format is listed in Table 38-498.

Table 38-497.iWARP Common WQE Fields

Byte Offset	[Bit Range]Field Name		
0	[63:0]	Operation Code Dependent	
8	[63:0]	Operation Code Dependent	
16	[63:0]	Operation Code Dependent	
24	[63] WQE_Valid [62] Signaled Completion [61] Local Fence [60] Read Fence [59] (wait_for_rcvFPDU) Operation Code Dependent [58] Mode) Operation Code Dependent (Streaming [57] Flag) Operation Code Dependent (Inline Data		[56] Push WQE [55 RSVD [54:48] Operation Code Dependent(Inline Data Length) [47] RSVD [46] RSVD [45:41] RSVD [40:38] AdditionalFragmentCount [37:32] OP [31:0] Operation Code Dependent

OP (6)

iWARP operation code. See Table 38-496 for the specific values.

WQE_Valid (1)

The WQE_Valid bit for Work Queue Entries (WQE) is a bit that indicates that a WQE is ready to be processed by the 10 GbE controller. The polarity of the valid bit changes each time the WQ wraps from the last entry back to the first entry. This change in polarity reduces software overhead associated with the need to clear a valid bit and also to enable the 10 GbE controller to read ahead in the WQ to reduce the need for doorbell rings. See Section 38.36.2.1.7 for more information on submitting work to a QP with the 10 GbE controller. Software is responsible to clear (set to 0b) all memory in a WQ initially at QP creation. The first iteration (and subsequent odd numbered iterations) through the WQ, software sets the *Valid* bit to 1b when it writes a new WQE. For the second iteration (and all even numbered iterations) through the WQ, software sets the valid bit to 0b when it writes an new WQE.

AdditionalFragmentCount (3)

AdditionalFragmentCount specifies the number of additional fragment descriptors that are valid for this WQE. Up to 6 additional fragment descriptors (listed in Table 38-498) can be added to a WQE for a maximum total of 7 fragments.

Push WQE (1)

Indicates that the WQE was pushed to the push page associated with the QP. If this bit is set and the 10 GbE controller dropped the push operation the *Push Dropped* bit is set in the next CQE following this WQE to indicate that software to stop pushing WQEs until an empty SQ condition has been observed. Using this rule prevents additional CPU and PCI bus bandwidth from being consumed when the Ethernet fabric is congested or the 10 GbE controller is unable to process push mode operations temporarily.

Signaled_Completion (1)

0b = Do not generate a CQE for this message unless there is an error associated with the WQE.

1b = Generate a CQE for this WQE, a CEQE might or might not be generated depending on the state of the CQ.

Local_Fence (1)

This bit if = 1b signifies that the current WQE MUST NOT start until all prior SQ WQEs on the QP completed.

Read_Fence (1)

This bit if = 1b signifies that the current WQE MUST NOT start until any outstanding RDMA read requests on the SQ complete.

Table 38-498.iWARP Additional Fragment Descriptor Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	Tagged Offset		
8	[63:32]	STag	[31:0]	Fragment Length

Tagged offset

The tagged offset (relative to the STag) associated with the data described by the additional fragment descriptor.

STag

STag associated with the data described by the additional fragment descriptor.

Fragment Length

Length in bytes of the data described by the additional fragment descriptor.

SQ WQE Format - NOP

Table 38-499.iWARP SQ NOP WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	RSVD		
8	[63:0]	RSVD		
16	[63:0]	RSVD		
24	[63] [62] [61] [60] [59] [58] [57]	WQE_Valid Signaled_Completion Local_Fence Read_Fence RSVD RSVD RSVD	[56] [55:48] [47] [46] [45:38] [37:32] [31:0]	Push WQE RSVD RSVD RSVD RSVD OP RSVD

OP (6)

This WQE format is valid for the NOP operation. See Section "Common iWARP SQ Descriptor Format Fields" for the associated opcode values.

WQE_Valid (1), Signaled_Completion (1), Local_Fence (1), Read_Fence(1), Push_WQE(1),

See Section "Common iWARP SQ Descriptor Format Fields" for more detail.

SQ WQE Format - Send

Table 38-500.iWARP SQ Send WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	Tagged_Offset		
8	[63:32]	STag	[31:0]	Fragment_Length
16	[63:0]	RSVD		
24	[63] [62] [61] [60] [59] [58] [57]	WQE_Valid Signaled_Completion Local_Fence Read_Fence wait_for_rcvFPDU Streaming_Mode Inline_Data_Flag	[56] [55:41] [40:38] [37:32] [31:0]	Push WQE RSVD AdditionalFragmentCount OP Remote_Invalidate_STag

OP (6)

This WQE format is valid for the send, send with invalidate, send with solicited event, and send with solicited event and invalidate, as well as the versions of these operations that include immediate data. See Section "Common iWARP SQ Descriptor Format Fields" for the associated opcode values.

WQE_Valid (1), Signaled_Completion (1), Local_Fence (1), Read_Fence(1), wait_for_rcvFPDU (1), Push_WQE(1), and AdditionalFragmentCount (3)

See Section "Common iWARP SQ Descriptor Format Fields".

Streaming mode (1)

This bit indicates that this WQE contains data that is to be sent in streaming mode (normal TOE, not iWARP); the only valid OP value with this bit is send, the total length of the message must be less than or equal to MSS or an AE (AE_WQE_LSMM_TOO_LONG) is generated and the QP state is transitioned to terminate.

wait_for_rcvFPDU (1)

Only valid if Streaming Mode = 1b. When this bit is set, the 10 GbE controller does not process the WQE following this one until a valid inbound iWARP ULPDU is received. This is useful for the last WQE of the IETF response frame.

Inline_Data_Flag (1)

Set to 0b for no inline data.

Remote_Invalidate_STag (32)

The STag on the remote peer that is to be invalidated by the message. This field is only valid if the operation type indicates an invalidate operations.

Tagged_Offset (64), STag (32), Length (32)

These fields describe the first fragment of the message to be sent. Additional fragments can be added to the WQE by setting AdditionalFragmentCount to a non-zero value.

SQ WQE Format - Send With Inline Data

Table 38-501.iWARP SQ Send With Inline Data WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	Data		
8	[63:0]	Data		
16				
24	[63]	WQE_Valid	[55]	RSVD
	[62]	Signaled Completion	[54:48]	Inline Data Length
	[61]	Local Fence	[47]	RSVD
	[60]	Read Fence	[46]	RSVD
	[59]	RSVD	[45:41]	RSVD
	[58]	RSVD	[40:38]	RSVD
	[57]	Inline_Data_Flag	[37:32]	OP
	[56]	Push WQE	[31:0]	Remote Invalidate STag

OP (6)

This WQE format is valid for the send, send with invalidate, send with solicited event, and send with solicited event and invalidate operations. See Section "Common iWARP SQ Descriptor Format Fields" for the associated opcode values.

WQE_Valid (1), Signaled_Completion (1), Local_Fence (1), Read_Fence(1), and Push_WQE(1)

See Section "Common iWARP SQ Descriptor Format Fields".

Inline_Data_Flag (1)

The Inline_Data_Flag must be set to 1b. This bit specifies that the data is contained inline with the WQE. If Inline_Data_Flag = 0b, the WQE is using descriptors and the WQE format is listed in Table 38-500. If Inline_Data_Flag = 1b the WQE has inline data and the WQE format is listed in Table 38-501.

Inline_Data_Length (7)

Inline_Data_Length indicates the number of bytes included in the WQE and all subsequent additional fragment descriptors.

Remote_Invalidate_STag (32)

See Section "SQ WQE Format - Send".

Data (up to 16 bytes)

The first 16 bytes of data are included in this field.

SQ WQE Format - RDMA Write

Table 38-502.iWARP SQ RDMA write WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	Tagged_Offset		
8	[63:32]	STag	[31:0]	Fragment_Length
16	[63:0]	Remote Tagged Offset		
24	[63] [62] [61] [60] [59] [58] [57]	WQE_Valid Signaled Completion Local Fence Read Fence RSVD RSVD RSVD	[56] [55:48] [47] [46] [45:41] [40:38] [37:32] [31:0]	Push WQE RSVD RSVD RSVD RSVD AdditionalFragmentCount OP Remote STag

OP (6)

See Section "Address Translation and Protection Overview" for the associated opcode value.

WQE_Valid (1), Signaled_Completion (1), Local_Fence (1), Read_Fence(1), Push_WQE(1), and AdditionalFragmentCount (3)

See Section "Address Translation and Protection Overview."

Tagged_Offset (64), STag (32), length (32)

These fields describe the first fragment (data source) of the message to be sent. Additional fragments can be added to the WQE by setting AdditionalFragmentCount to a non-zero value.

Remote Tagged_Offset (64), Remote_STag (32)

These fields describe the remote buffer (data sink).

SQ WQE Format - RDMA Write With Inline Data

Table 38-503.iWARP SQ RDMA Write With Inline Data WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	Data		
8	[63:0]	Data		
16	[63:0]	Remote Tagged Offset		
24	[63] [62] [61] [60] [59] [58] [57] [56]	WQE_Valid Signaled Completion Local Fence Read Fence RSVD RSVD Inline_Data_Flag Push WQE	[55] [54:48] [47] [46] [45:41] [40:38] [37:32] [31:0]	RSVD Inline Data Length RSVD RSVD RSVD RSVD OP Remote STag

OP (6)

See Section "Address Translation and Protection Overview" for the associated opcode value.

WQE_Valid (1), Signaled_Completion (1), Local_Fence (1), Read_Fence(1), and Push_WQE(1)

See Section "Address Translation and Protection Overview."

Inline_Data_Flag (1)

The Inline_Data_Flag must be set to 1b for inline data operations. This bit specifies that the data is contained inline with the WQE. If Inline_Data_Flag = 0b, the WQE is using descriptors and the WQE format is listed in Table 38-502. If Inline_Data_Flag =1b, the WQE has inline data and the WQE format is listed in Table 38-503.

Inline_Data_Length (7)

Inline_Data_Length indicates the number of bytes included in the WQE and all subsequent additional fragment descriptors.

Remote Tagged_Offset (64), Remote_STag (32)

These fields describe the remote buffer (data sink).

Data (up to 16 bytes)

The first 16 bytes of data are included in this field.

SQ WQE Format - RDMA Read

Table 38-504.iWARP RDMA Read WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	Tagged Offset		
8	[63:32]	STag	[31:0]	Fragment Length
16	[63:0]	Remote Tagged Offset		
24	[63] [62] [61] [60] [59] [58] [57] must be 0)	WQE_Valid Signaled Completion Local Fence Read Fence RSVD RSVD RSVD (Inline_Data_Flag,	[56] [55:48] [47] [46] [45:41 [40:38] [37:32] [31:0]	Push WQE RSVD RSVD RSVD RSVD RSVD OP Remote STag

OP (6)

This WQE format is valid for the RDMA read and RDMA read with local invalidate operations. See Section "Address Translation and Protection Overview" for the associated opcode value.

WQE_Valid (1), Signaled_Completion (1), Local_Fence (1), Read_Fence(1), and Push_WQE(1)

See Section "Address Translation and Protection Overview."

Tagged_Offset (64), STag (32), Length (32)

These fields describe the local target (data sink) of the RDMA read operation. STag also specifies the local STag to be invalidated for RDMA read with local invalidate operations.

Remote Tagged_Offset (64), Remote_STag (32)

These fields describe the remote buffer (data source).

SQ WQE Format - Memory Window Bind

Table 38-505.iWARP SQ Bind Memory Window WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	Memory Window Base VA		
8	[63:32]	Memory Window STag	[31:0]	Parent Memory Region STag
16	[63:46]	RSVD	[45:0]	Memory Window Length
24	[63] [62] [61] [60] [59] [58] [57] [56] [55-53]	WQE_Valid Signaled Completion Local Fence Read Fence RSVD RSVD RSVD Push WQE RSVD	[53] [52:48] [47] [46] [45:41] [40:38] [37:32] [31:0]	VA_Based_TO STag_Rights RSVD RSVD RSVD RSVD OP RSVD

OP (6)

See Section "Address Translation and Protection Overview" for the associated opcode value.

WQE_Valid (1), Signaled_Completion (1), Local_Fence (1), Read_Fence(1), and Push_WQE(1)

See Section "Address Translation and Protection Overview."

Memory window base virtual address (64)

This field specifies the starting point of the memory window within the parent memory region's virtual address range for the memory window.

Memory window length (46)

This field specifies the size of the memory window. Specifying 0b as a length results in an AE code of AE_AMP_MWBIND_INVALID_BOUNDS.

Memory window STag (32)

This field specifies STag of the memory window including the index and additional keys.

Parent memory region STag (32)

This field specifies STag of the parent memory region to which this memory window is bound including the index and additional keys.

VA_Based_TO (1)

VA_Based_TO specifies if the memory window is zero based or VA based. The memory window is VA based if VA_Based_TO is set (1b); else the memory window is zero based.

STag_Rights (5)

Indicates the rights assigned to this STag. The valid bits for this field are:

4 = Enable remote read.

8 = Enable remote write.

All other bits are unused.

SQ WQE Format - Local Invalidate

Table 38-506.iWARP Local Invalidate WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	RSVD		
8	[63:32]	STag	[31:0]	RSVD
16	[63:0]	RSVD		
24	[63] [62] [61] [60] [59] [58] [57]	WQE_Valid Signaled Completion Local Fence Read Fence RSVD RSVD RSVD	[56] [55:48] [47] [46] [45:41] [40:38] [37:32] [31:0]	Push WQE RSVD RSVD RSVD RSVD AdditionalFragmentCount OP RSVD

OP (6)

See Section "Address Translation and Protection Overview" for the associated opcode value.

WQE_Valid (1), Signaled_Completion (1), Local_Fence (1), Read_Fence(1), and Push_WQE(1)

See Section “Address Translation and Protection Overview.”

AdditionalFragmentCount (3)

Must be set to 0b for local invalidate operations.

STag (32)

STag specifies the STag to be invalidated.

SQ WQE Format - Fast Register

Fast register support for the 10 GbE controller has two modes of operation that depend on the number of available PBL resources. The prerequisite for issuing a fast register operation is to allocate an invalid memory region with a specific number of page list entries (or PBLEs for the 10 GbE controller). Since the 10 GbE controller enables host software to directly populate PBLs for memory registration, it is desirable to keep the same approach for fast register operations. The issue with allowing only the mode where host software directly populates the PBLs for fast register is apparent if an application issues fast register, send, local invalidate, fast register operations using the same memory region without waiting for completions for the local invalidate operation. If the 10 GbE controller only allowed the mode where software directly populated PBLs for fast register operations in these cases, memory corruption would be highly likely to occur. Software mechanisms to resolve the races end up with extremely inefficient fast register operations if they could be made to work at all. In order to not penalize every application for the previous behaviors but still provide proper handling of these behaviors, the following algorithm should be used for fast register operations:

1. During the allocate memory region operation, software must reserve the full number of PBL resources necessary to satisfy the allocate memory region request.
2. Software must keep an reference count per allocated memory region to track the number of outstanding fast register operations that are outstanding against the memory region that used the PBL resources allocated during the allocate memory region operation.
3. On fast register operation, if there are enough free PBL resources to satisfy the new request, software should populate a new area of PBLEs and issue the fast register operation without setting the Copy_Host_PBLs flag.
4. If there are not enough free PBL resources to satisfy the new request but the reference count of the users of the PBLs allocated during the allocate memory region operation is zero, software can use the PBL resources but must increment the reference count for the memory region.
5. If there are not enough free PBL resources to satisfy the new request but the reference count is one or more, host software must allocate a pinned buffer large enough to hold the page list specified on the fast register operation, populate that buffer with the page list specified by the fast register operation, and set the Copy_Host_PBLs bit and increment the reference count.
6. When the memory region associated with the fast register has been invalidated, the reference count for the memory region must be decremented.

This algorithm minimizes that cases where the page list associated with a fast register operation must be copied across the PCI bus while maintaining safe access to the memory region state and page list.

Table 38-507.iWARP SQ Fast Register WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0] Buffer Offset	Virtual Address or First		
8	[63:12]	PBL_Address	[11:0]	first_pm_pbl_index_high
16	[63:48] [47:46]	first_pm_pbl_index_low RSVD	[45:0]	STag_Length
24	[63] [62] [61] [60] [59] [58] [57] [56] [55:54]	WQE_Valid Signaled Completion Local Fence Read Fence RSVD RSVD RSVD Push WQE RSVD	[53] [52:48] [47] [46] [45:44] [43] [42:41] [40:38] [37:32] [31:08] [07:00]	VA_Based_TO STag_Rights RSVD Host_Page_Size Leaf_PBL_Size Copy_Host_PBLs RSVD AdditionalFragmentCount OP Driver_Key_STag_Index Consumer_Key

OP (6)

See Section "Address Translation and Protection Overview" for the associated opcode value.

WQE_Valid (1), Signaled_Completion (1), Local_Fence (1), Read_Fence(1), Push_WQE(1)

See Section "Address Translation and Protection Overview."

AdditionalFragmentCount (3)

Must be 0b for fast register operations.

VA_Based_TO (1)

VA_Based_TO specifies if the STag is zero based or VA based. Zero based STags carry only the first buffer offset in the virtual address or *First Buffer Offset* field. VA-based STags carry the full base VA including first buffer offset in the VA or *First Buffer Offset* field. The STag is VA based if VA_Based_TO is set (1b); else the STag is zero based.

Virtual Address or first buffer offset (64)

Indicates the base VA for this region/window for VA-based entries and indicates the first buffer offset for zero-based entries.

STag length (46)

Length of the memory region or memory windows specified by the STag index specified by Driver_Key_Stag_Index. Specifying 0 as a length results in an AE code of AE_AMP_FASTREG_INVALID_LENGTH. If Leaf_PBL_Size is set to 1b and Host_Page_Size = 0b then AE_AMP_FASTREG_INVALD_LENGTH is also generated if the size exceeds 2^{28} x 4096.

Access_Rights (5)

Indicates the rights assigned to this STag. The values for this field are 1 (enable local read), 2 (enable local write), 4 (enable remote read), 8 (enable remote write), and 16 (enable window bind).

Host_Page_Size (1)

Host_Page_Size specifies the page size of the backing pages for the STag. The values for this field are:

0b = 4 KB pages.

1b = 2 MB pages.

Driver_Key_STag_Index (24)

Index and *Driver Key* fields of the STag associated with the memory region. The 10 GbE controller supports a variable size STag index. This means that the number of bits used for *Driver Key* and for STag_Index are dependent on the maximum number of STags supported for a given PCI function. For example, if a PCI function read the FPMPEMRSZ field and found that the maximum number of MRTEs was 64 KB, the lower 16 bits of this field would be the STag index and the upper 8 bits would be a driver key that the driver can randomize to make guessing the MRTE layout more difficult to guess.

Consumer key (8)

Consumer key is the least significant 8-bit portion of the STag. This field is supplied by the user application or the driver.

Copy_Host_PBLs (1)

PBLs for an STag are located in host memory for the 10 GbE controller in the pages allocated for the Product Name HMC. In most situations, software copies backing pages for an STag directly to the HMC pages to optimize performance. If software needs the 10 GbE controller to populate the HMC pages, then Copy_Host_PBLs must be set to 1b and PBL_Address must point to the physical address of the backing pages in host memory.

Note: If the Copy_Host_PBLs bit is set, the number of PBLEs must be rounded up to an even number. That is, if the number of PBLEs is odd, software must allocate space for one more. The buffer must be a multiple of 16 bytes and it must be aligned on a 16-byte boundary.

Leaf_PBL_Size (2)

The 10 GbE controller supports physically contiguous STags and two forms of virtually contiguous STags. Physically contiguous STag do not require any PBLs and store physical address of the first page of the STag directly with the STag (no leaf PBL). Virtually contiguous STags that can be represented with a single HMC virtually contiguous address range require a single level PBL of Variable size. Virtually contiguous STags that are large (or in cases where the HMC address space for PBLs becomes fragmented) might require two level PBLs. In this case, the 10 GbE controller needs to know the length of the leaf PBLs in order to properly manage access to the PBLs. The valid settings for Leaf_PBL_Size are the following:

0 = No leaf PBL.

1 = Variable (one level).

2 = 256 bytes (two level).

3 = 4 KB (two level).

PBL Address (52)

Only valid if Copy_Host_PBLs is set (1b) or if Leaf_PBL_Size is set to 0b. If Copy_Host_PBLs is set and Leaf_PBL_Size is not set to 0b, PBL address is the physical address of the PBLs in host memory if they were not copied to HMC pages by software. If Leaf_PBL_Size is 256 or 4 KB, PBL address contains the physical address of a packed array of Root PBLEs. The format of the Root PBLEs is listed in Table 38-507. If Leaf_PBL_Size is variable, then PBL_Address contains the physical address of the page

list in host memory to be copied to first_pm_pbl_index. if Copy_Host_PBLs is clear and Leaf_PBL_Size is set to 0b, then the physical address of the physically contiguous memory is contained in this field.

first_pm_pbl_index (28)

This field defines the HMC PBLE object index used for the memory region page list. If Copy_Host_PBLs is set (1b), the 10 GbE controller copies the page list from the host address specified by PBL Address to the HMC object specified by first_pm_pbl_index. If Copy_Host_PBLs is clear (0b), first_pm_pbl_index designates the HMC base address for the PBLs for this STag that software has already initialized with PBL information.

38.36.4.5.2 SQ WQE Format - Connection Established

Table 38-508.iWARP SQ Established WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	RSVD		
8	[63:32]	RSVD		
16	[63:00]	RSVD		
24	[63] [62] [61:60] [59] [58] [57]	WQE_Valid RSVD (Signaled_Completion)	[56] [55:48] [47] [46] [45:41] [40:38] RSVD [37:32] [31:0]	RSVD RSVD RSVD RSVD OP RSVD

OP (6)

This WQE tells the chip to notify software when the connection is established. An AE (AE_LLP_CONNECTION_ESTABLISHED) will inform software that the connection is complete. This WQE is not valid forUDA QPs. See Section 38.38 for the associated opcode values.

WQE_Valid (1)

See Section 38.38.

38.36.4.5.3 RQ WQE Format

Table 38-509.iWARP RQ WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	Tagged Offset		
8	[63:32]	STag	[31:0]	Fragment Length
16	[63:00]			
24	[63] [62:41] [40:38]	WQE_Valid RSVD AdditionalFragmentCount	[37:32] [31:0]	RSVD (OP) RSVD

WQE_Valid (1) and AdditionalFragmentCount (3)

See Section "Address Translation and Protection Overview."

Tagged_Offset (64), STag (32), Length (32)

These fields describe the first fragment of the byte stream to be sent. Additional fragments can be added to the WQE by setting AdditionalFragmentCount to a non-zero value.

38.36.5 UDA Functionality

The UDA feature expands a user-space networking communication to a wide scope of Ethernet-based protocols beyond iWARP.

The UDA host interface and semantics are very similar to one exposed by iWARP.

- Sending and receiving data
- Application uses UDA QPs and CQs to send and receive data, see Section 38.36.3.3.2 and Section 38.36.3.3.3 for UDA QP and CQ management requests. UDA QPs and CQs resources are shared with iWARP.
- Completion and AE management
- CE notification mechanism is identical to one used for iWARP, see Section 38.36.3.4.2 and Section 38.36.3.5 for CEQ and AEQ allocation description. UDA probably shares event and completion notification structures with iWARP.
- Application buffer memory management
- An application should register its buffers (see Section 38.36.3.3.4) to enable direct placement to and from application buffers. Memory registration used for UDA buffers is identical to the memory registration of iWARP, and uses resources from the same pool of memory regions available for PCI function.
- UDA enables send and receive Ethernet frames by posting frame payload and ULP protocol headers to UDA QP using post send and post receive operations described in Section 38.36.5.8.1 and Section 38.36.5.8.2. Separate SQ/RQ WQE is used to send and receive single Ethernet frame.

UDA provides a protocol agnostic interface with limited stateless accelerations for the well known protocols. To enable secure userspace networking for non-privileged consumers UDA limits scope of supported protocols to IP-based protocol only, by offloading Ethernet and IP header generation to hardware. For the privileged consumer in the trusted environment it enables application fully control of header generation without limiting protocols to be IP-based or having any headers generated by hardware.

Hardware provides partial stateless acceleration for the limited scope of protocols, discussed in Section 38.36.5.1 and Section 38.36.5.2 later.

UDA acceleration enables a wide scope of applications to take advantage of direct communication with hardware bypassing system overhead. UDA brings a variety of benefits, including deterministic and low latency, low-latency deviation and jitter, high-message processing rate with a very low network communication overhead, lock free networking and linear scalability with number of cores in the system without compromising system security.

38.36.5.1 Transmit User-space Direct Access Hardware Acceleration

Transmit UDA hardware provides a limited number of partial stateless accelerations.

UDA traffic is not expected to use a dedicated internal switching resources (VSIs). It shares an internal switch virtual port with iWARP traffic. Depending on the assignment of the user priority, UDA traffic might share a QS with iWARP QPs, or use a dedicated QS. Assignment of UDA traffic to the QS is done at QP creation time, and is similar to assignment of iWARP QP.

The 10 GbE controller generates Ethernet and IP headers based on information provided in address vector for IP-based user-space protocols (see Section 38.36.3.3.2). Offloading Ethernet and IP headers generation does not bring much performance improvement, but does improve security of UDA for IP-based protocols. In the standard deployment configuration, UDA supports only IP-based protocols, with an option available for the privileged software running in trusted environment allowing to generate all headers by software. Privileged mode can be enabled by setting *Privileged Header Generation Enable* bit in the UDA QP context, see Section 38.37.5.4.

The 10 GbE controller supports header generation for non-fragmented datagrams. Hardware is responsible for the *IP Identification* field. Hardware maintains one instance of the *IP Identification* field per UDA QP.

Software is responsible for generating upper protocol headers (TCP, etc.). Those headers can be provided as a part of the payload WQE fragment or as a separate WQE fragment. In privileged mode, software can provide headers in the same WQE fragment with payload or using one or more dedicated WQE fragments. The 10 GbE controller also supports an option of inline data, when both upper layer protocol headers and a frame payload are copied directly to the WQE. See Section 38.36.5.8.2 for more details. push mode, described in Section , can be used to transmit UDA WQEs with inline data. See Section 38.36.5.8.3 for more details.

The 10 GbE controller offloads checksum generation for non-fragmented TCP packets.

UDA traffic uses standard internal switching capabilities provided by the 10 GbE controller to internally switch unicast and multicast traffic between different VSIs. Unlike traffic generated by standard operating system stack, UDA QPs also require additional internal switching capabilities within VSI, similar to iWARP. This enables internal switching between user-space processes directly communicating with hardware using UDA QPs.

UDA and the host operating system networking stack independently generates the *IP Identification* field. To avoid collision of the IP identification space, the 10 GbE controller has a configurable option to override I*P Identification* fields generated by the host networking stack and UDA and forces the MSB of the IP identification to different values. The *IP Identification* field override can be enabled per VSI using the USEENTIREIDRANGE bit in the IPCONFIG register.

38.36.5.2 Receive User-space Direct Access Hardware Filtering and Acceleration

Receive UDA acceleration is focusing on identifying UDA frames and delivering those frames to the associated UDA RQ. This section describes how UDA frames can be identified using various Product Name filters, lists the 10 GbE controller filters that can be programmed for UDA traffic identification and refers to the sections describing programming of each filter in details. UDA Product Name filtering descriptions are organized by the traffic types.

38.36.5.2.1 TCP UDA Filtering and Acceleration

TCP UDA acceleration enables taking advantage of UDA QPs to send and receive unfragmented TCP segments directly from the application address space or from the kernel. iWARP connection management and exception handling is the usage model for this capability.

On the transmit side, TCP UDA acceleration can take advantage of the generic UDA acceleration capabilities described in Section 38.36.5.1.

This section describes receive filtering capabilities and hardware accelerations that can be used to accelerate TCP segments.

UDA TCP acceleration is enabled as soon as iWARP capabilities are enabled for the 10 GbE controller. Similar to iWARP, PCI function for the inbound TCP segments are identified using destination MAC address.

UDA TCP acceleration does not support fragmented IP datagrams. All fragments are processed by stateless filters and delivered to the standard host stack.

UDA TCP acceleration uses an accelerated port table to filter out accelerated traffic based on the destination TCP port. Software should allocate a port to be used for UDA TCP accelerated traffic using the CQP operation described in Section 38.36.3.3.2.

TCP packets that hits an accelerated port table are filtered through the quad hash table.

The quad hash table is a perfect filter that carries an entry for each accelerated TCP connection that is offloaded to the 10 GbE controller's PE. the quad hash table has two kinds of entries used for TCP traffic.

- Quad-touple entries carrying and using as input to hash function a quad of source and destination IP addresses and source and destination TCP ports.
- Du-touple entries carrying and using as an input to the hash functions a pair of destination IP address and destination TCP port.

The du-entries are allocated to filter out connection establishment packets that are targeting accelerated connections. Those entries are used to filter out TCP packets with a *SYN* bit set and *ACK* bit clear.

The quad-touple entries are allocated to filter out packets targeting already established accelerated QPs.

The passive side connection establishment flow:

- Listen.
  - — Allocate du-touple entry with local IP address and local port socket is bound to.
- SYN is received on QP associated with du-touple.
  - — Allocate quad-touple entry with local/remote IP addresses and local/remote TCP ports.
  - — Send SYN-ACK
- ACK is received on the QP associated with quad-touple.

The active side connection establishment flow:

- Connect.
  - — Allocate quad-touple entry with local/remote IP addresses and local/remote TCP ports
  - — Allocate quad-touple entry with local/remote IP addresses and local/remote TCP ports

  - Send SYN
- SYN-ACK is received on QP associated with quad-touple.
  - Send ACK.

UDA TCP uses the quad hash table in the same way this table is used for iWARP traffic. A hit in the table resolves a number of destination QPs. The 10 GbE controller uses information in QP context to identify iWARP and UDA TCP traffic. If a packet misses the quad hash table, it is forwarded for the further processing to the stateless filters.

TCP UDA software should program the quad hash table with du-touple entry to establish UDA accelerated TCP connection (such as listen or connect socket calls) and to add a quad-touple entry once connection has been established. To avoid race and loss of data, the quad-touble entry can be allocated before the connection establishment handshake completed.

The 10 GbE controller does not terminate TCP connections for UDA traffic. It forwards identified UDA TCP segments to the corresponding RQ. Multiple TCP tuples can be configured to be delivered to the same RQ. Each WQE in UDA TCP RQ is consumed by single TCP segment.

In addition to forwarding UDA TCP segments, the 10 GbE controller provides several hardware accelerations.

- Header and data separation. The 10 GbE controller enables an option to configure UDA TCP QP to separate Ethernet/IP and TCP protocol headers from the payload, and deliver it to the separate WQE fragments or to the private header ring buffer. See Section 38.36.5.8.1 and Section 38.37.5.4 for more details.
- TCP checksum calculation and validation. The 10 GbE controller supports calculation and validation of TCP checksum for UDA TCP segments. TCP packets with invalid a checksum is dropped and not delivered to RQ.

38.36.5.3 User-space Direct Access Programming Interface

The UDA programming interface is based on the verb semantics described in Section 38.36.2.1. It uses same basic constructs as iWARP: QP, CQ, CEQ, AEQ and memory regions. Those constructs are described in detail in Section 38.36.2.1. In addition to those constructs, UDA introduces address vectors and multicast groups described later in this section.

38.36.5.3.1 CEQ

CEQ construct enables hardware to provide software with asynchronous completion notifications. Application has a full control over requesting asynchronous completion event for the particular CQ. UDA uses CEQ construct defined for iWARP described in Section 38.36.2.1.2.

CEQs are allocated per MSI-X vector per PCI function and if UDA and iWARP applications are deployed by the same PCI functions or the same application uses both types of traffic, it shares the same CEQ. Each entry in CEQ indicates that a requested CE was received by specified CQ and the application should poll CQ and retrieve completion of transmit or receive operation.

38.36.5.3.2 AEQ

AEQ construct enables hardware to report AE and error notifications. UDA uses AEQ defined for iWARP described in Section 38.36.2.1.1. AEQ is allocated per PCI function and shared by iWARP and UDA if application(s) using both types of traffic are deployed in the same PCI function.

38.36.5.3.3 CQ

CQ is a construct that enables an application to receive notification about completed transmit and receive operations directly in the application address space. UDA uses CQ construct defined for iWARP and described in Section 38.36.2.1.3. CQ is associated with RQ and SQ to report completion of transmit and receive operations. The same CQ can be configured to report completion of transmit and receive operations on the same and different QPs. Application is allowed to use same CQ to report completion of UDA and iWARP traffic.

CQ entry indicates what operation has been completed, and depending on the completed operation, it carries additional information. The format of a UDA CQ entry is described in Section 38.36.5.5. Among other fields, CQ entry carries a 64-bit pointer to the QP completion information. This field can be used to identify a QP that posted completed operation.

Completion of receive operation is always reported in associated CQ. Completion of transmit operation is not necessarily reported and the application can control whether it wants to be notified about a completed transmit operation. Application MUST request completion of transmit operation at least once per SQ size worth of transmit requests.

38.36.5.3.4 QP

QP is a construct that enables an application to post transmit and receive operations directly from the application address space. QP consists of the pair of queues: SQ and RQ. UDA uses QP construct defined for iWARP described in Section 38.36.2.1.5.

Unlike iWARP QP, UDA QP does not have to be associated with any particular connection or connection oriented service and enables mixing of various ULPs on the same QP.

User-space QP is associated with a particular local IP address. Non-privileged UDA consumers need to allocate multiple UDA QPs to send UDA packets using different local IP addresses. Local IP address in QP context is used for UDA IP header generation only. The 10 GbE controller filters can be configured to receive packets targeting different local IP addresses on the same QP.

Software should use WQEs described in Section 38.36.5.6 to post new transmit and receive work to the UDA QP.

UDA QP is associated with a particular TC, and VLAN. If the application needs to use multiple TCs and VLANs, it must create multiple QPs and spread traffic, respectively.

All software errors such as invalid WQE format, invalid STag and memory region boundary violation are considered to be critical errors and result in a transition of QP to the error state and immediately suspend transmit and receive operations of that QP. Respective AE would be reported via AEQ associated with QP. See Section 38.36.2.6.

38.36.5.3.5 Send Operation

The structure of SQ is a process of posting WRs to UDA SQ and flow is similar to one described for iWARP SQs in Section 38.36.2.1.7.

UDA QP enables the application to transmit individual Ethernet frames per posted SQ WQE. Each transmit WQE must refer to a single Ethernet frame. Software is responsible to limit the frame size to the configured MSS. If frame size exceeds MSS, frame would be discarded and error reported in CQE.

Software is allowed to post Ethernet frames shorter than a minimum Ethernet frame length. Hardware pads Ethernet frames to the minimal Ethernet frame length.

Hardware is responsible for Ethernet FCS and IP checksum generation. Software might request hardware to generate TCP checksum for non-fragmented datagrams.

In default operation mode, Ethernet and IP headers are generated by hardware using information provided in the address vector. Software can provide additional ULP header either as a part of the application buffer referred by SQ WQE fragments or using a private header ring buffer (if enabled for the QP). If size of the extended header exceeds the size of the private header ring buffer entry, the packet is discarded and the error is reported in CQE.

Privileged consumers are allowed to post Ethernet frames with all headers. To enable privileged SQ operation mode, the *Privileged Header Generation* bit should be set at QP creation (see Section 38.37.5.4).

Software can selectively request completion of WQEs by setting the *Signaled* bit. Completions with errors are returned regardless of the *Signaled* bit setting. UDA WQE is completed as soon as hardware finished processing WQE and validated lengths and fragments. Completion of UDA SQ WQE does not indicate that respective Ethernet frames are transmitted or received by the destination.

38.36.5.3.6 Receive Operation

The structure of RQ is a process of posting WRs to UDA RQ and flow is similar to one described for iWARP RQs in sub-section Section , "RQ".

UDA QPs can be configured to use SRQ. The structure of SRQ is a process of posting WRs to UDA SRQ and flow is similar to one described for iWARP SRQs in sub-section Section , "RQ".

UDA QP enables an application to receive an individual Ethernet frame in pre-posted RQ WQEs. Each inbound Ethernet frame consumes one RQ WQE. Software needs to make sure that buffers posted to RQ WQE are large enough. If size of the Ethernet frame exceeds the size of RQ WQE, the frame is truncated and an error is reported in CQE.

Ethernet frames failing CRC or checksum checks (IP/TCP) are dropped and not delivered to UDA QPs.

If an Ethernet frame is received while UDA RQ is empty, such frame is dropped and the GLPES_PFIP4RXDISCARD statistics counter is incremented.

Software can enable header/data split feature per RQ. If header/data split is enabled, then hardware splits the frame header and payload and places those separately to the designated locations. Payload of the received Ethernet frame is always placed in the buffers referred to by RQ WQE fragments. Depending on the RQ configuration, a frame header can placed either to the first fragment of the RQ WQE and then the frame payload is placed in RQ WQE starting with second fragment. Or, the frame header can be placed in the private header ring buffer and then the frame payload is placed in the RQ WQE starting with first fragment.

If SRQ and header/data split are enabled, the header can be placed to the first fragment only. The dedicated header ring buffer can be enabled for RQ only.

If RQ is configured with a header/data split option and the size of the frame header exceeds the size of the first fragment of the size of the private ring buffer entry, the frame is dropped and an error is reported in CQE.

RQ completion is delivered with each completed RQ WQE. If a packet carries one or more VLAN tags, the most inner VLAN tag is reported by RQ CQE. Note that VLAN tag is not stripped from the packet header.

38.36.5.3.7 Memory Registration

Memory region is a construct that enables hardware to access data directly from the application buffers. To enable direct hardware access, an application must register application buffers using the register memory region operation. UDA uses memory regions defined for iWARP described in Section 38.36.2.1.4.

Memory region construct is not associated with any particular traffic type and registered memory region can be accessed both by iWARP and UDA traffic. This enables an application to have the same application buffer be directly accessed by iWARP and UDA traffic. For example, data received via one of UDA QPs can be then accessed via iWARP QP without requiring copy operation.

UDA application must register application buffers to be used to send and receive UDA packets.

UDA supports only basic memory registration capabilities and does not support fast memory registration, bind and invalidate operations.

38.36.5.3.8 Push Mode Support

Push mode enables software to reduce processing latency of the short messages by pushing corresponding WQE with inline data to the memory mapped adapter address space. Pushed WQE with inline data can be immediately processed by the adapter and transmitted to eliminate the need to read WQE and data from the host memory. See Section "SQ Push Mode" for a detailed description of push mode concept and constructs.

UDA QPs take advantage of push mode defined for iWARP. Software can use push mode WQEs, described in Section 38.36.5.8.3 to reduce processing latency of the short UDA messages. Similar to iWARP, the size of the UDA push mode message is limited to 96 bytes. The UDA push mode message cannot use private ULP header buffers and the entire message with ULP headers must fit the push mode message size restrictions and posted as inline data. The remainder of push mode message software processing flow is identical to one described for iWARP messages in sub-section Section "SQ Push Mode".

38.36.5.4 User-Space Direct Access QP Context Format

Users-pace direct access QP context is used by software to initialize or update the 10 GbE controller's QP context. Create and Modify QP CQP operations (see Section 38.36.3.3.2) reference this structure.

User-space QP is associated with a particular local IP address. Non-privileged UDA consumers need to allocate multiple UDA QPs to send UDA packets using different local IP addresses. The local IP address in QP context is used for UDA IP header generation only. The 10 GbE controller filters can be configured to receive packets targeting different local IP Addresses on the same QP.

Table 38-510.UDA QP Context Structure Format (Sheet 1 of 2)

Byte Offset	[Bit Range]	Field Name	[Bit Range]	Field Name
0	[63:62] [61:48] [47] [46:42] [41:32] [31] [30] [29] [28] [27] [26] [25:20] [19] [18:16] [15] RSVD(drop_out_of_order_seg) [14]	RSVD RSVD Push Mode Enable RSVD Push Page Index SQ_TPH_en RQ_TPH_en XMIT_TPH_en RCV_TPH_en RSVD RSVD RSVD RSVD RSVD(dupack_thresh) RSVD	[13:12] [11] [10] [9:8] [7] [6] [5] [4] [3] [2] [1:0]	RSVD(limit) RSVD RSVD RQ_WQE_Size RSVD(timestamp) RSVD(Insert VLAN Tag) RSVD(NoNagle[1]) RSVD RSVD(iwarp_ddp_ver[2])
8	[63:0]	SQ_Address		
16	[63:0]	RQ_Address		
24	[63:48] [47:32] [31:24] [23] [22] [21:16]	RSVD(Dest Port Number) RSVD(Source Port Number) RSVD(Traffic Class or TOS RSVD(avoid_stretch_ack) RSVD	[15:12] [11:8] [7:0]	SQ Size RQ Size RSVD(Hop Limit or TTL)
32	[63:32]	RSVD(Dest_IP_Address_1)	[31:0]	RSVD(Dest_IP_Address_0)
40	[63:32]	RSVD(Dest_IP_Address_3)	[31:0]	RSVD(Dest_IP_Address_2)
48	[63:48] [47:32] [31:30]	RSVD(ARP Index) RSVD(VLAN Tag) RSVD	[29:16] [15:14] [13:0]	snd_mss RSVD rsvd (rcv_mss)
56	[63] [62:48] [47:44] [43:40] [39:36] [35:32]	RSVD pd_index RSVD RSVD(Snd_wscale) RSVD RSVD(Rcv_wscale)	[31:28] [27:24] [23] RSVD(ignore_tcp_uns_options) [22] [21] [20] [19:00]	RSVD(TCP_state) RSVD RSVD(ignore_tcp_options) RSVD RSVD(wscale) RSVD(Flow Label)
64	[63:50] [49:48] [47:34] [33:32]	RSVD RSVD	[31:6] [5] [4] [1] [0]	RSVD RSVD RSVD RSVD PrivilegedHdrGenEnable
72	[63:00]			
80	[63:00]			
88-128	[63:00]	RSVD		
136	[63:49] [48:32]	RSVD RxCmpQueueNum	[31:17] [16:0]	RSVD TxCmpQueueNum
144	[63:0]	RSVD(Q2_Address)		
152	[63:47] [46:32]	RSVD	[31:16] [15:8] [7:0]	RSVD RSVD RSVD(last_byte_sent)

Table 38-510.UDA QP Context Structure Format (Sheet 2 of 2)

Byte Offset	[Bit Range]Field Name			
160	[63:57] [56:48] [47:41] [40:32] [31] [30] (assume_aligned_headers) [29] [28] [27] [26]	RSVD RSVD(snd_mrk_offset) RSVD RSVD(rcv_mrk_offset) RSVD(rcv_no_mpa_crc) RSVD RSVD(Receive Markers) RSVD(iWARP Mode) RSVD RSVD	[25] [24] [23] [22] [21] [20] [19:18] [17:16] [15:7] [6:0]	PrivilegedEnable RSVD(FastRegisterEnable RSVD(BindEnable) RSVD(Send Markers-) RSVD(rdmard_ok) RSVD(rdmawr_rdresp_ok) RSVD RSVD(IRD_Size) RSVD RSVD(ORD_Size)
168	[63:0]	QP Completion Context		
176	[63:44] [43:32] (Exception_UDA_Queue) [31:26]	RSVD RSVD RSVD	[25:16] [15:8] [7:0]	QS_Handle RQ_TPH_value SQ_TPH_value
184	[63:32]		[31:0]	
192	[63:32]		[31:0]	
198-248	[63:0]	RSVD		

Notes:

1. This variable is a cached variable. See Cached_Variables_Valid in Modify QP operations for more details on cached context variables.
2. Only Valid for iWARP QPs.

This bit is used to enable privilege mode on STag's. If this bit is set, then a local STag of zero (STag 0) is allowed on this connection and the *TO* field should be treated as a physical address. If this bit is cleared, then STag 0 is NOT allowed for a local STag and an AEQE is generated with a privilege error indicated.

PrivilegedHdrGenEnable (1)

This bit is used to enable privilege mode on SQ and enabling software to generate all Ethernet frame headers. When set, hardware assumes that software provides all headers for the Ethernet frame to transmit and does not generate Ethernet/IP headers. Standard anti-spoofing still apply, based on VSI configuration. This mode usually can be enabled for consumers operating in privileged mode (root).

1b = Application is allowed to generate all header, including Ethernet and IP, and hardware does not apply any security checks beyond standard anti-spoofing checks done by internal switch.

0b = Hardware generates Ethernet and IP headers for all packets using information provided in address handle. Upper layer protocol headers can be supplied by application or verb library.

Default mode has this bit clear (0b) to ensure secure Ethernet and IP header generation by hardware. Insecure header generation is allowed for privileged applications operating in trusted environments.

XMIT_TPH_en (1)

If set (1b), TPH is enabled for data reads associated with this QP. If clear (0b), THP is not used for data reads associated with this QP.

SQ_TPH_en (1)

If set (1b), TPH is enabled for the SQ of this QP. If clear (0b), THP is not used for this resource.

SQ_TPH_value (8)

If *SQ_TPH_en* is set (1b), TPH STag associated with SQ operations is initialized with *SQ_TPH_value*. If *SQ_TPH_en* is clear (0b), this field is ignored.

SQ_HDR_TPH_en (1)

If set (1b), TPH is enabled for ULP header read associated with this QP. If clear (0b), THP is not used for ULP header read associated with this QP. Takes effect only if *SQ_Hdr_Ring_Buffer_En* is set (1b).

SQ_Size (4)

This field encodes the maximum size for the WQ. The encoding of the SQ sizes are $4*2^{SQ\_Size}$ in terms of 32-byte quanta of memory. The following values are allowed for *SQ_Size*:

0 = 128 bytes
1 = 256 bytes
2 = 512 bytes
3 = 1024 bytes
4 = 2048 bytes
5 = 4096 bytes
6 = 8192 bytes
7 = 16384 bytes
8 = 32768 bytes
9 = 65536 bytes
10 = 131072 bytes
11 = 262144 bytes
12 = 524288 bytes

Software can only allocate N-1 WQEs on the SQ, where N is a SQ size previously defined. Each WQE is variable in size and can consume up to 128 bytes of memory. For more information see Section 38.36.2.1.7. The minimum size for a SQ is four WQEs of the maximum size that is used for the SQ.

RCV_TPH_en (1)

If set (1b), TPH is enabled for data placement associated with this QP. If clear (0b), THP is not used for data placement associated with this QP.

RQ_TPH_en (1)

If set (1b), TPH is enabled for the RQ of this QP. If clear (0b), THP is not used for this resource.

RQ_TPH_value (8)

If RQ_TPH_en is set (1b), TPH STag associated with RQ operations is initialized with *RQ_TPH_value*. If RQ_TPH_en is clear (0b), this field is ignored.

RQ_HDR_TPH_en (1)

If set (1b), TPH is enabled for packet header placement associated with this QP. If clear (0b), THP is not used for packet header placement associated with this QP. Takes effect only if *RQ_Hdr_Split_En* is set (1b).

RQ_Size (4)

This field encodes the maximum size for the WQ. The encoding of the RQ sizes are $4*2^{RQ\_Size}$ in terms of 32-byte quanta of memory. The following values are allowed for *RQ_Size*:

0 = 128 bytes
1 = 256 bytes
2 = 512 bytes
3 = 1024 bytes
4 = 2048 bytes
5 = 4096 bytes
6 = 8192 bytes
7 = 16384 bytes
8 = 32768 bytes
9 = 65536 bytes
10 = 131072 bytes
11 = 262144 bytes
12 = 524288 bytes

The actual number or WQEs that can be posted to the RQ is the size of the WQ divided by the WQE size determined from RQ_WQE_Size. Software can only submit N-1 WQEs to a WQ without processing completions for the WQ without exposing the possibility of a WQ overflow. Where N is an RQ size previously defined. WQ overflow results in indeterministic behavior for the affected WQ. The minimum size for an RQ is four WQEs.

RQ_WQE_Size (2)

Specifies the number of 32 byte chunks of memory included with each RQ WQE. The maximum number of additional fragments allowed for an RQ WQE is 6 for a total of 7 fragments and a maximum WQE size of 128 bytes. Valid values are:

0 = 32 bytes per WQE (no additional fragments)
1 = 64 bytes per WQE (1 or 2 additional fragments)
2 = 128 bytes per WQE (3 to 6 additional fragments)
3 = Reserved

RxCmpQueueNum (17)

This field specifies which of the 128 K CQs is used for receive completion notification. The Rx and Tx completions can be mapped to the same CQ or different queues.

TxCmpQueueNum (17)

This field specifies which of the 128 K CQs is used for transmit completion notification. The Rx and Tx completions can be mapped to the same CQ or different queues.

SQ Address (57)

If the *VirtualWQ* bit is clear, then this field holds the SQ base physical address. It must be aligned to an address divisible by 128 bytes. If the *VirtualWQ* bit is set, then this field specifies the first HMC PBLE index of the 1 level page list for the SQ (first first_pm_pbl_index).

RQ Address (57)

If the *VirtualWQ* bit is clear, then this field holds the RQ base physical address. It must be aligned to an address divisible by 128 bytes. This is the RQ base pointer when this is an iWARP accelerated connection. If the *VirtualWQ* bit is set, then this field specifies the first HMC PBLE index of the 1 level page list for the SQ (first first_pm_pbl_index).

QS_Handle (10)

This field specifies the Tx scheduler queue set handle associated with the TC for this QP. For VFs, the *QS_Handle* is checked to ensure that the VF issuing the CQP command is associated with the *QS_Handle*.

pd_index (15)

Protection domain for this context. This specifies which one of the 32 KB UDA protection domains this connection context belongs too. There are no reserved index values.

QP_Completion Context (64)

This field is reported in CQEs and also in AEQEs.

Push Page Index (10)

This field identifies the push page associated with the QP. For VFs, the push page index is function relative.

Push Mode Enable (1)

This field indicates if push mode is enabled for the QP.

snd_mss (14)

This field specifies a maximum MSS size allowed on that QP. MSS includes all headers and payload of Ethernet frame, both generated by hardware and provided by software. If software posts a SQ WQE that exceeds this value, the frame is dropped, AE_UDA_XMIT_DGRAM_TOO_LONG AE is reported and QP transitions to the error state.

Use_Statistics_Instance (1)

This field indicates if the default per PMF statistics are used or if one of the additional RDMA statistics instances are used for this QP. When this field is zero, the default statistics are used. When this field is one, the statistics instance indicated by the Statistics_Instance_Index field are used.

Statistics_Instance_Index (5)

This field specifies which of the additional RDMA statistics indexes are used if Use_Statistics_Instance is one. This field is ignored if Use_Statistics_Instance is zero.

38.36.5.5 User-Space Direct Access CQ Entry Formats

Table 38-511.UDA CQ Entry Format

Byte Offset	[Bit Range]Field Name			
0	[63:48] [47:32]	VLAN Tag RSVD(Partial_Checksum)	[31:25] [24:16] [15:14] [13:0]	RSVD Header_Length RSVD Payload Length
8	[63:0]	QP Completion Context		
16	[63:50] [49:32]	RSVD QP_ID	[31:18] [17:16] [15:2] [1:0]	RSVD L4Protocol RSVD L3Protocol
24	[63] [62] [61:56] [55] [54] [53]	CQE_Valid SQ RSVD (OP) Error RSVD (Solicited Event) RSVD (STag)	[52] [51] [50] [49:46] [45:32] [31:16] [15:0]	 Push Dropped(UDA SQ only) VLAN Tag Valid RSVD WQ_Desc_Index Major Error Code Minor Error Code

CQE_Valid (1)

The *CQE_Valid* bit for CQE is a bit that indicates that a CQE is ready to be processed. The polarity of the valid bit changes each time the (CQ wraps from the last entry back to the first entry). This change in polarity reduces software overhead by avoiding the need to clear the valid bit once software has processed a valid CQE. Software is responsible to clear (set to 0b) all memory in a CQ initially at CQ creation. The first iteration (and subsequent odd numbered iterations) through the CQ, the 10 GbE controller sets the *Valid* bit to 1b when it writes a new CQE. For the second iteration (and all even numbered iterations) through the CQ, the 10 GbE controller sets the *Valid* bit to a 0b when it writes an new CQE.

Error (1)

0b = No error.

1b = An error occurred when processing the WQE associated with this CQE and that the Error Code field is valid.

SQ (1)

0b = RQ.

1b = SQ.

Push Dropped (1)

Valid only when SQ is set.

0b = Push operations are being processed successfully.

1b = A recent push mode operation has been dropped by the 10 GbE controller, software should refrain from submitting additional push mode operations until the SQ has encountered an empty condition.

WQ_Desc_Index (14)

WQ are split up into 32-byte descriptor quanta. Every WQE must start with a 32-byte descriptor on a 32-byte boundary. WQ_Desc_Index reports the 32-byte quanta index of the WQE associated with the completion.

Payload_Length (14)

Total payload length of the completed packet (up to a jumbo frame). This field is only valid for RQ WQE completions.

Header_Length (9)

Length of the packet header placed to the private header ring buffer. if zero, then the header was placed to the payload buffer per *Header_Split_En* bit in QP context. This field is only valid for RQ WQE completion.

QP_ID (18)

QP associated with the completed message.

Major Error_Code (16)

Valid if Error is set (1b). See Table 38-494 for defined values. Software can also report any value via the Flush WQEs CQP operation (see Section 38.36.3.5.4).

Minor Error_Code (16)

Valid if Error is set (1b). See Table 38-494 for defined values. Software can also report any value via the Flush WQEs CQP operation (see Section 38.36.3.5.4).

L3Protocol (2)

This field carries an indication of a L3 type of the received packet.

- (00b) = IPv4 packet
- (01b) = IPv6 packet
- (1xb) = other type of packet.

This field is only valid for RQ WQE completions (when SQ is clear).

L4Protocol (2)

This field is valid for IPv4/IPv6 packets only. Indicates the L4 protocol of the packet.

- (00b) = TCP
- (01b) = RSVD
- (10b) = RSVD
- (11b) = other.

This field is only valid for RQ WQE completions (when SQ is clear).

VLAN Tag (16)

A most inner VLAN Tag detected in the packet. Note that VLAN Tag is not stripped from the packet. This field is only valid for RQ WQE completions (when SQ is clear).

VLAN Tag Valid (1)

Indication that CQE carries a valid most inner VLAN Tag detected in the packet. This field is always 0 when SQ is set.

QP Completion Context (64)

Completion context pointer. This field is transferred to the CQE from QP Context.

38.36.5.6 UDA QP Completion Error Codes

UDA QP completion errors are typically reported via AEs with the 10 GbE controller. After fielding the AEs, software might issue the Flush WQES CQP operation (see Section 38.36.3.5.4) to complete any pending operations. Table 38-512 lists the

completion codes reported by the 10 GbE controller in a CQ entry (Table 38-511). Received packets that failed checksum validation are dropped by the chip and not delivered to RQ.

Table 38-512.UDA QP Error Codes

Major Error Code	Minor Error Code	Completion Reason	Description
0x0001	0x0001	WQE Flushed	The WQE has been flushed due to a ModifyQP state transition.
0x0002	0x0001	RQ WQE Too Short	The RQ WQE length was shorter than a length of received Ethernet frame. Frame was truncated, and error reported.
0x0002	0x0002	RQ WQE Hdr Too Short	The Header of RQ WQE was shorter than a header of received Ethernet frame. This error can be reported only if header/data split option has been enabled for RQ. Depending on the RQ configuration, Ethernet frame header exceeded either the size of the private receive header buffer, or a size of the first fragment.
0x0002	0x0003	SQ WQE Too Long	The SQ WQE length exceeded an MSS configured for the QP. SQ WQE length should include size of the payload, hardware generated Ethernet and IP headers, and extended header provided by software.
0x0002	0x0004	SQ WQE Hdr Too Long	The SQ WQE extended header size exceeded size of the private header buffer entry.

38.36.5.7 UDA QP Asynchronous Error Codes

UDA uses asynchronous errors to report unrecoverable critical errors. Those errors mostly used to identify invalid use of UDA host interface. UDA uses asynchronous error notification mechanism described in Section 38.36.5.3.2. UDA shares asynchronous error codes with iWARP, see Section 38.36.2.6.

As a rule, once unrecoverable critical error is detected on QP. QP is transitions to the error state, and suspends all transmit and receive operations. Any incoming packet targeting a QP in an error state is silently discarded. To release buffers posted to a SQ and a RQ, software has to use the Flush WQE CQP operation described in Section 38.36.3.5.4.

Following is a list of few UDA specific errors and their mapping to iWARP asynchronous error codes described in Section 38.36.2.6.

- Software attempted to post SQ WQE on non-privileged QP without specifying a valid address handle index. Reported AE is AE_PRIV_OPERATION_DENIED.
- Software attempts to transmit UDA packet using address handle with IPv4 destination IP address, via QP with IPv6 local IP address, or vise versa. Reported asynchronous error is AE_UDA_XMIT_IPADDR_MISMATCH.

38.36.5.8 User-Space Direct Access Descriptor Formats

38.36.5.8.1 UDA SQ WQE Format - Send

Table 38-513.UDA SQ Send WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	Tagged_Offset		
8	[63:32]	STag	[31:0]	Fragment_Length
16	[63] [62:56] [55] [54:48] [47:41] [40:32]	 MACLEN RSVD IPLEN RSVD	[31:30] [29:28] [27:24] [23:16] [15:0]	L4T IIPT L4LEN
24	[63] [62] [61] [60] [59] [58] [57] [56]	WQE_Valid Signaled_Completion RSVD (local fence must be 0) RSVD (Read_Fence, must be 0) RSVD (wait_for_rcvFPDU, must be 0) RSVD (Streaming_Mode, must be 0) RSVD(Inline_Data_Flag) Push WQE	[55:4] [45] [44] [43:42] [41] [40:38] [37:32] [31:0]	RSVD RSVD DoLoopback RSVD AdditionalFragmentCount OP RSVD

OP (6)

This WQE format is valid for the send operations. See Section 38.38 for the associated opcode values.

WQE_Valid (1), Signaled_Completion (1), PUSH WQE (1) and AdditionalFragmentCount (3)

See Section 38.38.

Tagged_Offset (64), STag (32), Fragment_Length (32)

These fields describe the first fragment of the byte stream to be sent. Additional fragments can be added to the WQE by setting *AdditionalFragmentCount* to a non-zero value. Note that the total length of the packet posted to UDA SQ WQE should not exceed MSS configured for that QP.

DoLoopback(1)

Valid only if *PrivilegedHdrGenEn* bit is set in QP context.

If the *DoLoopback* bit is set (1b) hardware should internally switch this packet within same VSI. Multicast packet should be both internally switched and transmitted via external port.

L4T (2)

For use by privileged consumer only. Required to enable hardware checksum generation offload when all packet headers are generated by software.

L4 packet type.

- 00b = RSVD
- 01b = TCP
- 10b = RSVD

- 11b = RSVD

When the *L4T* field is set to the type other than RSVD the L4LEN must be set as well. If L4T is set to TCP, the respective TCP checksum is generated by hardware. When the L4T value is set to 00b (RSVD), the checksum (TCP or UDP) is not generated by hardware.

IIPT (2)

For privileged consumers. Required to enable hardware checksum generation offload when all packet headers are generated by software.

For non-privileged consumers, this field strictly determines the statistic that is incremented.

Inner IP header type.

- 00b = Non-IP packet, or packet type is not defined by software
- 01b = IPv6 packet
- 10b = RSVD
- 11b = IPv4 with IP checksum offload.

This field is not used for IP fragments.

MACLEN (7)

For use by privileged consumer only. Required to enable hardware checksum generation offload when all packet headers are generated by software.

MAC header length defined in words.

IPLEN (7)

For use by privileged consumer only. Required to enable hardware checksum generation offload when all packet headers are generated by software.

IP header length (including IP optional/extended headers) defined in Dwords.

L4LEN (4)

For use by privileged consumer only. Required to enable hardware checksum generation offload when all packet headers are generated by software.

The L4 header length in Dwords. It should be equal or larger than 5 (or 20 bytes) for TCP.

38.36.5.8.2 UDA SQ WQE Format - Send With Inline Data

Table 38-514.UDA SQ Send With Inline Data WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	Data		
8	[63:0]	Data		
16	[63] [62:56] [55] [54:48] [47]	 MACLEN RSVD IPLEN RSVD	[31:30] [29:28] [27:24] [23]	L4T IIPT L4LEN
24	[63] [62] [61] [60] [59] be 0) [58] 0) [57] [56]	WQE_Valid Signaled_Completion RSVD (Local_Fence must be 0) RSVD (Read_Fence, must be 0) RSVD (wait_for_rcvFPDU, must RSVD (Streaming_Mode, must be Inline_Data_Flag Push WQE	[55] [54:48] [47] [46] [45] [44] [43:42] [41] [40:38] [37:32] [31:0]	RSVD Inline Data Length RSVD RSVD DoLoopback RSVD FwdProgConfirm OP RSVD

OP (6)

This WQE format is valid for the send operations. See Section "Common iWARP SQ Descriptor Format Fields" for the associated opcode values.

WQE_Valid (1), Signaled_Completion (1), Push_WQE(1))

See Section "Common iWARP SQ Descriptor Format Fields."

Inline_Data_Flag (1)

The Inline_Data_Flag is not valid for NOP operations. It specifies whether the data is contained inline with the WQE or if the descriptors point to the data. If Inline_Data_Flag == 0b, the WQE is using descriptors and the WQE format is listed in Table 38-500. If Inline_Data_Flag == 1b the WQE has inline data and the WQE format is listed in Table 38-501.

If the Inline_Data_Flag is set, all upper layer protocol headers must be provided as an inline data together with the payload, and cannot be provided as a separate WQE fragment, or inside private header ring buffer.

Inline_Data_Length (7)

Inline_Data_Length indicates the number of bytes included in the WQE and all subsequent additional fragment descriptors.

AdditionalFragmentCount (3)

Must be 0b for the send with inline data.

Inline data length is used to determine the number of additional fragments consumed by the inline data.

DoLoopback(1)

Valid only if the *PrivilegedHdrGenEn* bit is set in QP context.

If the *DoLoopback* bit is set (1b) hardware should internally switch this packet within the same VSI. Multicast packets should be both internally switched and transmitted via an external port.

No_Checksum(1)

0b = Upper protocol checksum for the supported protocols (TCP) should be generated by hardware.

1b = Hardware should not generate an upper protocol checksum, regardless of the protocol type. Software is expected to perform a checksum calculation and provide a valid checksum. This option must be used for IP fragments.

IIPT(2)

See Section 38.36.5.8.1.

L4T(2), L4LEN(4), IPLEN(7), MACLEN(7)

For use by privileged consumers only. Valid only if *Address_Vector_Valid* is clear (0b). Required to enable hardware checksum generation offload when all packet headers are generated by software. See Section 38.36.5.8.1.

38.36.5.8.3 UDA SQ WQE Format - NOP

Table 38-515.UDA SQ Send WQE Format

Byte Offset	[Bit Range]Field Name			
0	[63:0]	RSVD((Tagged_Offset)		
8	[63:32]	RSVD(STag)	[31:0]	RSVD(Fragment_Length)
16	[63:41] [40:32] (Extended_Header_Length	RSVD RSVD	[31:24] [23:16] Next_Header) [15:0] Index)	RSVD RSVD(Protocol/ RSVD(Address Handle
24	[63] [62] [61] [60] [59] [58] [57] [56]	WQE_Valid Signaled_Completion RSVD (Local_Fence must be 0) RSVD (Read_Fence must be 0) RSVD (wait_for_rcvFPDU, must be 0) RSVD (Streaming_Mode, must be 0) RSVD(Inline_Data_Flag) Push WQE	[55:47] [46] [45] [44] [43:42] [41] [40:38] [37:32] [31:0]	RSVD RSVD(Address_Vector_Valid) RSVD RSVD(DoLoopback) RSVD RSVD(FwdProgConfirm) AdditionalFragmentCount OP RSVD

OP (6)

This WQE format is valid for the NOP operations. See Section "Common iWARP SQ Descriptor Format Fields" for the associated opcode values.

WQE_Valid (1), Signaled_Completion (1), Push_WQE(1), and AdditionalFragmentCount (3)

See Section "Common iWARP SQ Descriptor Format Fields".

AdditionalFragmentCount (3)

Must be zero.

38.36.5.8.4 UDA RQ WQE Format

One RQ WQE provides application buffers for a single received Ethernet frame. The application should provide sufficient buffers to include the entire Ethernet frame, both headers and payload. If RQ is configured with a header/data separation option, then both payload and header buffers should be large enough to include the biggest header

and payload supported by the application. If either payload or header size exceeds the buffers provided by RQ WQE, hardware truncates the header or payload respectively, and reports an error in CQE.

Entire packet (both header and data), are placed in the buffers provided by RQ WQE, starting with the first WQE fragment.

Table 38-516.UDA RQ WQE Format

Byte Offset	[Bit Range]	Field Name		
0	[63:0]	Tagged Offset		
8	[63:32]	STag	[31:0]	Fragment Length
16	[63:0]			
24	[63] [62:54] [53:41]	WQE_Valid RSVD RSVD	[40:38] AdditionalFragmentCount [37:32] [31:0]	 RSVD (OP) RSVD

WQE_Valid (1) and AdditionalFragmentCount (3)

See Section "Common iWARP SQ Descriptor Format Fields."

Tagged_Offset (64), STag (32), Length (32)

These fields describe the first fragment of the byte stream to be sent. Additional fragments might be added to the WQE by setting *AdditionalFragmentCount* to a non-zero value.

38.36.6 PE Statistics

38.36.6.1 Summary

Table 38-517 lists the RFCs relevant to the definition of PE statistics.

Table 38-517.Summary of MIBs Supported With Product Name Hardware Statistics Counters

RFC	Description
1213	Title: Management Information Base for Network Management of TCP/IP-Based Internets: MIB-II. Status: RFC1213 obsoletes RFC1158. RFC1213 is updated by RFC 2011(IP), 2012(TCP), 2013(UDP), 2863(Interfaces). There are MIB groups in RFC1213 that will be implemented by the 10 GbE controller SNMP agent (such as the System group) but these do not require the 10 GbE controller hardware accelerations.
2011	Title: SNMPv2 MIB for the Internet Protocol using SMIv2. Description: Defines objects for managing implementations of IPv4 and ICMP. Status: Updates RFC1213, Obsoleted by RFC4293.
2012	Title: SNMPv2 MIB for the Transmission Control Protocol using SMIv2. Description: Defines objects for managing implementations of TCP. Status: Updates RFC1213, Obsoleted by RFC4022.
2013	Title: SNMPv2 MIB for the User Datagram Protocol using SMIv2. Description: Defines objects for managing implementations of UDP. Status: Updates RFC1213, Obsoleted by RFC4113.
4293	Title: MIB for the Internet Protocol (IP). Description: Defines objects for managing implementations of IPv4 and ICMP. Status: Obsoletes RFC2011, PROPOSED STANDARD.
4022	Title: MIB for the Transmission Control Protocol (TCP). Description: Defines objects for managing implementations of TCP. Status: Obsoletes RFC2012, PROPOSED STANDARD.

Table 38-517.Summary of MIBs Supported With Product Name Hardware Statistics Counters

RFC	Description
4113	Title: MIB for the User Datagram Protocol (UDP). Description: Defines objects for managing implementations of UDP. Status: Obsoletes RFC2013, PROPOSED STANDARD.

There is a set of PE statistics instantiated for each virtual and physical PCI function. The 10 GbE controller supports 16 RDMA enabled PFs and up to 32 RDMA enabled VFs. Mapping of VFs to PFs is not static. PFs have an instance of each PE statistics register. VF PE statistics registers are accessed via PF address space, and indexed by HMC VF index. This means there are 48 instances of the statistics set are listed in Table 38-518. Stats colored pink are new for the 10 GbE controller.

Naming convention for the PE Statistics register is as following:

- Physical function instance starts with GLPES_PF prefix following by the register name.
- Virtual function instance starts with GLPES_VF prefix following by the register name.

Table 38-518 lists PE registers by their physical function names. Virtual function register name can be recovered by substituting prefix.

Table 38-518.PE Statistics Instantiated Per-PCI Function

Name	Size	MIB	Description
GLPES_PFRXVLANERR	24b	private	Ethernet received packets with incorrect VLAN_ID.
GLPES_PFIP4RXOCTS	48b	IP	IPv4 octets received.
GLPES_PFIP4RXPKTS	48b	IP	IPv4 packets received.
GLPES_PFIP4RXDISCARD	32b	IP	IPv4 packets received and discarded.
GLPES_PFIP4RXTRUNC	32b	IP	IPv4 packets received and truncated due to insufficient buffering space in UDA RQ.
GLPES_PFIP4RXFRAGS	48b	IP	IPv4 fragments received.
GLPES_PFIP4RXMCPKTS	48b	IP	IPv4 multicast packets received.
GLPES_PFIP4RXMCOCTS	48b	IP	IPv4 multicast octets received.
GLPES_PFIP6RXOCTS	48b	IP	IPv6 octets received.
GLPES_PFIP6RXPKTS	48b	IP	IPv6 packets received.
GLPES_PFIP6RXDISCARD	32b	IP	IPv6 packets received and discarded.
GLPES_PFIP6RXTRUNC	32b	IP	IPv6 packets received and truncated due to insufficient buffering space in UDA RQ.
GLPES_PFIP6RXFRAGS	48b	IP	IPv6 fragments received.
GLPES_PFIP6RXMCPKTS	48b	IP	IPv6 multicast packets received.
GLPES_PFIP6RXMCOCTS	48b	IP	IPv6 multicast octets received.
GLPES_PFIP4TXOCTS	48b	IP	IPv4 octets supplied by the PE to the lower layers for transmission.
GLPES_PFIP4TXPKTS	48b	IP	IPv4 packets supplied by the PE to the lower layers for transmission.
GLPES_PFIP4TXFRAGS	48b	IP	IPv4 fragments transmitted.
GLPES_PFIP4TXMCPKTS	48b	IP	IPv4 multicast packets transmitted.
GLPES_PFIP4TXMCOCTS	48b	IP	IPv4 multicast octets transmitted.
GLPES_PFIP6TXOCTS	48b	IP	IPv6 octets supplied by the PE to the lower layers for transmission.
GLPES_PFIP6TXPKTS	48b	IP	IPv6 packets supplied by the PE to the lower layers for transmission.

Table 38-518.PE Statistics Instantiated Per-PCI Function

Name	Size	MIB	Description
GLPES_PFIP6TXFRAGS	48b	IP	IPv6 fragments transmitted.
GLPES_PFIP6TXMCPKTS	48b	IP	IPv6 multicast packets transmitted.
GLPES_PFIP6TXMCOCTS	48b	IP	IPv6 multicast octets transmitted.
GLPES_PFIP4TXNOROUTE	24b	IP	IPv4 datagrams discarded due to routing problem (no hit in ARP table).
GLPES_PFIP6TXNOROUTE	24b	IP	IPv6 datagrams discarded due to routing problem (no hit in ARP table).
GLPES_PFTCPRXSEGSLO	48b	TCP	TCP segments received.
GLPES_PFTCPRXOPTERR	24b	Intel	TCP segments received with unsupported TCP options or TCP option length errors.
GLPES_PFTCPRXPROTOERR	24b	Intel	TCP segments received that are dropped by TRX due to TCP protocol errors.
GLPES_PFTCPTXSEG	48b	TCP	TCP segments transmitted.
GLPES_PFTCPRTXSEG	32b	TCP	Total number of TCP segments retransmitted.
GLPES_PFRDMARXWRS	48b	Microsoft	RDMAP total RDMA write messages received.
GLPES_PFRDMARXRDS	48b	Microsoft	RDMAP total RDMA read request messages received.
GLPES_PFRDMARXSNDS	48b	Microsoft	RDMAP total RDMA send-type messages received.
GLPES_PFRDMATXWRS	48b	Microsoft	RDMAP total RDMA write messages sent.
GLPES_PFRDMATXRDS	48b	Microsoft	RDMAP total RDMA read request messages sent.
GLPES_PFRDMATXSNDS	48b	Microsoft	RDMAP total RDMA send-type messages sent.
GLPES_PFRDMAVBND	48b	Microsoft	RDMA verbs total bind operations carried out.
GLPES_PFRDMAVINV	48b	Microsoft	RDMA verbs total invalidate operations carried out.

Note: A simple, conservative estimate for octet counter wrap times at 40 Gb/s

Note: 32-bit counter wraps in 0.859 s.

Note: 36-bit counter wraps in 13.744 s.

Note: 40-bit counter wraps in 3.665 m.

Note: 48-bit counter wraps in 15.64 h.

Note: 56-bit counter wraps in 166.8 d.

There is a set of PE statistics instantiated only once. Table 38-519 lists these statistics, which are all defined by Intel and not part of a MIB.

Table 38-519.PE Intel-Specific Statistics Instantiated Once (Sheet 1 of 2)

Name	Size	Description
GLPES_RDMARXUNALIGN	32b	RDMA stat: TCP segments that probably have unaligned FPDUs.
GLPES_RDMARXMULTFPDUS	56b	RDMA stat: TCP segments that probably have multiple FPDUs.
GLPES_RDMARXOOONOMARK	32b	RDMA stat: FPDUs received out-of-order, with no MPA marker.
GLPES_RDMARXOOODDPLO	56b	RDMA stat: Number of out-of-order placed DDP segments.
GLPES_TCPRXPUREACKS	56b	TCP stat: Number of pure ACKs received.
GLPES_TCPRXONEHOLE	56b	TCP stat: Increments when an accelerated connection opens a first TCP hole.
GLPES_TCPRXTWOHOLE	56b	TCP stat: Increments when an accelerated connection opens a second TCP hole.
GLPES_TCPRXTHREEHOLE	56b	TCP stat: Increments when an accelerated connection opens a third TCP hole.
GLPES_TCPRXFOURHOLE	56b	TCP stat: Increments when an accelerated connection opens a fourth TCP hole.

Table 38-519.PE Intel-Specific Statistics Instantiated Once (Sheet 2 of 2)

Name	Size	Description
GLPES_TCPTXRETRANSFAST	56b	TCP stat: Number of TCP re-transmits.
GLPES_TCPTXTOUTSFAST	56b	TCP stat: Number of TCP retransmission timeouts on connections attempting fast re-transmit.
GLPES_TCPTXTOUTS	56b	TCP stat: Number of TCP retransmission timeouts on connections that are *not* currently attempting fast re-transmit.

38.36.7 SR-IOV PE Functionality

The 10 GbE controller PE features are supported in virtualized operating systems that support SR-IOV direct assignment and IOMMUs. The 10 GbE controller also supports para-virtualized drivers in virtualized operating systems environments. The goal for the 10 GbE controller PE programming model for VFs is to preserve as much of the non-virtualized programming model as possible to maximize the existing software investment. HMC resource profiles provide resource distribution/partitioning required to support this independent programming model in the VF. Backing pages for VF HMC pages for most HMC objects and page descriptor pages are allocated from the PF driver to provide improved security and stability. Backing pages for VF HMC PBL objects are allocated in the guest operating systems address space, the doorbell page from the VF BAR is mapped to guest user-space addresses and CQP operations are issued directly from the guest. The differences are in the programming model for VFs and show up in two areas:

- HMC objects— A HMC PCI function needs to be allocated for the VF and a few HMC objects are owned by the PF instead of the VF (specifically multicast groups and PE quad hash objects), additionally most HMC objects are allocated and managed by the PF driver. The VF driver uses the VF-to-PF mailbox or operating system-specific back channel mechanism to coordinate this activity.
- Interaction with the LAN portion of the software device driver.

In order to enable a VF driver to take advantage of PE functionality, an HMC PCI function must be allocated. There are 32 HMC VF PCI functions available in the 10 GbE controller. Firmware allocates HMC PCI functions on an as needed basis between the PFs on a given adapter. This allocation takes place when the VF is allocated by the PF driver to a given guest operating system. In the case of multicast group HMC objects, these objects are always owned by PFs and require the VF driver to coordinate access to them by using VF-to-PF communication mechanisms. The difference in interaction with the LAN portion of the driver is operating system specific due to the different approaches that are taken on each vendor.

A sample initialization flow is as follows on top of the usual LAN VF initialization of interrupts, etc.:

1. When the PF driver created the PF CQP instance, it selected one of the SR-IOV HMC resource profiles to enable RDMA for VFs.
2. The PF driver (typically in the hypervisor or privileged host VM) allocates an HMC function on behalf of the VF using the PF CQP instance. (see Section 38.36.3.4.1 for the WQE format).
3. The VF driver in the guest creates a new CQP instance. Section 38.36.3.2.1.
4. The VF driver (or optionally the PF driver on the VF drivers behalf) issues a query FPM values operation. Section 38.36.3.5.2.
5. The VF driver (or optionally the PF driver on the VF drivers behalf) issues a commit FPM values operation. Section 38.36.3.5.3.

6. The PF driver allocates backing pages and page descriptors for all static HMC resources required for the VF and programs the SDs for the HMC function assigned to the VF using Update PE SDs operations. Section 38.36.3.5.1. Note that backing pages are not allocated for PBLE HMC objects in the model, but page descriptors are. Also note that direct backing pages are not supported for VF PBLE HMC objects.
7. The VF driver (or optionally the PF driver on the VF drivers behalf) issues a static HMC pages allocated operation. Section 38.36.3.5.10.
8. The PF driver (via communication from the VF driver) allocates backing pages and page descriptors for the initial set of HMC resources required by the VF driver. This step involves issuing one or more Update PE SD operations.
9. The VF driver uses its CQP instance to create CQ0, the CEQ, and AEQ for the VF.
10. During runtime, the VF driver signals the PF driver to allocate and initialize more HMC backing pages and/or PDs on its behalf. When PBLEs are needed during memory registration in the VF, the VF driver issues manage VF PBLE backing pages operations. Section 38.36.3.3.9.

38.36.7.1 SR-IOV Protocol Engine State Migration

Given that the PE uses the HMC's FPM concept, it is theoretically possible to migrate an entire VF's PE state (QPs, CQs, etc) from one adapter to another even if the second adapter is not on the same computer. Enablement of this type of migration is required and substantial.

38.37 System Manageability

Network management is an important requirement in today's networked computer environment.

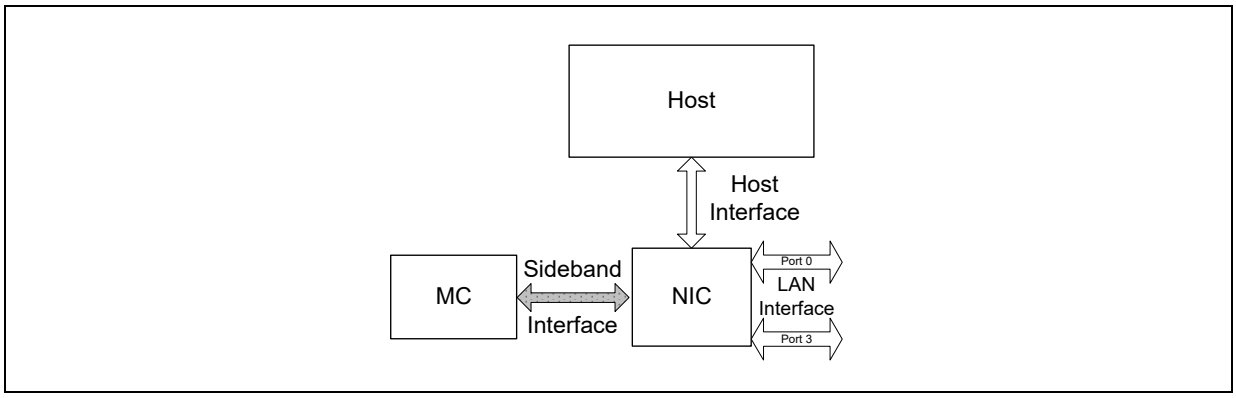
Software-based management applications provide the ability to administer systems while the operating system is functioning in a normal power state (not in a pre-boot state or powered-down state). The Intel® Out of Band Management fills the management void that exists when the operating system is not running or fully functional. This is accomplished by providing mechanisms by which manageability network traffic can be routed to and from a Management Controller (MC).

This section describes the supported management interfaces and hardware configurations for platform system management. It describes the interfaces to an external MC, the partitioning of platform manageability among system components, and the functionality provided by the 10 GbE controller in each platform configuration.

38.37.1 Pass-Through (PT) Functionality

PT is the term used when referring to the process of sending and receiving Ethernet traffic over the sideband interface. The 10 GbE controller has the ability to route Ethernet traffic to the host operating system as well as the ability to send Ethernet traffic over the sideband interface to an external MC. See Figure 38-128.

Figure 38-128.Sideband Interface



The sideband interface provides a mechanism by which the 10 GbE controller can be shared between the host and the MC. By providing this sideband interface, the MC can communicate with the LAN without requiring a dedicated Ethernet controller. The 10 GbE controller supports three sideband interfaces:

- SMBus (legacy or as part of MCTP)
- NC-SI
- PCIe (together with MCTP) - when the system is up.

Note: The usable bandwidth for either direction is up to 1 Mb/s when using SMBus and 100 Mb/s for the NC-SI interface. When working over PCIe, the bandwidth is limited only by the PCIe bandwidth and the the 10 GbE controller processing capabilities and can sustain any network bandwidth. The 10 GbE controller should support MCTP over PCIe pass-through traffic at a rate of up to 1 Gb/s. The maximum packet size supported for traffic received from the LAN to the MC is 1518 bytes and an additional STag or VLAN tags. For traffic from the MC to the LAN the maximum supported packet size is 1536 bytes including all tags.

Note: In MCTP mode, the PCIe and SMBus interface can receive MCTP commands in parallel. For example, the MCTP enumeration process can be done both over SMBus and over PCIe. However, only one of the interfaces can receive NC-SI commands or pass-through traffic.

38.37.1.1 Supported Topologies

The 10 GbE controller can support up to two channels to management controllers in some topologies. The following connections are available:

- Connection via legacy SMBus.
- Connection via NC-SI over RBT (RMII) (see Section 38.37.6).
- Connection via NC-SI over MCTP. This connection can be over SMBus, PCIe or both. This connection can be used either for pass through or for control only (see Section 38.37.7).
- Two connections — A connection via NC-SI over RBT for pass-through traffic and a connection via MCTP used only for control traffic (see Section 38.37.7).

The channel used is for pass through is defined in the *Redirection Sideband Interface* field in the Common Manageability Parameters NVM word and is common to all the ports in the device.

38.37.1.2 PT Packet Routing

When an Ethernet packet reaches the 10 GbE controller, it is examined and compared to a number of configurable filters. These filters are configurable by the MC and include, but not limited to, filtering on:

- MAC address
- IP address
- UDP/IP ports
- VLAN tags
- Ether-type

If the incoming packet matches any of the configured filters, it is passed to the MC. Otherwise, it is not passed.

The packet filtering process is described in Section 38.37.3.

38.37.2 Components of the Sideband Interface

There are two components to a sideband interface:

- Physical layer
- Logical layer

38.37.2.1 Physical Layer

This is the electrical connection between the 10 GbE controller and the MC.

38.37.2.1.1 SMBus

The SMBus physical layer is defined by the SMBus specification. The interface is made up of two connections: data and clock. There is also an optional third connection: the alert line. This line is used by the 10 GbE controller to notify the MC that there is data available for reading in legacy SMBus mode. Refer to the SMBus specification for details.

The SMBus can run at three speeds: 100 kHz (standard SMBus), or 400 kHz (I^2C fast mode) or 1 MHz (I^2C fast mode plus).The speed used while the 10 GbE controller is the master of the bus is selected by the *SMBus Connection Speed* field in the SMBus Notification Timeout and Flags NVM word. When acting as a slave, the 10 GbE controller can receive transactions with a clock running at up to 1 MHz.

38.37.2.1.2 PEC Support

SMBus transactions can be protected by using Packet Error Code (PEC). Packet error checking, when applicable, is implemented by appending a PEC byte at the end of each message transfer. The PEC byte is a CRC8 calculation on all the message bytes.

PEC is added in transmit and expected in receive for the following SMBus packets:

- ARP packets
- MCTP over SMBus transactions.

For ARA cycles and legacy SMBus transactions, a PEC is not expected.

The following table lists the behavior of the 10 GbE controller in each PEC configured mode for transactions directly handled by hardware after receiving packets with or without PEC.

Table 38-520.SMBus PEC Modes[1]

SMBus Transaction/PEC Mode		Target PEC Mode	
SMBus Transaction (Relative to the 10 GbE controller)	**Product Name PEC Mode**	**PEC Enabled**	**PEC Disabled**
Master Write[2]	Enabled	(A) Target ACKs the PEC byte.	(A) Target NACKs the PEC byte.
Master Write	Disabled	(A) Target receives stop before expected PEC byte.	(A) PEC byte is not expected.
Slave Write[3]	Enabled	(A) Target ACKs last data byte; PEC byte is NACKed.	(A) Target NACKs last data byte; No PEC byte is written by the slave.
Slave Write-	Disabled	(A) Target ACKs last data byte; PEC byte is 0xFF.	(A) Target NACKs last data byte and generates stop afterwards.
Slave Read[4]	Enabled	(A) Target sends the PEC byte; PEC byte is ACKed by the slave.	(A) Target does not send PEC byte and generates stop afterwards.
Slave Read	Disabled	(R) Target sends the PEC byte; PEC byte is NACKed by the slave.	(A) Target does not send PEC byte and generates stop afterwards.

Notes:

1. (A) - Accept transaction; (R) - Reject transaction.
2. Used in Legacy SMBus writes commands (direct receive) and in MCTP over SMBus (transmitted transactions.
3. Used in Legacy SMBus Read commands.
4. Used in Legacy SMBus mode (alert/async-notify) and in MCTP over SMBus (received transactions).

Note: In both SMBus ARP and MCTP, the specification indicates that PEC must be used. However, if PEC is not used by the master, the transaction is still accepted and processed by the 10 GbE controller.

The PEC behavior is controlled by the SMBus transaction PEC bit in the SMBus Notification Timeout and Flags NVM word. If this bit is set, PEC is added for master SMBus write transactions. A PEC is added to slave read transactions and can be received in slave write transaction. If this bit is cleared, PEC is not added to master write or slave read transactions, a slave write transaction with PEC is dropped. This bit should be set for MCTP mode and should be cleared in legacy SMBus mode.

38.37.2.1.3 NC-SI

The 10 GbE controller uses the DMTF standard sideband interface. This interface consists of seven lines for transmission and reception of Ethernet packets and two optional lines for arbitration among more than one physical network controller.

The physical layer of NC-SI is very similar to the RMII interface, although not an exact duplicate. Refer to the NC-SI specification for details of the differences.

Electrical characteristics

The 10 GbE controller complies with the electrical characteristics defined in the NC-SI 1.0.1 specification.

The 10 GbE controller's NC-SI behavior is configured at power-up in the following manner:

- The multi-drop NC-SI NVM bit defines the NC-SI topology (point-to-point or multi-drop; the default is point-to-point).

The 10 GbE controller dynamically drives its NC-SI output signals (NC-SI_DV and NC-SI_RX) as required by the sideband protocol:

- At power-up, the 10 GbE controller floats the NC-SI outputs.
- If the 10 GbE controller operates in point-to-point mode, it starts driving the NC-SI outputs some time following power-up.
- If the 10 GbE controller operates in a multi-drop mode, it drives the NC-SI outputs as configured by the BMC.

38.37.2.1.4 PCIe Vendor Defined Messages (VDMs)

The 10 GbE controller uses VDMs over PCIe defined in the DMTF MCTP specification to convey pass-through traffic or NC-SI control traffic.

38.37.2.2 Logical Layer

38.37.2.2.1 Legacy SMBus

The protocol layer for SMBus consists of commands the MC issues to configure filtering for the 10 GbE controller management traffic and the reading and writing of Ethernet frames over the SMBus interface. There is no industry standard protocol for sideband traffic over SMBus. The protocol layer for SMBus on the 10 GbE controller is Intel proprietary.

38.37.2.2.2 NC-SI

The DMTF defines the protocol layer for the NC-SI interface. NC-SI compliant devices are required to implement a minimum set of commands. The specification also provides a mechanism for vendors to add additional capabilities through the use of OEM commands. Intel OEM NC-SI commands for the 10 GbE controller are discussed in Section 38.37.6.4. For information on base NC-SI commands, see the NC-SI specification.

NC-SI traffic can run on top of three different Physical layers:

1. NC-SI physical layer as described in Section 38.37.2.1.3.
2. MCTP over PCIe VDM.This protocol enables control and pass-through traffic over PCIe of a NIC or a LOM device. The NC-SI over MCTP protocol is slightly different than the standard NC-SI as it includes additional NC-SI commands. This mode is usually paired with an MCTP over SMBus, where this mode is used in S0 states and the SMBus interface is used in Sx state.The MCTP protocol and the differences from standard NC-SI is described in Section 38.37.7.
3. MCTP over SMBus. As previously described, this layer is paired with the MCTP over PCIe to support Sx modes.

The 10 GbE controller exposes one NC-SI package with four channels, one per port. The 10 GbE controller implements a type C NC-SI interface (single package, common bus buffers and shared RX queue) as described in Section 5.2 of the *NC-SI Specification*.

Package ID Setting

The package ID can be set from the NVM *Package ID* field in the NC-SI Configuration 1 NVM word.

Channel ID Mapping

The mapping of the channels to physical ports is according to the NC-SI Channel to Port Mapping NVM word if the NC-SI Channel to Port Mapping Table *Valid* bit is set. If this bit is not set, the following algorithm should be used:

```
Channel_ID = 0

NC-SI_channel[3 :0] = -1 // ports not associated to channels yet.

For func = 0 to 15 { // loop on all functions

        Port_ID = PFGEN_PORTNUM[func] // Port associated with function.

        If (PRTGEN_STATUS.PORT_VALID[Port_ID]  && NC-SI_channel[Port_ID] == -1 ) {

        // Port is valid and port is not already associated to a channel

        NC-SI_channel[Port_ID] = Channel_ID; // assign channel

        Channel_ID++; // go to next channel

        }

    }
```

This algorithm maps channel numbers that match the order of the PCI function numbers. If more than one function is defined on a port, the function with the lowest value associated with this port is used.

38.37.3 Packet Filtering

Since both the host operating system and an MC use the 10 GbE controller to send and receive Ethernet traffic, there needs to be a mechanism by which incoming Ethernet packets can be identified as those that should be sent to the MC rather than the host operating system.

There are two different types of filtering available. The first is filtering based upon the MAC address. With this filtering, the BMC has at least one dedicated MAC address and incoming Ethernet traffic with the matching MAC address(es) are passed to the MC. This is the simplest filtering mechanism to use and it enables the MC to receive all types traffic (including, but not limited to, IPMI, NFS, HTTP etc).

The other type available uses a highly configurable mechanism by which packets can be filtered using a wide range of parameters. Using this method, the MC can share a MAC address (and IP address, if desired) with the host operating system and receive only specific Ethernet traffic. This method is useful if the MC is only interested in specific traffic, such as IPMI packets.

38.37.3.1 Manageability Receive Filtering

This section describes the manageability receive packet filtering flow. Packet reception by the 10 GbE controller can generate one of the following results:

- Discarded
- Sent to host memory
- Sent to the MC
- Sent to both the MC and host memory

The decisions regarding forwarding of packets to the host and to the MC are separate and are configured through two sets of registers. However, the MC might define some types of traffic as exclusive. This traffic is forwarded only to the MC, even if it passes the filtering process of the host. These types of traffic are defined using the PRT_MNG_MNGONLY register.

An example of packets that might be necessary to send exclusively to the MC might be specific TCP/UDP ports of a shared MAC address or a MAC address dedicated to the MC. If the MC configures the manageability filters to send these ports to the MC, it should configure the settings to not send them to the host; otherwise, these ports are received and handled by the host operating system.

The MC controls the types of packets that it receives by programming receive manageability filters. The following filters are accessible to the MC:

Table 38-521.Filters Accessible to MC

Filters	Functionality	When Reset?
Filters Enable	General configuration of manageability filters.	RSMRST#
Manageability Only	Enables routing of packets exclusively to manageability.	RSMRST#
Manageability Decision Filters [7:0]	Configuration of manageability decision filters.	RSMRST#
MAC Address [3:0]	Four unicast MAC manageability addresses.	RSMRST#
VLAN Filters [7:0]	Eight VLAN tag values.	RSMRST#
UDP/TCP Port Filters [15:0]	16 destination port values.	RSMRST#
Flexible 128 bytes TCO Filter	Length and values for one flex TCO filter.	RSMRST#
IPv4 and IPv6 Address Filters [3:0]	IP address for manageability filtering.	RSMRST#
Special Filters Modifier	Used to define some special filtering options like 24-bit filtering of IPv6 addresses and TCP/UDP selection of ASF ports.	RSMRST#

All filtering capabilities are available on both the NC-SI and legacy SMBus interfaces. In NC-SI modes, part of the filters are programmed via standard NC-SI commands and part of the filter are programmed via the Intel OEM commands described in Section 38.37.6.4. In legacy SMBus, the filtering is programmed either from NVM or via the commands previously described.

Note: Internal power on reset flows include a PWRBTN# override event. Refer to the Power Management flows previously described for more detail. Since all filters are reset, the BMC must reconfigure the necessary filters for correct functionality.

The high-level structure of manageability filtering is done using two or three steps.

1. The packet is routed by the switch. If the switch determines the packet should be routed to the manageability VSIs, the next steps are taken.
2. The packet is parsed and fields in the header are compared to programmed filters.
3. A set of decision filters are applied to the result of the first step.

The following sections describe steps 2 and 3 previously listed.

Some general rules apply:

- Fragmented packets are passed to manageability but not parsed beyond the IP header.
- Packets with L2 errors (CRC, alignment, etc.) are not forwarded to the MC.
- Packets longer than 2 KB are filtered out.

The following sections describe the manageability filtering, followed by the final filtering rules.

The filtering rules are created by programming the decision filters as described in Section 38.37.3.4.

All filters are reset only on internal power on reset. Register filters that enable filters or functionality are also reset by a firmware reset in NC-SI mode. These registers can be loaded from the NVM following a reset in SMBus mode. Refer to the NVM map for more details.

Note: Internal power on reset flows include the PWRBTN# override event. Refer to the Power Management flows previously described for more detail. Since all filters are reset, the BMC must reconfigure the necessary filters for correct functionality. Internal power on reset flows include a PWRBTN# override event. Refer to the Power Management flows previously described for more detail. Since all filters are reset, the BMC must reconfigure the necessary filters for correct functionality.

This reconfiguration is required because the management controller inside Lewisburg is also included in the global reset flow and the NC-SI configuration data is reset. Registers such as the BMC MAC address will require reconfiguration. Since BMC bring up network in first boot, the BMC firmware will need to repeat the network bring-up flow after a global reset event. There should be no network link drop after a host partition reset. Implementation options vary depending on BMC implementations. Below are available options:

1. If GBL_RST_WARN pin is already routed to the BMC, the BMC will get notice when the pin is asserted and be aware of a global reset occurring, and re-configure the network interface accordingly.

2. If GBL_RST_WARN pin is not routed to BMC, and there are no plan for a board spin, then there is a need to find a way for the BMC to be aware of a global reset, or link drop event, so that the BMC can do the re-configuration. One likely and reliable way is to let the BIOS notify the BMC through an existing communication channel, such as a GPIO pin.

3. The BMC can monitor eSPI_RESET# as it is triggered on a global reset.

4. (SLP_S3# or SLP_S4# or SLP_S5#) and PLT_RST# will be triggered on different resets. In order to verify if the Management filters in X722 were cleared on the global reset, the BMC can send any NC-SI command besides "Clear Initial State" and if the command succeeds then there is no additional action. If the command responds that initialization is required (Response code 0x01 Reason code 0x01 "Interface Initialization Required") then the BMC will need to re configure the NC-SI interface as well as the pass-through filters.

Global resets are a rare occurrence; however, the network controller will need to be reconfigured after this type of reset.

38.37.3.2 L2 Filters

38.37.3.2.1 MAC and VLAN Filters

The manageability MAC filters allow a comparison of the destination MAC address to one of 4 filters defined in the PRT_MNG_MMAH and PRT_MNG_MMAL registers.

The VLAN filters allow a comparison of the 12 bit inner VLAN tag to one of 8 filters defined in the PRT_MNG_MAVTV registers.

38.37.3.2.2 Ether-type Filters

Manageability L2 Ether-type filters enable filtering of received packets based on the Layer 2 Ether-type field. The L2 type field of incoming packets is compared against the Ether-type filters programmed in the manageability Ether-type filter (PRT_MNG_METF; up to 4 filters); the result is incorporated into decision filters.

Each manageability Ether-type filter can be configured as pass (positive) or reject (negative) using a polarity bit. In order for the reverse polarity mode to be effective and block certain type of packets, the Ether-type filter should be part of all the enabled decision filters.

An example for using L2 Ether-type filters is to determine the destination of 802.1X control packets. The 802.1X protocol is executed at different times in either the management controller or by the host. L2 Ether-type filters are used to route these packets to the proper agent.

In addition to the flexible Ether-type filters, the 10 GbE controller supports two fixed Ether-type filters used to block NC-SI control traffic (0x88F8) and flow control traffic (0x8808) from reaching the manageability interface. The NC-SI Ether-type is used for communication between the MC on the NC-SI link and the 10 GbE controller. Packets coming from the network are not expected to carry this Ether-type and such packets are blocked to prevent attacks on the MC. Flow control packets should be consumed by the MAC and as such are not expected to be forwarded to the management interface.

Note: Ether-type filters shouldn't configured with IPv4 or IPv6 Ether-type values.

38.37.3.3 L3/L4 Filtering

The manageability filtering stage combines checks done at previous stages with additional L3/L4 checks to make a decision about whether to route a packet to the MC. The following sections describe the manageability filtering done at layers L3/L4 and final filtering rules.

38.37.3.3.1 ARP Filtering

ARP filtering — The 10 GbE controller supports filtering of ARP request packets (initiated externally) and ARP responses (to requests initiated by the MC).

In legacy SMBus mode, the ARP filters can be used as part of the ARP offload described in Section 38.37.5.4. ARP offload is not specifically available when using NC-SI. However, the general filtering mechanism is used to filter incoming ARP traffic as requested using the Enable Broadcast Filtering NC-SI command.

In order to limit the reception of ARP packets to the ARP packets dedicated to this station (ARP target IP = MC IP), the ARP request/response filter can be bound to a specific IP address by setting both the ARP Request/Response and the IP AND bits in an MDEF filter. Note that the IP bit is also set if there is a match on the target IP (the TPA field in the ARP packet) of an ARP request or an ARP response.

Note: If the OR section of the MDEF is cleared and one of the IPv4 address are set, then ARP packets matching the IP address pass the filter. If these packets should be dropped, then an OR Ether-type filter with a a value of 0x0800 (IPv4) should be added.

Refer to the section describing the format of ARP packets.

38.37.3.3.2 Neighbor Discovery Filtering and MLD

The 10 GbE controller supports filtering of the following ICMPv6 packets.

Neighbor discovery packets:

1. 0x86 (134d) - Router Advertisement.
2. 0x87 (135d) - Neighbor Solicitation.
3. 0x88 (136d) - Neighbor Advertisement.
4. 0x89 (137d) - Redirect.

MLD packets:

1. 0x82 (130d) - MLD Query
2. 0x83 (131d) - MLDv1 Report
3. 0x84 (132d) - MLD Done
4. 0x8F (143d) - MLDv2 Report

The neighbor discovery packets has dedicated enables for each type in the decision filters. For MLD, a single enable controls the forwarding of all the MLD packets. This means that either all the MLD packets types are selected for reception or none of them.

Refer to the section describing the format of ICMPv6 packets.

38.37.3.3.3 RMCP Filtering

The 10 GbE controller supports filtering by fixed destination port numbers, port 0x26F and port 0x298. These ports are IANA reserved for RMCP.

UDP or TCP protocols can be included in the comparison using the *PRT_MNG_MSFM.PORT_26F/298_UDP/TCP* fields.

In SMBus mode, there are filters that can be enabled for these ports. When using NC-SI, they are not specifically available. However, the general filtering mechanism can be utilized to filter incoming RMCP traffic.

38.37.3.3.4 ICMP Filtering

The 10 GbE controller supports filtering by ICMPv4. This filter matches if the IP *Protocol* field equals to 1b.

Refer to the section describing the format of ICMP packets.

38.37.3.3.5 Flexible Port Filtering

The 10 GbE controller implements 16 flex destination port filters. The 10 GbE controller directs packets whose L4 destination port matches to the MC. The MC must ensure that only valid entries are enabled in the decision filters.

For each flex port filter, filtering can be enabled for UDP, TCP or both. It can be enabled either on source or destination port.

38.37.3.3.6 IP Address Filtering

The 10 GbE controller supports filtering by destination IP address using IPv4 and IPv6 address filters. These are dedicated to manageability. The 10 GbE controller provides four IPv6 address filters and four IPv4 address filters.

For each IPv6 filter, the matching *PRT_MNG_MSFM*.IPV6_n_MASK bit defines if all the IP address should be compared to the PRT_MNG_MIPAF6 register or only the 24 LSBits should be compared to the 24 LSBits of the PRT_MNG_MIPAF6 register.

The IPv4 match also rises for ARP packets for which the target IP matches the IP address in the PRT_MNG_MIPAF4 register.

38.37.3.3.7 Checksum Filtering

The 10 GbE controller might be instructed to direct packets to the MC only if they pass L3/L4 checksum (if they exist) in addition to matching other filters previously described.

Enabling the XSUM filter when using the SMBus interface is accomplished by setting the *Enable XSUM Filtering to Manageability* bit. This is done using the Update Management Receive Filter Parameters command.

To enable the XSUM filtering when using NC-SI, use the Enable Checksum Offloading command. See Section 38.37.6.4.13.

38.37.3.4 Flexible 128-Byte Filter

The 10 GbE controller provides one flex TCO filter. This filter looks for a pattern match within the first 128 bytes of the packet.

From the first 128 bytes, some of the fields are skipped for this comparison. These are field that are not exposed to the MC. The tag skipped is STag.

The flex filter programming should ignore the presence of these fields.

Note: The flex filter comparison should be disabled in the MDEF registers while the flex filter is being updated.

Note: The flexible filter is not applied to transmit packets and a transmit packet is considered as if it did not pass the filter.

38.37.3.4.1 Flexible Filter Structure

The filter is composed of the following fields:

1. Flexible filter length — This field indicates the number of bytes in the packet header that should be inspected. The field also indicates the minimal length of packets inspected by the filter. Packet below that length is not inspected. Valid values for this field are: 8*n, where n=1…16.
2. Data — This is a set of up to 128 bytes comprised of values that header bytes of packets are tested against.
3. Mask — This is a set of 128 bits corresponding to the 128 data bytes that indicate for each corresponding byte if is tested against its corresponding byte. The general filter is 128 bytes that the MC configures; all of these bytes might not be needed or used for the filtering, so the mask is used to indicate which of the 128 bytes are used for the filter.

Each filter tests the first 128 bytes (or less) of a packet, where not all bytes must necessarily be tested.

38.37.3.4.2 TCO Filter Programming

Programming each filter is done using the following commands (NC-SI or SMBus) in a sequential manner:

1. Filter mask and length — This command configures the following fields:
   a. Mask — A set of 16 bytes containing the 128 bits of the mask. Bit 0 of the first byte corresponds to the first byte on the wire.
   b. Length — A 1-byte field indicating the length.

2. Filter data — The filter data is divided into groups of bytes as follows:

Group	Test Bytes
0x0	0-29
0x1	30-59
0x2	60-89
0x3	90-119
0x4	120-127

Each group of bytes need to be configured using a separate command, where the group number is given as a parameter. The command has the following parameters:

a. Group number — A 1-byte field indicating the current group addressed.

b. Data bytes — Up to 30 bytes of test-bytes for the current group.

Flexible TCO Filter Configuration in NVM (Global MNG Offset 0x05)

This section describes the NVM module used to store the flex filter initial data in SMBus mode.

This module is pointed to by global offset 0x08 of the manageability module header section.

Section Header — Offset 0x0

Bits	Name	Default	Description
15:0	Block Length	0xC	Section length in words (including CRC word and length word).

Flexible Filter Length and Control — Offset 0x01

Bits	Name	Default	Description
15:8	Flexible Filter Length (bytes)		
7:5	Reserved		Reserved.
4	Last Filter		
3	Apply Filter to LAN 3		
2	Apply Filter to LAN 2		
1	Apply Filter to LAN 1		
0	Apply Filter to LAN 0		

Flexible Filter Enable Mask — Offset 0x02 – 0x09

Bits	Name	Default	Description
15:0	Flexible Filter Enable Mask		

Flexible Filter Data — Offset 0x0A – 0x49

Bits	Name	Default	Description
15:0	Flexible Filter Data		

Note: This section loads all of the flexible filters. The control + mask + filter data are repeatable as the number of filters. Section length in offset 0 is for all filters.

Section Footer — Offset Block Length

Bits	Name	Default	Description
15:8	CRC 8		CRC8 of the previous section.
7:0	Reserved		Reserved.

38.37.3.5 Configuring Manageability Filters

There are a number of pre-defined filters that are available for the MC to enable, such as ARPs and IPMI ports 0x298 and 0x26F. These are generally enabled by setting the appropriate bit within the PRT_MNG_MANC register using specific commands.

For more advanced filtering needs, the MC has the ability to configure a number of configurable filters. It is a two-step process to use these filters. They must first be configured and then enabled.

38.37.3.5.1 Manageability Decision Filters

Manageability Decision Filters (MDEF) are a set of eight filters, each with the same structure. The filtering rule for each decision filter is programmed by the MC and defines which of the L2, VLAN, Ether-type and L2/L3 filters participate in decision making. Any packet that passes at least one rule is directed to manageability and possibly to the host.

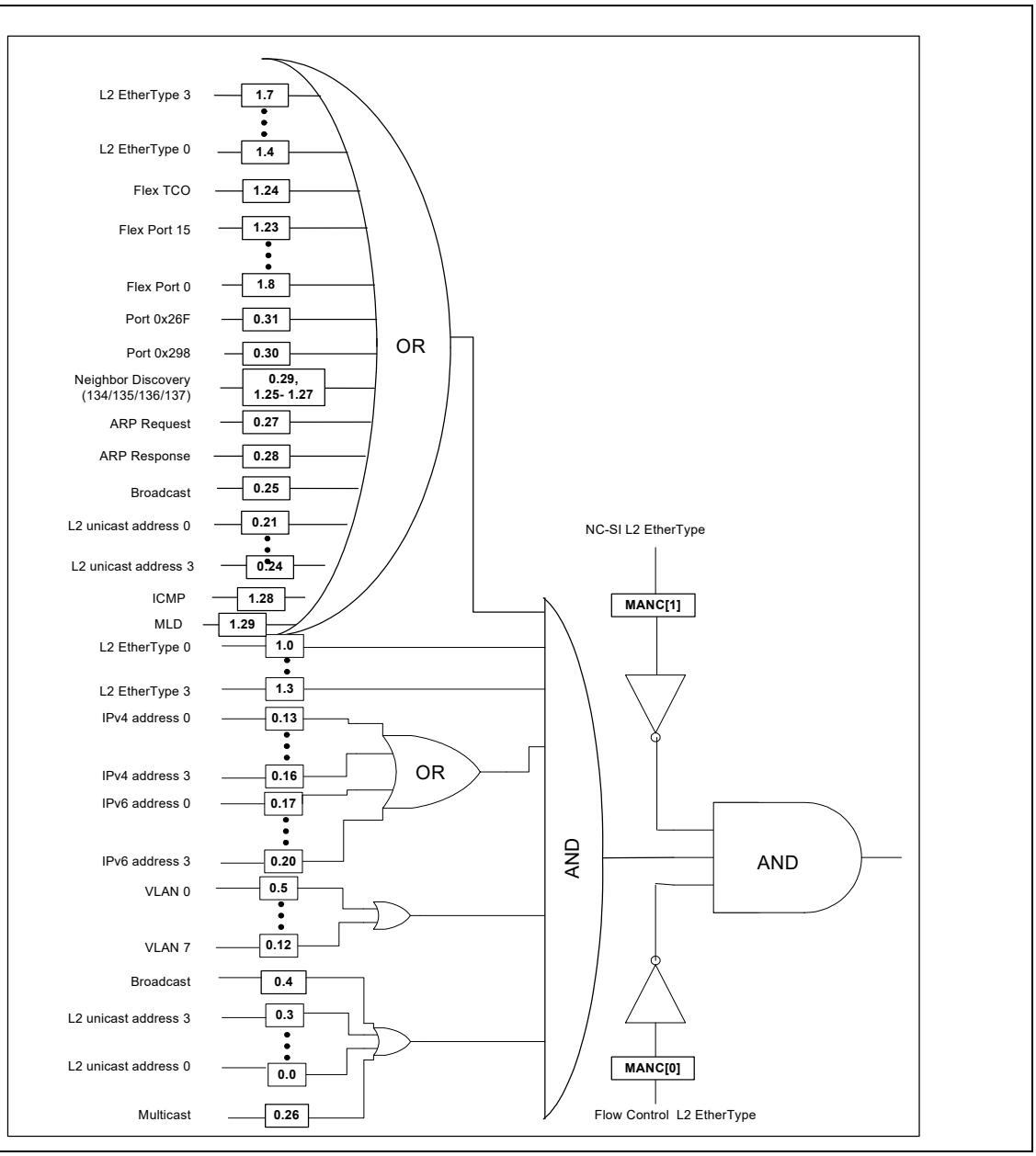
The inputs to each decision filter are:

- Packet passed a valid management L2 exact address filter.
- Packet is a broadcast packet.
- Packet has a VLAN header and it passed a valid manageability VLAN filter.
- Packet matched one of the valid IPv4 or IPv6 manageability address filters.
- Packet is a multicast packet.
- Packet passed ARP filtering (request or response).
- Packet passed neighbor solicitation filtering.
- Packet passed MLD filtering.
- Packet passed 0x298/0x26F port filter.
- Packet passed a valid flex port filter.
- Packet passed a valid flex TCO filter.
- Packet is an ICMPv4 packet.
- Packet passed or failed an L2 Ether-type filter.
- Packet passed or failed Flow Control or NC-SI L2 Ether-type Discard filter.

The structure of each decision filter is shown in Figure 38-129. A boxed number indicates that the input is conditioned by a mask bit defined in the MDEF register and MDEF_EXT register for this rule. Decision filter rules are as follows:

- At least one bit must be set in one of the two registers. If all bits are cleared (MDEF/MDEF_EXT = 0x0000), then the decision filter is disabled and ignored.
- All enabled AND filters must match for the decision filter to match. An AND filter not enabled in the MDEF/MDEF_EXT registers is ignored. If an AND filter is preceded by a OR filter, then at least one of the enabled OR inputs must match for the filter to pass.
- If no OR filter is enabled in the register, the OR filters are ignored in the decision (the filter might still match).
- If one or more OR filters are enabled in the register, then at least one of the enabled OR filters must match for the decision filter to match.

Figure 38-129.Manageability Decision Filters



A decision filter (for any of the eight filters) defines which of the previously described inputs are enabled as part of a filtering rule. The MC programs two 32-bit registers per rule (MDEF[7:0] and MDEF_EXT[7:0]). A set bit enables its corresponding filter to participate in the filtering decision.

In addition to the controls previously described, the *PRT_MNG_MDEF_EXT.apply_to_host_traffic* and *PRT_MNG_MDEF_EXT.apply_to_network_traffic* bits define which traffic is compared to this filter. At least one of these bits must be set for the filter to be valid.

If the *PRT_MNG_MDEF_EXT.apply_to_host_traffic* bit is set, the traffic from the host is a candidate for this filter. If the *PRT_MNG_MDEF_EXT.apply_to_network_traffic* bit is set, the traffic from the network is a candidate for this filter. If both bits are set, this filter is applied to all traffic.

38.37.3.5.2 Exclusive Traffic

The decisions regarding forwarding of packets to the host for LAN traffic or to the LAN for host traffic are independent from the management decision filters. However, the MC might define some types of traffic as exclusive. The behavior for such traffic is defined by the using the bits corresponding to the decision filter in the PRT_MNG_MNGONLY register (one bit per each of the eight decision rules) and the *PRT_MNG_MDEF_EXT.apply_to_host_traffic* and *PRT_MNG_MDEF_EXT.apply_to_network_traffic* bits. Table 38-522 lists the behavior in each case. If one or more filters match the traffic and at least one of the filters is set as exclusive, the traffic is treated as exclusive.

Table 38-522.Exclusive Traffic Behavior

	Filter Match		Filter Does Not Match
Traffic Source	**PRT_MNG_MNGONLY = 0**	**PRT_MNG_MNGONLY = 1**	**N/A**
From network	Traffic is forwarded to manageability. Traffic is forwarded to the host according to host filtering.	Traffic is forwarded only to manageability.	Traffic is forwarded to the host according to host filtering.
From host	Traffic is forwarded to manageability and to the LAN.	Traffic is forwarded only to manageability.	Traffic is forwarded to the LAN.

Any traffic matching any of the configurable filters (see Section 38.37.3.5.1) can be used as filters to pass traffic to the host.

Table 38-523.PRT_MNG_MNGONLY Register Description and Usage

Bits	Description	Default
0	Decision Filter 0	Determines if packets that have passed decision filter 0 are sent exclusively to the manageability path.
1	Decision Filter 1	Determines if packets that have passed decision filter 1 are sent exclusively to the manageability path.
2	Decision Filter 2	Determines if packets that have passed decision filter 2 are sent exclusively to the manageability path.
3	Decision Filter 3	Determines if packets that have passed decision filter 3 are sent exclusively to the manageability path.
4	Decision Filter 4	Determines if packets that have passed decision filter 4 are sent exclusively to the manageability path.
5	Unicast and Mixed	NC-SI mode: Determines if unicast and mixed packets are sent exclusively to the manageability path. SMBus mode: Determines if packets that have passed decision filter 5 are sent exclusively to the manageability path.
6	Global Multicast	NC-SI mode: Determines if multicast packets are sent exclusively to the manageability path. SMBus mode: Determines if packets that have passed decision filter 6 are sent exclusively to the manageability path.
7	Broadcast	NC-SI mode: Determines if broadcast packets are sent exclusively to the manageability path. SMBus mode: Determines if ARP packets are sent exclusively to the manageability path.
31:8	Reserved	Reserved.

When using the SMBus interface, the MC enables these filters by issuing the Update Management Receive Filter Parameters command (with the parameter of 0x0F).

PRT_MNG_MNGONLY is also configurable when using NC-SI using the Set Intel Filters — Manageability Only command.

All manageability filters are controlled by the MC only and not by the LAN software device driver.

Note: PRT_MNG_MNGONLY should not be set for traffic type that the MC might also send to the network (such as multicast traffic). If set, the MC might receive the packets it sent and the packets might not be sent to the network or the host.

38.37.3.5.3 Global Controls

On top of the PRT_MNG_MDEF filters, the PRT_MNG_MANC registers contain some global controls applied to all the packets in order to be a candidate for manageability filtering:

- Receive enable bits:
  - The *RCV_TCO_EN* field controls the reception of manageability traffic. It should be set only if one of the following bits is also set.
  - The *EN_BMC2OS* bit controls the reception of manageability traffic from the host.
  - The *EN_BMC2NET* bit controls the reception of manageability traffic from the network.
- VLAN filtering— In order to support the NC-SI VLAN modes the following controls are provided:
  - The *FIXED_NET_TYPE* field controls if only VLAN tagged or VLAN untagged traffic is received. If this bit is cleared both types are received. If it is set, only the type described by the *NET_TYPE* field is accepted.
  - If set, the NET_TYPE field indicates that only VLAN tagged traffic is received, if cleared only packets without VLAN is accepted. This field is validated by the *FIXED_NET_TYPE* field.

  Both fields relates to the inner VLAN.

Table 38-524 lists the relationship between the previously mentioned bits and the forwarding decisions:

Table 38-524.PRT_MNG_MANC Bits Impact

CASE\ PRT_MNG_MANC Bits	RCV_TCO_EN=0b	FIXED_NET_TYPE= 1b and NET_TYPE!= What's in the Packet	EN_BMC2OS=0b (Assume EN_BMC2NET = 1b)	EN_BMC2NET=0b (Assume EN_BMC2HOST = 1b)
Packet sent from host and hits MDEF filters (host-to-MC traffic).	Packet is not sent to the MC.	Packet is not sent to the MC.	Packet is not sent to the MC.	Packet is sent to the MC.
Packet sent from host and matches one of the EMP VSI (host-to-EMP traffic).	Packet is sent to the EMP.	Packet is sent to the EMP.	Packet is sent to the EMP.	Packet is sent to the EMP.
Packet received from LAN and hits MDEF filters (LAN-to-MC traffic).	Packet is not sent to the MC.	Packet is not sent to the MC.	Packet is sent to the MC.	Packet is not sent to the MC.
Packet received from LAN and matches on of the EMP VSI (LAN-to-EMP traffic).	Packet is sent to the EMP.	Packet is sent to the EMP.	Packet is sent to the EMP.	Packet is sent to the EMP.
Packet sent from EMP and matches one of the host VSIs (EMP-to-host traffic)	Packet is sent to a host (and optionally to LAN).	Packet is sent to the host (and optionally to LAN).	Packet is sent to the host (and optionally to LAN).	Packet is sent to the host (and optionally to LAN).

Table 38-524.PRT_MNG_MANC Bits Impact

CASE\ PRT_MNG_MANC Bits	RCV_TCO_EN=0b	FIXED_NET_TYPE= 1b and NET_TYPE!= What's in the Packet	EN_BMC2OS=0b (Assume EN_BMC2NET = 1b)	EN_BMC2NET=0b (Assume EN_BMC2HOST = 1b)
Packet sent from the EMP and does not match one of the host VSIs (EMP-to-LAN traffic).	Packet is sent to the LAN.	Packet is sent to the LAN.	Packet is sent to the LAN.	Packet is sent to the LAN.
Packet sent from the MC and matches one of the host VSIs (MC-to-host traffic).	Packet is sent to the host (and optionally to LAN).	Packet is sent to the host (and optionally to LAN).	Packet is sent to the LAN.	Packet is sent to the host (and optionally to LAN).
Packet sent from the MC and does not match one of the host VSIs (MC-to-LAN traffic).	Packet is sent to the LAN.	Packet is sent to the LAN.	Packet is sent to the LAN.	Packet is not sent to the LAN (optionally sent to host).

38.37.3.6 Filtering Programming Interfaces

The 10 GbE controller provides multiple options to program the forwarding filters, depending on the interface used and the level of flexibility needed. Table 38-525 lists the different options and points to the description of the relevant commands.

Table 38-525.Filtering Programming Interfaces

Interface	Flexible/Abstract	Description
NC-SI (over RMII or over MCTP)	Abstract (dedicated MAC address)	The regular NC-SI commands can be used to enable forwarding based on a dedicated MAC address. The list of supported commands can be found in Section 38.37.6.2.1. When using these commands, one of the two other modes can be used to add finer grain filtering.
	Abstract (Shared MAC and IP)	The Intel OEM commands described in Section 38.37.3.6.1 and in Section 38.37.6.4.14 can be used to define which part of the shared MAC or shared IP traffic should be forwarded. When using these commands, the flexible filtering interface should not be used. This mode is activated using the Set Shared mode command.
	Flexible	This interface described in most of the sub-sections of Section 38.37.6.4. It uses the packet reduction commands to reduce the forwarding scope of the filters set by the regular NC-SI commands and the packet addition commands to add new packet types to the forwarding rules.
SMBus	Abstract	The Set Common filter command can be used to set the most common filters. When using this commands the flexible filtering interface should not be used. When sending this command, all previous filtering requests are cleared.
	Flexible	The Update MNG RCV Filter Parameters can be used to define the exact filtering rules to be applied.

38.37.3.6.1 Shared MAC and Shared IP Support

The 10 GbE controller operates in systems where the same MAC and IP are shared between a platform's host operating system and its MC. In order to support such systems, the 10 GbE controller supports additional shared MAC filtering options on top of what was supported in previous products. This section describes these options and the NC-SI commands used to program them.

Note: All filtering capabilities are exposed via the regular NC-SI packet reduction and packet addition commands and via the SMBus Set Filtering command. The interface described in this section is a more abstract NC-SI interface.

Sharing an IP and MAC Address

NC-SI over MCTP is used in desktop and mobile platforms. These platforms are typically used in enterprise environments outside of a data center. IP subnets in these environments are commonly designed such that more than 50% of their available addresses are assigned.

Hence, assigning a second IP address to an MC would generally necessitate a subnet redesign. Instead, a single IP address is typically shared between the host operating system and an MC in these platforms.

Because it's possible to bind multiple IP addresses to a single MAC address, the 10 GbE controller needs to know the IP address shared by an MC in order to deliver packets to it. An MC uses the *Set IP Address* command to communicate its IP address to the 10 GbE controller.

In order to notify the 10 GbE controller that the MC intends to use a shared MAC, the *Set Shared Mode* command should be given before programming any filter using the regular NC-SI commands (Set MAC address or Set VLAN) or the Intel OEM commands (Section 38.37.6.4.14).

TCP/UDP Ports Owned by an MC

A small subset of the TCP and UDP ports might be dedicated to an MC. The remaining ports are assigned to the host operating system. Hence, port-based filtering and the commands to configure it is required. For example, port-based filtering would be used to route WS-management packets to an MC.

The 10 GbE controller needs to know the ports owned by an MC in order to deliver packets to it. An MC uses the *Set Port* command to communicate its ports to the 10 GbE controller. The 10 GbE controller supports 10 port filters.

The Set Binding command is used to define the combination of MAC, VLAN, IP and ports that should be met to forward packets to the MC.

Sharing Network Infrastructure Packets

In addition to management traffic, an MC needs to monitor network infrastructure traffic along with the host. For each flow, it is possible to define if it should include host traffic only, both host and network or only network.

ARP Filters Enhancement

ARP request message filtering is controlled by the Enable Broadcast Filter command. However, as currently defined, this command causes either all or no ARP requests to go to an MC. For MCTP over SMBus, attempting to forward all ARP requests within a subnet to an MC can easily overwhelm the available bandwidth. Therefore, an option to have the 10 GbE controller forward only ARP requests that contain a Target IP Address value that matches the IP address used by an MC. An amendment to the Enable Broadcast Filter command is defined in the sections that follow to address this requirement.

38.37.3.7 Possible Configurations

This section describes ways of using management filters. Actual usage might vary.

38.37.3.7.1 Dedicated MAC Packet Filtering

- Select one of the eight rules for dedicated MAC filtering.
- Load the host MAC address to one of the management MAC address filters and set the appropriate bit in field 3:0 of the MDEF register.
- Set other bits to qualify which packets are allowed to pass through. For example:
  - Set bit 5 in the MDEF register to qualify with the first manageability VLAN.
  - Set relevant bits 13 to 20 in the MDEF register to qualify with a match to one of the IP addresses.
  - Set any L3/L4 bits (bits 27 to 31 in the MDEF register and bits 16 to 23 in MDEF_EXT) to qualify with any of a set of L3/L4 filters.

38.37.3.7.2 Broadcast Packet Filtering

- Select one of the eight rules for broadcast filtering.
- Set bit 25 in the MDEF register of the decision rule to enforce broadcast filtering.
- Set other bits to qualify which broadcast packets are allowed to pass through. For example:
  - Set bit 5 in the MDEF register to qualify with the first manageability VLAN.
  - Set relevant bits 13 to 20 in the MDEF register to qualify with a match to one of the IP addresses.
  - Set any L3/L4 bits (bits 27 to 31 in the MDEF register and bits 16 to 23 in MDEF_EXT) to qualify with any of a set of L3/L4 filters.

38.37.3.7.3 VLAN Packet Filtering

- Select one of the eight rules for VLAN filtering.
- Set bit 5 to 12 in the MDEF register to qualify with the relevant manageability VLANs.
- Set other bits to qualify which VLAN packets are allowed to pass through. For example:
  - Set any L3/L4 bits (bits 27 to 31 in the MDEF register and bits 16 to 23 in MDEF_EXT) to qualify with any of a set of L3/L4 filters.

38.37.3.7.4 IPv6 Filtering

IPv6 filtering is done using the following IPv6-specific filters:

- IP unicast filtering — requires filtering for link local address and a global address. Filtering setup might depend on whether or not the MAC address is shared with the host or dedicated to manageability:
  - Dedicated MAC address (for example, dynamic address allocation with DHCP does not support multiple IP addresses for one MAC address). In this case, filtering can be done at L2 using two dedicated unicast MAC filters.
  - Shared MAC address (for example, static address allocation sharing addresses with host). In this case, filtering needs to be done at L3, requiring two IPv6 address filters, one per address.
- A neighbor discovery filter — The 10 GbE controller supports IPv6 neighbor discovery protocol. Since the protocol relies on multicast packets, the 10 GbE controller supports filtering of these packets. IPv6 multicast addresses are translated into corresponding Ethernet multicast addresses in the form of 33-33-xx-xx-xx-xx, where the last 32 bits of the address are taken from the last 32 bits of

the IPv6 multicast address. As a result, two direct MAC filters can be used to filter IPv6 solicited-node multicast packets as well as IPv6 all node multicast packets.

38.37.3.7.5 Receive Filtering with Shared IP

When using the legacy SMBus interface or the MCTP interface, it is possible to share the host MAC and IP address with an MC. This functionality is also available when using base NC-SI using Intel OEM commands.

When an MC shares the MAC and IP address with the host, receive filtering is based on identifying specific flows through port allocation. The following setting might be used when using the legacy SMBus interface:

- Select one of the eight rules.
- Set a manageability dedicated MAC filter to the host MAC address and set the matching bit (0-3) in the MDEF register.
- If VLAN is used for management, load one or more management VLAN filters and set the matching bit (5- 12) in the MDEF register.

ARP filter/neighbor discovery filter is enabled when an MC is responsible for handling the ARP protocol. Set bit 27 or bit 28 in the MDEF register for this functionality.

In NC-SI over MCTP, dedicated commands are used to enable shared IP filtering.

38.37.3.8 Determining Manageability MAC Address

If an MC needs to use a dedicated MAC address or configure the automatic ARP response mechanism (only available in SMBus mode), it might be beneficial for an MC to be able to determine the MAC address used by the host.

Both the NC-SI and SMBus interfaces provide an Intel OEM command to read the system MAC address.

A possible use for this is that the MAC address programmed at manufacturing time does not increment by one each time, but rather by two. In this way, an MC can read the system MAC address and add one to it and be guaranteed of a unique MAC address.

Note: Determining the IP address being used by the host is beyond the scope of this document.

38.37.4 OS-to-BMC Traffic

38.37.4.1 Overview

Traditionally, the communication between a host and a local MC is not handled through the network interface and requires a dedicated interface such as an IPMI KCS interface. The 10 GbE controller enables the host and the local MC communication via the regular pass-through interface, and thus enable management of a local console using the same interface used to manage any MC in the network.

When this flow is used, the host sends packets to an MC through the network interface. The 10 GbE controller examines these packets and it then decides if they should be forwarded to an MC. On the inverse path, when an MC sends a packet on the pass-through interface, the 10 GbE controller checks if it should be forwarded to the network, the host, or both. Figure 38-130 shows the flow for OS-to-BMC traffic for the NC-SI over RBT case. OS2BMC is also available when operating over MCTP. It is not available in legacy SMBus mode.

The OS-to-BMC flow can be enabled using the *OS2BMC Enable* field for the relevant port in the OS-to-BMC configuration structure of the NVM.

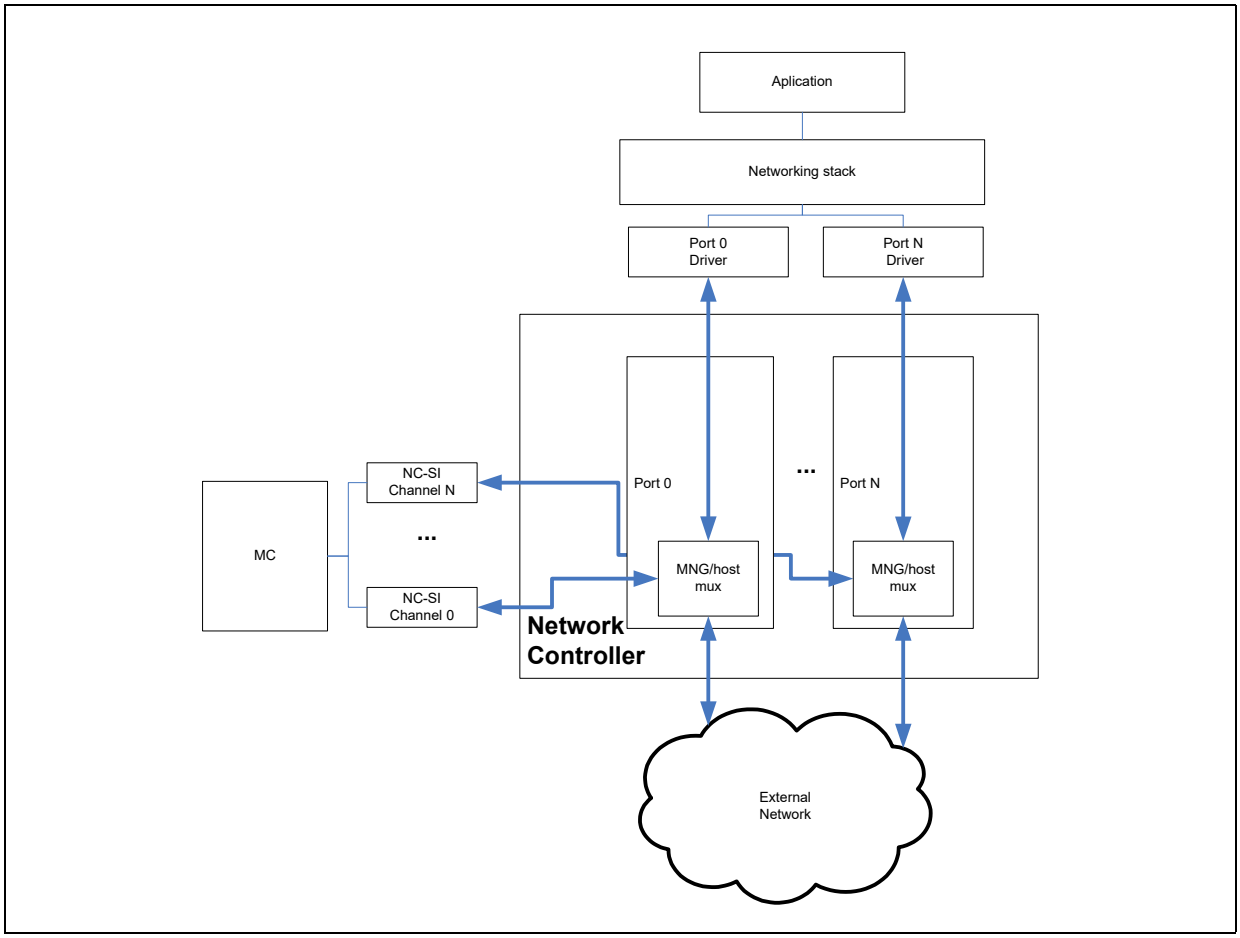


Figure 38-130.OS-to-BMC Flow

The OS-to-BMC flow is enabled only for ports enabled by the NC-SI Enable Channel command or via the *OS2BMC Enable* field for the relevant port in the OS-to-BMC configuration structure of the NVM.

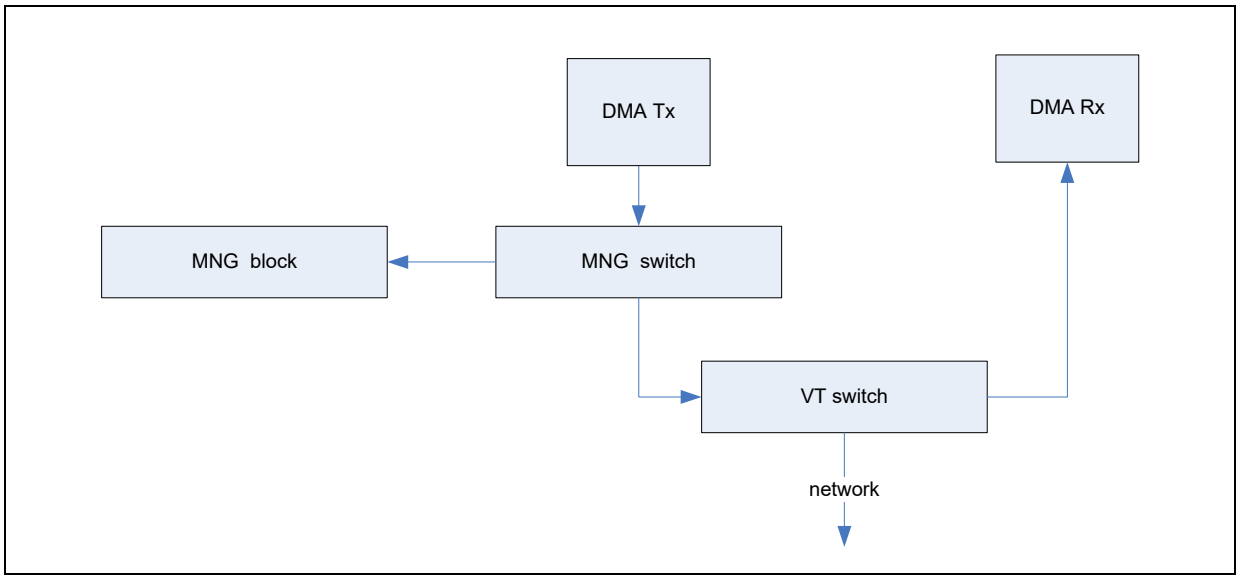
OS-to-BMC traffic must comply with NC-SI specifications and is therefore limited to maximum sized frames of 1536 bytes (in both directions).

38.37.4.2 Filtering

38.37.4.2.1 OS-to-BMC Filtering

Refer to the flow used to filter packets from the MC to the host previously described.

Figure 38-131.OS-to-BMC and VM-to-VM Filtering



Note: Traffic coming from the EMP should not be a candidate for OS2BMC traffic filtering.

Note: In the 10 GbE controller, traffic from the EMP is considered for OS2BMC traffic.

38.37.4.2.2 BMC-to-OS Filtering

Refer to the flow used to filter packets from the host to the BMC previously described.

Note: Traffic sent from the MC does not cause a PME event, even if it matches one of the wake-up filters set by the port.

38.37.4.3 Blocking network-to-BMC Flow

In some systems the MC might have its own private connection to the network and might use a Shortened Product Name port only for the OS-to-BMC traffic. In this case, the BMC-to-network flow should be blocked while enabling the OS-to-BMC and OS-to-network flows.

This can be done by clearing the *PRT_MNG_MANC.EN_BMC2NET* bit for the relevant port. The MC can control this functionality using the Enable Network-to-BMC flow and Disable Network-to-BMC flow NC-SI OEM commands. This can also be controlled using the *Network to BMC disable* field in the NVM OS2BMC Configuration Structure.

Note: The NC-SI channel should not be enabled for receive or transmit before at least one of the *PRT_MNG_MANC.EN_BMC2NET* or *PRT_MNG_MANC.EN_BMC2OS* fields is set, unless only used for AEN transmissions. In this case, the channel might be enabled for receive, but all receive filters should be cleared.

38.37.4.4 OS-to-BMC and Flow Control

The traffic between the host and manageability uses the same buffers as any loopback traffic. Thus, it flows through the transmit buffer and then through the receive buffer. If the transmit buffer is flow controlled, then the host-to-MC traffic is also stopped. If the receive buffer is full, the traffic is dropped or the transmit is stopped according to the flow control policy of this traffic class.

Packets received by manageability (either from the host or from the network) might be dropped if the manageability internal buffers are full.

38.37.4.5 Statistics

Packets sent from the operating system to the MC should be counted by all statistical counters as packets sent by the operating system. If they are sent to both the network and to the MC, then they are counted once.

Packets sent from the MC to the host are counted as packets received by the host. If they are sent to the host and to the network, then they are counted both as received packets and as packet transmitted to the network.

38.37.4.6 OS-to-BMC Enablement

The 10 GbE controller supports the unified network software model for OS-to-BMC traffic, where the OS- to-BMC traffic is shared with the regular traffic. In this model, there is no need for a special configuration of the operating networking stack or the BMC stack, but if the link is down, then the OS-to-BMC communication is stopped.

In order to enable OS-to-BMC either:

- Enable OS2BMC in the port traffic type field in the Traffic Type Parameters NVM word for the relevant port.
- Send an Enable Network-to-BMC command

Note: When OS2BMC is enabled, the operating system must avoid sending packets longer than 1.5 KB to the MC. Such packets are dropped.

38.37.5 SMBus PT Interface

SMBus is the system management bus defined by Intel. It is used in personal computers and servers for low-speed system management communications. This section describes how the SMBus interface operates in legacy PT mode.

38.37.5.1 General

The SMBus sideband interface includes standard SMBus commands used for assigning a slave address and gathering device information as well as Intel proprietary commands used specifically for the pass-through interface.

38.37.5.2 PT Capabilities

This section details manageability capabilities the 10 GbE controller provides while in SMBus mode. PT traffic is carried by the sideband interface as described in Section 38.37.1.

These services are not available in NC-SI mode.

- When operating in SMBus mode, in addition to exposing a communication channel to the LAN for the MC, the 10 GbE controller provides the following manageability services to the MC:
- ARP handling — The 10 GbE controller can be programmed to auto-ARP replying for ARP request packets to reduce the traffic over the MC interconnect.
- Default configuration of filters by NVM — When working in SMBus mode, the default values of the manageability receive filters can be set according to the PT LAN and flex TCO NVM structures.
- Padding of short packets. Packets smaller than 60 bytes but larger than 14 bytes are padded to a legal Ethernet packet.
- CRC calculation — The 10 GbE controller adds an Ethernet CRC on all sent packets.

38.37.5.3 Port-to-SMBus Mapping

The 10 GbE controller is identified on the SMBus manageability link as four different devices (for example, via four different SMBus addresses on which each device is connected to a different LAN port). There is no logical connection between the four devices.

The fail-over between the LAN ports is done by the MC (by sending/receiving packets through different devices). The status report to the MC, ARP handling, DHCP, and other pass-through functionality are unique for each port and configured by the MC.

38.37.5.4 Automatic Ethernet ARP Operation

The 10 GbE controller can offload the Ethernet Address Resolution Protocol (ARP) for the MC in order to reduce the bandwidth required on the SMBus link.

Automatic Ethernet ARP parameters are loaded from the NVM when the 10 GbE controller is powered up or configured through the sideband management interface. The following parameters should be configured in order to enable ARP operation:

- ARP auto-reply enabled
- ARP IP address (to filter ARP packets)
- ARP MAC addresses (for ARP responses)

These are all configurable over the sideband interface using the advanced version of the Receive Enable command.

When an ARP request packet is received and ARP auto-reply is enabled, the 10 GbE controller checks the targeted IP address (after the packet has passed L2 checks and ARP checks). If the targeted IP matches the IP configuration for the 10 GbE controller, it replies with an ARP response.

The 10 GbE controller responds to ARP request targeted to the ARP IP address with the configured ARP MAC address. In case that there is no match, the 10 GbE controller silently discards the packets. If the 10 GbE controller is not configured to do an auto-ARP response, it can be configured to forward the ARP packets to the MC, which can respond to ARP requests.

When the external MC uses the same IP and MAC address of the operating system, the ARP operation should be coordinated with the host operating system.

Note: If sharing the MAC and IP with the host operating system is possible, the 10 GbE controller provides the ability to read the system MAC address, enabling the MC to share the MAC address. However, there is no mechanism provided by the 10 GbE controller to read the IP address. The host operating system (or an agent within) and the MC must coordinate the sharing of IP addresses.

38.37.5.5 SMBus Transactions

This section gives a brief overview of the SMBus protocol. Following is an example for a format of a typical SMBus transaction.

Table 38-526.Typical SMBus Transaction

1	7	1	1	8	1	8	1	1
S	Slave Address	Wr	A	Command	A	PEC	A	P
	1100 001	0	0	0000 0010	0	[Data Dependent]	0	

The top row of the table identifies the bit length of the field in a decimal bit count. The middle row (bordered) identifies the name of the fields used in the transaction. The last row appears only with some transactions, and lists the value expected for the corresponding field. This value can be either hexadecimal or binary.

The SMBus controller is a master for some transactions and a slave for others. The differences are identified in this document.

Shorthand field names are listed in Table 38-527 and are fully defined in the SMBus specification.

Table 38-527.Shorthand Field Names

Field Name	Definition
S	SMBus START Symbol.
P	SMBus STOP Symbol.
PEC	Packet Error Code.
A	ACK (Acknowledge).
N	NACK (Not Acknowledge).
Rd	Read Operation (Read Value = 1b).
Wr	Write Operation (Write Value = 0b).

38.37.5.5.1 SMBus Addressing

The SMBus is presented as up to four SMBus devices on the SMBus (four addresses). All PT functionality is duplicated on the SMBus address, where each SMBus address is connected to a different LAN port. Note that it is not permitted to configure multiple ports to the same SMBus address. When a LAN function is disabled, the corresponding SMBus address is not presented to the MC.

SMBus addresses (enabled from the NVM) can be re-assigned using the SMBus ARP protocol.

In addition to the SMBus address values, all parameters of the SMBus (SMBus channel selection, address mode, and address enable) can be set only through NVM configuration. Note that the NVM is read at the 10 GbE controller's power up and resets.

38.37.5.5.2 SMBus ARP Functionality

The 10 GbE controller supports the SMBus ARP protocol as defined in the SMBus 2.0 specification. The 10 GbE controller is a persistent slave address device so its SMBus address is valid after power-up and loaded from the NVM. The 10 GbE controller supports all SMBus ARP commands defined in the SMBus specification both general and directed.

SMBus ARP capability can be disabled through the NVM.

38.37.5.5.3 SMBus ARP Flow

SMBus ARP flow is based on the status of two flags:

- Address Valid (AV): This flag is set when the 10 GbE controller has a valid SMBus address.
- Address Resolved (AR): This flag is set when the 10 GbE controller SMBus address is resolved (SMBus address was assigned by the SMBus ARP process).

These flags are internal Product Name flags and are not exposed to external SMBus devices.

Since the 10 GbE controller is a Persistent SMBus Address (PSA) device, the AV flag is always set, while the AR flag is cleared after power up until the SMBus ARP process completes. Since AV is always set, the 10 GbE controller always has a valid SMBus address.

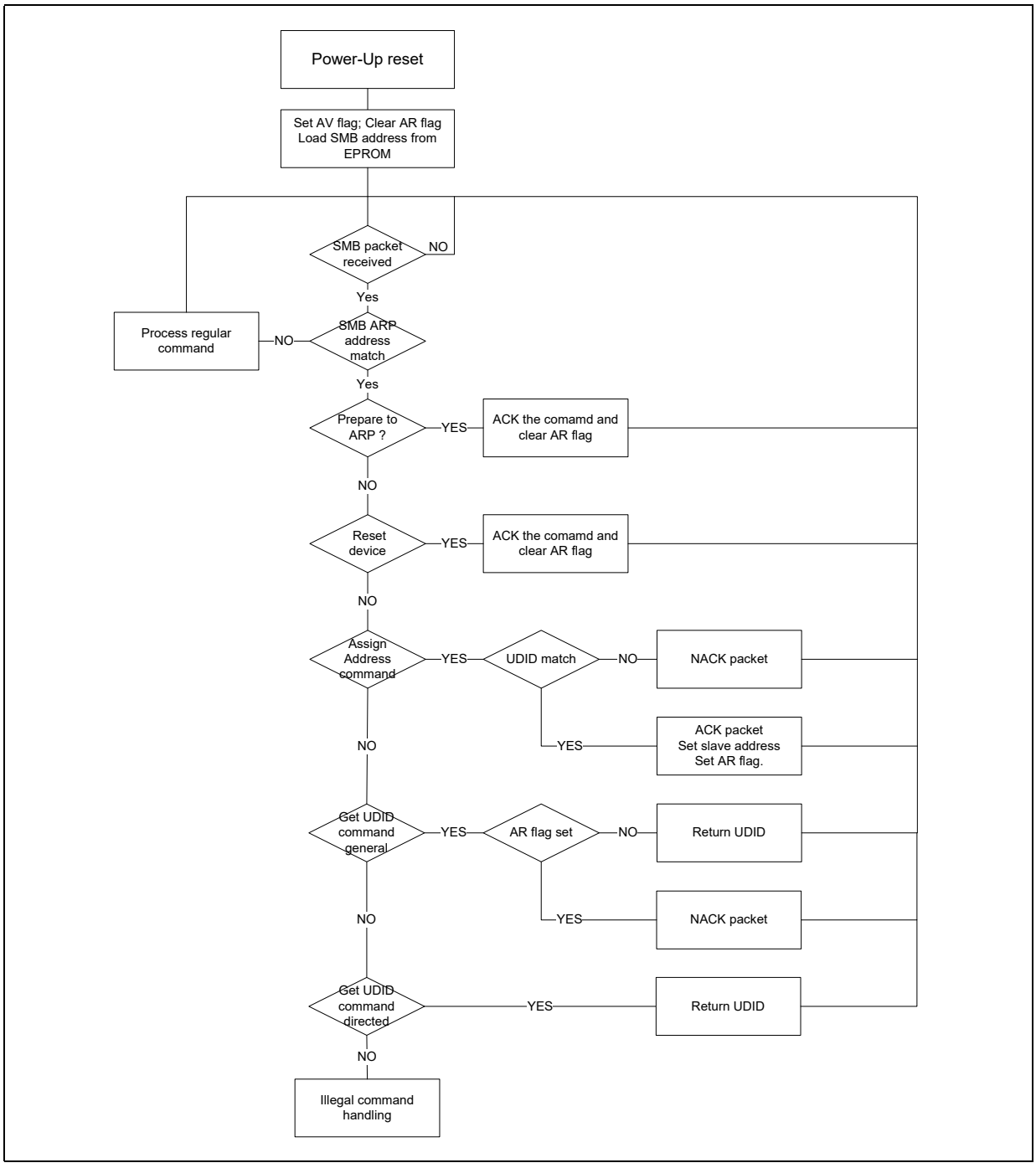
When the SMBus master needs to start an SMBus ARP process, it resets (in terms of ARP functionality) all devices on SMBus by issuing either Prepare to ARP or Reset Device commands. When the 10 GbE controller accepts one of these commands, it clears its AR flag (if set from previous SMBus ARP process), but not its AV flag (the current SMBus address remains valid until the end of the SMBus ARP process).

Clearing the AR flag means that the 10 GbE controller responds to SMBus ARP transactions that are issued by the master. The SMBus master issues a Get UDID command (general or directed) to identify the devices on the SMBus. The 10 GbE controller always responds to the Directed command and to the General command only if its AR flag is not set.

After the Get UDID, The master assigns the 10 GbE controller SMBus address by issuing an Assign Address command. The 10 GbE controller checks whether the UDID matches its own UDID and if it matches, it switches its SMBus address to the address assigned by the command (byte 17). After accepting the Assign Address command, the AR flag is set and from this point (as long as the AR flag is set), the 10 GbE controller does not respond to the Get UDID General command. Note that all other commands are processed even if the AR flag is set. The 10 GbE controller stores the SMBus address that was assigned in the SMBus ARP process in the NVM, so at the next power up, it returns to its assigned SMBus address.

Figure 38-132 shows the 10 GbE controller SMBus ARP flow.

Figure 38-132.SMBus ARP Flow



38.37.5.5.4 SMBus ARP UDID Content

The UDID provides a mechanism to isolate each device for the purpose of address assignment. Each device has a unique identifier. The 128-bit number is comprised of the following fields:

Table 38-528.UDID

1 Byte	1 Byte	2 Bytes	2 Bytes	2 Bytes	2 Bytes	2 Bytes	4 Bytes
Device Capabilities	Version/ Revision	Vendor ID	Device ID	Interface	Subsystem Vendor ID	Subsystem Device ID	Vendor Specific ID
See notes that follow	See notes that follow	0x8086	0x3A84	0x0004/ 0x0024	0x0000	0x0000	See notes that follow
MSB							LSB

Where:

Vendor ID:	The device manufacturer's ID as assigned by the SBS Implementers' Forum or the PCI SIG. Constant value: 0x8086.
Device ID:	The device ID as assigned by the device manufacturer (identified by the Vendor ID field). Constant value: 0x3A84.
Interface:	Identifies the protocol layer interfaces supported over the SMBus connection by the device. Bits 3:0 = 0x4 indicates SMBus Version 2.0. Bit 5 (ASF bit) = 1 in MCTP mode.
Subsystem Fields:	These fields are not supported and return zeros.

Device Capabilities: Dynamic and Persistent Address, *PEC Support* bit:

7	6	5	4	3	2	1	0
Address Type		Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	PEC Supported
0b	1b	0b	0b	0b	0b	0b	0/1b[1]
MSB							LSB

Notes:

1. The value is set according to the SMBus transaction PEC bit in the NVM.

Version/Revision: UDID Version 1, Silicon Revision:

7	6	5	4	3	2	1	0
Reserved (0)	Reserved (0)	UDID Version			Silicon Revision ID		
0b	0b	001b			See the following table		
MSB							LSB

Silicon Revision ID:

Silicon Version	Revision ID
A0	000b
B0	001b
B1	010b

Vendor Specific ID: Four LSB bytes of the device Ethernet MAC address of the relevant port. The port Ethernet address is taken from the *PRTGL_SAL* registers of the relevant ports. Note that in the 10 GbE controller there are four MAC addresses (one for each port).

1 Byte	1 Byte	1 Byte	1 Byte
MAC Address, Byte 3	MAC Address, Byte 2	MAC Address, Byte 1	MAC Address, Byte 0
MSB			LSB

38.37.5.5.5 SMBus ARP and Multi-Port

The 10 GbE controller responds as four SMBus devices having four sets of AR/AV flags (one for each port). The 10 GbE controller responds four time to the SMBus ARP master, once each for each port. All SMBus addresses are taken from the SMBus ARP address word of the NVM.

Note that the Unique Device Identifier (UDID) is different for the four ports in the version ID field, which represents the MAC address and is different for the four ports. The 10 GbE controller first respond as port 0, and only when an address is assigned, then start responding as port 1, 2 and 3 to the Get UDID command.

38.37.5.5.6 Concurrent SMBus Transactions

The SMBus interface is single threaded. Thus, concurrent SMBus transactions are not permitted. Once a transaction starts, it must complete before an additional transaction is initiated.

A transaction is defined as:

- All the SMBus commands used to receive a packet.
- All the SMBus commands used to send a packet.
- Refer to the read and write SMBus commands used as part of read parameters previously described.
- Refer to the single write SMBus commands previously described.

38.37.5.6 SMBus Notification Methods

The 10 GbE controller supports three methods of notifying the MC that it has information that needs to be read by the MC:

- SMBus alert — Refer to Section 38.37.5.6.1.
- Asynchronous notify.
- Direct receive — refer to Section 38.37.5.6.3.

The notification method used by the 10 GbE controller can be configured from the SMBus using the Receive Enable command. The default method is set by the NVM in the *Notification Method* field in LAN Receive Enable 1.

Note: The SMBus notification method used must be the same for all ports.

The following events cause the 10 GbE controller to send a notification event to the MC:

- Receiving a LAN packet that is designated to the MC.
- Firmware was reset and requires re-initialization.
- Receiving a Request Status command from the MC initiates a status response.

- The 10 GbE controller is configured to notify the MC upon status changes (by setting the EN_STA bit in the Receive Enable command) and one of the following events happen:
  - TCO Command aborted.
  - Link status changed.
  - Power state change.

There can be cases where the MC is hung and not responding to the SMBus notification. The 10 GbE controller has a time-out value (defined in the NVM) to avoid hanging while waiting for the notification response. If the MC does not respond until the time out expires, the notification is de-asserted and all pending data is silently discarded.

Note that the SMBus notification time-out value can only be set in the NVM. The MC cannot modify this value.

38.37.5.6.1 SMBus Alert and Alert Response Method

The SMBus Alert# (SMBALERT_N) signal is an additional SMBus signal that acts as an asynchronous interrupt signal to an external SMBus master. The 10 GbE controller asserts this signal each time it has a message that it needs the MC to read and if the chosen notification method is the SMBus alert method. Note that the SMBus alert method is an open-drain signal which means that other devices besides the 10 GbE controller can be connected on the same alert pin. As a result, the MC needs a mechanism to distinguish between the alert sources.

The MC can respond to the alert by issuing an ARA Cycle command to detect the alert source device. The 10 GbE controller responds to the ARA cycle with its own SMBus slave address (if it was the SMBus alert source) and de-asserts the alert when the ARA cycle is completes. Following the ARA cycle, the MC issues a read command to retrieve the 10 GbE controller message.

Some MCs do not implement the ARA cycle transaction. These MCs respond to an alert by issuing a Read command to the 10 GbE controller (0xC0/0xD0 or 0xDE). The 10 GbE controller always responds to a Read command, even if it is not the source of the notification. The default response is a status transaction. If the 10 GbE controller is the source of the SMBus Alert, it replies the read transaction and then de-asserts the alert after the command byte of the read transaction.

Note: In SMBus Alert mode, the SMBALERT_N pin is used for notification. Each port generate alerts on events that are independent of each other.

Note: If two ports have events to notify, the second alert is asserted only after the first event is handled.

The ARA cycle is an SMBus receive byte transaction to SMBus Address 0001b - 100b. Note that the ARA transaction does not support PEC. The ARA transaction format is as follows:

1	7	1	1	8	1	1	1
S	Alert Response Address	Rd	A	Slave Device Address		A	P
	0001 100	1	0	Manageability Slave SMBus Address	0	1	

Note: If the MC does not react to the alert in the delay defined by the *SMBus Notification Timeout* NVM field, the ALERT pin is de-asserted and the Rx packet indication in the status word is cleared (and packet is dropped).

38.37.5.6.2 Asynchronous notify method

When configured using the asynchronous notify method, the 10 GbE controller acts as a SMBus master and notifies the BMC of one of the events listed in Section 38.37.5.6 by issuing a modified form of the write word transaction. The asynchronous notify transaction SMBus address and data payload is configured using the Section "Receive Enable Command" or using the NVM defaults. Note that the asynchronous notify is not protected by a PEC byte.

1	7	1	1	7	1	1	8	1	8	1	1
S	Target Address	Wr	A	Sending Device Address		A	Data Byte Low	A	Data Byte High	A	P
	MC Slave Address	0	0	MNG Slave SMBus Address	0	0	Interface	0	Alert Value	0	

The target address and data byte low/high are taken from the Section "Receive Enable Command" or NVM configuration.

If the MC does not read the status in the delay defined by the *SMBus Notification Timeout* NVM field, the Rx packet indication in the status word is cleared (and packet is dropped).

38.37.5.6.3 Direct Receive Method

If configured, the 10 GbE controller has the capability to send a message it needs to transfer to the external MC as a master over the SMBus instead of alerting the MC and waiting for it to read the message.

The message format follows. Note that the command that is used is the same command that is used by the external MC in the Block Read command. The opcode that the 10 GbE controller puts in the data is also the same as it put in the Block Read command of the same functionality. The rules for the *F* and *L* flags (bits) are also the same as in the Block Read command.

1	7	1	1	1	1	6	1	
S	Target Address	Wr	A	F	L	Command	A	...
	BMC Slave Address	0	0	First Flag	Last Flag	Receive TCO Command 01 0000b	0	

8	1	8	1		1	8	1	1
Byte Count	A	Data Byte 1	A	...	A	Data Byte N	A	P
N	0		0		0		0	

38.37.5.7 Receive PT flow

The 10 GbE controller is used as a channel for receiving packets from the network link and passing them to the external MC. The MC configures the 10 GbE controller to pass these specific packets to the MC. Once a full packet is received from the link and identified as a manageability packet that should be transferred to the MC, the 10 GbE controller starts the receive TCO flow to the MC.

The 10 GbE controller uses the SMBus notification method to notify the MC that it has data to deliver. Since the packet size might be larger than the maximum SMBus fragment size, the packet is divided into fragments, where the 10 GbE controller uses the maximum fragment size allowed in each fragment (configured via the NVM). The last fragment of the packet transfer is always the status of the packet. As a result, the packet is transferred in at least two fragments. The data of the packet is transferred as part of the receive TCO LAN packet transaction.

When SMBus alert is selected as the MC notification method, the 10 GbE controller notifies the MC on each fragment of a multi-fragment packet. When asynchronous notify is selected as the MC notification method, the 10 GbE controller notifies the MC only on the first fragment of a received packet. It is the MC's responsibility to read the full packet including all the fragments.

Any timeout on the SMBus notification results in discarding the entire packet. Any NACK by the MC causes the fragment to be re-transmitted to the MC on the next Receive Packet command.

The maximum size of the received packet is limited by the 10 GbE controller to 1536 bytes. Packets larger than 1536 bytes are silently discarded. Any packet smaller than 1536 bytes is processed.

38.37.5.8 Transmit PT Flow

The 10 GbE controller is used as the channel for transmitting packets from the external MC to the network link. The network packet is transferred from the MC over the SMBus and then, when fully received by the 10 GbE controller, is transmitted over the network link.

Each SMBus address is connected to a different LAN port. When a packet is received during a SMBus transaction using SMBus address #0, it is transmitted to the network using LAN port #0; it is transmitted through LAN port #1 if received on SMBus address #1, etc.

The 10 GbE controller supports packets up to an Ethernet packet length of 1536 bytes. Since SMBus transactions can only be up to 240 bytes in length, packets might need to be transferred over the SMBus in more than one fragment. This is achieved using the *F* and *L* bits in the command number of the transmit TCO packet Block Write command. When the *F* bit is set, it is the first fragment of the packet. When the *L* bit is set, it is the last fragment of the packet. When both bits are set, the entire packet is in one fragment. The packet is sent over the network link only after all its fragments are received correctly over the SMBus. The maximum SMBus fragment size is defined within the NVM and cannot be changed by the BMC.

The minimum packet length defined by the 802.3 specification is 64 bytes. The 10 GbE controller pads packets that are less than 64 bytes to meet the specification requirements (there is no need for the external MC to pad packets less than 64 bytes). If the packet sent by the MC is larger than 1536 bytes, the 10 GbE controller silently discards the packet. The minimal packet size that the 10 GbE controller can handle is 14 bytes.

The 10 GbE controller calculates the L2 CRC on the transmitted packet and adds its four bytes at the end of the packet. Any other packet field (such as XSUM or VLAN) must be calculated and inserted by the MC (the 10 GbE controller does not change any field in the transmitted packet, other than adding padding and CRC bytes).

If the network link is down when the 10 GbE controller has received the last fragment of the packet from the MC, it silently discards the packet. Note that any link down event during the transfer of any packet over the SMBus does not stop the operation since the 10 GbE controller waits for the last fragment to end to see whether the network link is up again.

38.37.5.8.1 Transmit Errors in Sequence Handling

Once a packet is transferred over the SMBus from the MC to the 10 GbE controller, the *F* and *L* flags should follow specific rules. The *F* flag defines the first fragment of the packet; the *L* flag that the transaction contains the last fragment of the packet. Table 38-529 lists the different flag options in transmit packet transactions.

Table 38-529.Flag Options During Transmit Packet Transactions

Previous	Current	Action/Notes
Last	First	Accept both.
Last	Not First	Error for the current transaction. Current transaction is discarded and an abort status is asserted.
Not Last	First	Error in previous transaction. Previous transaction (until previous First) is discarded. Current packet is processed. No abort status is asserted.
Not Last	Not First	Process the current transaction.

Note: Since every other Block Write command in TCO protocol has both *F* and *L* flags set, they cause flushing any pending transmit fragments that were previously received. When running the TCO transmit flow, no other Block Write transactions are allowed in between the fragments.

38.37.5.8.2 TCO Command Aborted Flow

The 10 GbE controller indicates to the MC an error or an abort condition by setting the *TCO Abort* bit in the general status. The 10 GbE controller might also be configured to send a notification to the MC.

Following is a list of possible error and abort conditions:

- Any error in the SMBus protocol (NACK, SMBus timeouts, etc.).
- If the MC does not respond until the notification timeout (programmed in the NVM) expires.
- Any error in compatibility between required protocols to specific functionality (for example, Rx Enable command with a byte count not equal to 1/14, as defined in the command specification).
- If the 10 GbE controller does not have space to store the transmitted packet from the MC (in its internal buffer space) before sending it to the link, the packet is discarded and the external MC is notified via the *Abort* bit.
- Error in the *F*/*L* bit sequence during multi-fragment transactions.
- An internal reset to the 10 GbE controller's firmware.

38.37.5.9 SMBus Link State Control

While in SMBus mode, the default setting of the link is defined by the *EMP_LINK_ON* bit in Common Firmware Parameters 2 NVM word.

When a channel is enabled through NVM setting or through the RCV_EN option of the Receive Enable command, the link is established (if not already required for other purposes).

If the channel is disabled by clearing of the RCV_EN option, then the link might move back to the default defined by the EMP_LINK_ON if not needed for other purposes.

Note: Before a transition to D3 it is the responsibility of the software device driver to request the PHY to be active for wake-up activities.

38.37.5.10 SMBus ARP Transactions

All SMBus ARP transactions include the PEC byte.

38.37.5.11 Prepare to ARP

This command clears the *Address Resolved* flag (set to false). It does not affect the status or validity of the dynamic SMBus address and is used to inform all devices that the ARP master is starting the ARP process:

1	7	1	1	8	1	8	1	1
S	Slave Address	Wr	A	Command	A	PEC	A	P
	1100 001	0	0	0000 0001	0	[Data Dependent Value]	0	

38.37.5.11.1 Reset device (general)

This command clears the *Address Resolved* flag (set to false). It does not affect the status or validity of the dynamic SMBus address.

1	7	1	1	8	1	8	1	1
S	Slave Address	Wr	A	Command	A	PEC	A	P
	1100 001	0	0	0000 0010	0	[Data Dependent Value]	0	

38.37.5.11.2 Reset Device (Directed)

The Command field is NACKed if bits 7:1 do not match the current SMBus address. This command clears the *Address Resolved* flag (set to false) and does not affect the status or validity of the dynamic SMBus address.

1	7	1	1	8	1	8	1	1
S	Slave Address	Wr	A	Command	A	PEC	A	P
	1100 001	0	0	Targeted Slave Address | 0	0	[Data Dependent Value]	0	

38.37.5.11.3 Assign Address

This command assigns SMBus address. The address and command bytes are always acknowledged.

The transaction is aborted (NACKed) immediately if any of the UDID bytes is different from Product Name UDID bytes. If successful, the manageability system internally updates the SMBus address. This command also sets the *Address Resolved* flag (set to true).

1	7	1	1	8	1	8	1	
S	Slave Address	Wr	A	Command	A	Byte Count	A	· · ·
	1100 001	0	0	0000 0100	0	0001 0001	0	

8	1	8	1	8	1	8	1	
Data 1	A	Data 2	A	Data 3	A	Data 4	A	· · ·
UDID Byte 15 (MSB)	0	UDID Byte 14	0	UDID Byte 13	0	UDID Byte 12	0	

8	1	8	1	8	1	8	1	
Data 5	A	Data 6	A	Data 7	A	Data 8	A	· · ·
UDID Byte 11	0	UDID Byte 10	0	UDID Byte 9	0	UDID Byte 8	0	

8	1	8	1	8	1	
Data 9	A	Data 10	A	Data 11	A	· · ·
UDID Byte 7	0	UDID Byte 6	0	UDID Byte 5	0	

8	1	8	1	8	1	8	1	
Data 12	A	Data 13	A	Data 14	A	Data 15	A	· · ·
UDID Byte 4	0	UDID Byte 3	0	UDID Byte 2	0	UDID Byte 1	0	

8	1	8	1	8	1	1
Data 16	A	Data 17	A	PEC	A	P
UDID Byte 0 (LSB)	0	Assigned Address	0	[Data Dependent Value]	0	

38.37.5.11.4 Get UDID (General and Directed)

The general get UDID SMBus transaction supports a constant command value of 0x03 and, if directed, supports a Dynamic command value equal to the dynamic SMBus address.

If the SMBus address has been resolved (*Address Resolved* flag set to true), the manageability system does not acknowledge (NACK) this transaction. If it's a General command, the manageability system always acknowledges (ACKs) as a directed transaction.

This command does not affect the status or validity of the dynamic SMBus address or the *Address Resolved* flag.

S	Slave Address	Wr	A	Command	A	S	· · ·
	1100 001	0	0	See Below	0		

7	1	1	8	1	
Slave Address	Rd	A	Byte Count	A	· · ·

7	1	1	8	1	
1100 001	1	0	0001 0001	0	

8	1	8	1	8	1	8	1	
Data 1	A	Data 2	A	Data 3	A	Data 4	A	· · ·
UDID Byte 15 (MSB)	0	UDID Byte 14	0	UDID Byte 13	0	UDID Byte 12	0	

8	1	8	1	8	1	8	1	
Data 5	A	Data 6	A	Data 7	A	Data 8	A	· · ·
UDID Byte 11	0	UDID Byte 10	0	UDID Byte 9	0	UDID Byte 8	0	

8	1	8	1	8	1	
Data 9	A	Data 10	A	Data 11	A	· · ·
UDID Byte 7	0	UDID Byte 6	0	UDID Byte 5	0	

8	1	8	1	8	1	8	1	
Data 12	A	Data 13	A	Data 14	A	Data 15	A	· · ·
UDID Byte 4	0	UDID Byte 3	0	UDID Byte 2	0	UDID Byte 1	0	

8	1	8	1	8	1	1
Data 16	A	Data 17	A	PEC	~Ã	P
UDID Byte 0 (LSB)	0	Device Slave Address	0	[Data Dependent Value]	1	

The Get UDID command depends on whether or not this is a Directed or General command.

The General Get UDID SMBus transaction supports a constant command value of 0x03.

The Directed Get UDID SMBus transaction supports a Dynamic command value equal to the dynamic SMBus address with the LSB bit set.

Note: Bit 0 (LSB) of Data byte 17 is always 1b.

38.37.5.12 SMBus PT Transactions

This section details commands (both read and write) that the 10 GbE controller SMBus interface supports for PT.

38.37.5.12.1 Write SMBus Transactions

This section details the commands that the MC can send to the 10 GbE controller over the SMBus interface. The SMBus write transactions table lists the different SMBus write transactions supported by the 10 GbE controller.

TCO Command	Transaction	Command	Fragmentation
Transmit Packet	Block Write	First: 0x84 Middle: 0x04 Last: 0x44	Multiple
Transmit Packet	Block Write	Single: 0xC4	Single
Request Status	Block Write	Single: 0xDD	Single
Receive Enable	Block Write	Single: 0xCA	Single
Force TCO	Block Write	Single: 0xCF	Single
Management Control	Block Write	Single: 0xC1	Single
Update MNG RCV Filter Parameters	Block Write	Single: 0xCC	Single
Set Common Filters	Block Write	Single: 0xC2	Single
Clear All Filters	Block Write	Single: 0xC3	Single

Transmit Packet Command

The Transmit Packet command behavior is detailed in Section 38.37.5.8. The transmit packet fragments have the following format.

The payload length is limited to the maximum payload length set in the NVM. If the overall packet length is bigger than 1536 bytes, the packet is silently discarded.

Function	Command	Byte Count	Data 1	...	Data N
Transmit first fragment	0x84	N	Packet data MSB	...	Packet data LSB
Transmit middle fragment	0x04				
Transmit last fragment	0x44				
Transmit single fragment	0xC4				

Request Status Command

An external MC can initiate a request to read Product Name manageability status by sending a Request Status command. When received, the 10 GbE controller initiates a notification to an external MC when status is ready. After this, the external controller is able to read the status, by issuing a Read Status command (see Section 38.37.5.12.3).

The format is as follows:

Function	Command	Byte Count	Data 1
Request Status	0xDD	1	0

Receive Enable Command

The Receive Enable command is a single fragment command used to configure the 10 GbE controller. This command has two formats: short, 1-byte legacy format (providing backward compatibility with previous components) and long, 14-byte advanced format (allowing greater configuration capabilities). The Receive Enable command format is as follows:

Function	CMD	Byte Count	Data 1	Data 2	...	Data 7	Data 8	...	Data 11	Data 12	Data 13	Data 14
Legacy Receive Enable	0xCA	1	Receive Control Byte	-	...	-	-	...	-	-	-	-
Advanced Receive Enable		14 (0x0E)		MAC Addr MSB		MAC Addr LSB	IP Addr MSB		IP Addr LSB	BMC SMBus Addr	I/F Data Byte	Alert Value Byte

Field	Bit(s)	Description
RCV_EN	0	Receive TCO Enable. 0b = Disable receive TCO packets. 1b = Enable Receive TCO packets. Setting this bit enables all manageability receive filtering operations. Enabling specific filters is done via the NVM or through special configuration commands. ***Note:*** When the *RCV_EN* bit is cleared, all receive TCO functionality is disabled, not just the packets that are directed to the MC (also auto ARP packets).
RCV_ALL	1	Receive All Enable. 0b = Disable receiving all packets. 1b = Enable receiving all packets. Forwards all packets received over the wire that passed L2 filtering to the external MC. This flag has no effect if bit 0 (Enable TCO packets) is disabled.
EN_STA	2	Enable Status Reporting. 0b = Disable status reporting. 1b = Enable status reporting.
EN_ARP_RES	3	Enable ARP Response. 0b = Disable the 10 GbE controller ARP response. The 10 GbE controller treats ARP packets as any other packet, for example, packet is forwarded to the MC if it passed other (non-ARP) filtering. 1b = Enable the 10 GbE controller ARP response. The 10 GbE controller automatically responds to all received ARP requests that match the IP address programmed by the MC. The MC IP address is provided as part of the Receive Enable message (bytes 8:11). If a short version of the command is used, the 10 GbE controller uses IP address configured in the most recent long version of the command in which the EN_ARP_RES bit was set. If no such previous long command exists, then the 10 GbE controller uses the IP address configured in the NVM as ARP Response IPv4 address in the PT LAN configuration structure. If the *CBDM* bit is set, the 10 GbE controller uses the BMC dedicated MAC address in ARP response packets. If the *CBDM* bit is not set, the MC uses the host MAC address. When the enable ARP response feature is activated, the 10 GbE controller uses the following registers to filter in ARP requests. MC should not modify these registers: Manageability Decision Filter – MDEF7 (and corresponding bit 7 in Management Only traffic register – MNGONLY). Fourth IPv4 Filter.

Field	Bit(s)	Description
NM	5:4	Notification Method. Define the notification method the 10 GbE controller uses. 00b = SMBUS alert. 01b = Asynchronous notify. 10b = Direct receive. 11b = Not supported. Changing the notification method in any port updates the notification method of all ports.
Reserved	6	Reserved. Must be set to 1b.
CBDM	7	Configure the MC Dedicated MAC Address. ***Note:*** This bit should be 0b when the *RCV_EN* bit (bit 0) is not set. 0b = The 10 GbE controller shares the MAC address for MNG traffic with the host MAC address, which is specified in NVM words 0x0-0x2. The MAC filtering is not enforced. See the note at the bottom of this table. 1b = The 10 GbE controller uses the MC dedicated MAC address as a filter for incoming receive packets. The MC MAC address is set in bytes 2-7 in this command. If a short version of the command is used, the 10 GbE controller uses the MAC address configured in the most recent long version of the command in which the *CBDM* bit was set. When the dedicated MAC address feature is activated, the 10 GbE controller uses the following registers to filter in all the traffic addressed to the BMC MAC. BMC can not modify these registers: Manageability Decision Filter – MDEF7 (and corresponding bit 7 in Management Only traffic register – PRT_MNG_MNGONLY). Manageability Decision Filter – MDEF6 (and corresponding bit 6 in Management Only traffic register – MNGONLY). Manageability MAC Address Low – PRT_MNG_MMAL[3]. Manageability MAC Address High – PRT_MNG_MMAH[3]. When the dedicated MAC address feature is cleared, these registers are not programmed and the BMC may use other filters to enforce MAC filtering using the Update Management Receive Filter Parameters command.

Management MAC Address (Data Bytes 7:2)

Ignored if the *CBDM* bit is not set. This MAC address is used to configure the dedicated MAC address. In addition, it is used in the ARP response packet when the *EN_ARP_RES* bit is set. This MAC address is also used when *CBDM* bit is set in subsequent short versions of this command.

Management IP Address (Data Bytes 11:8)

This IP address is used to filter ARP request packets.

Asynchronous Notification SMBus Address (Data Byte 12)

This address is used for the asynchronous notification SMBus transaction and for direct receive. The SMBus address is stored in bit 7:1 of this byte. Bit 0 is always 0b.

Interface Data (Data Byte 13)

Interface data byte used in asynchronous notification.

Alert Value Data (Data Byte 14)

Alert value data byte used in asynchronous notification.

Force TCO Command

This command causes the 10 GbE controller to perform a TCO reset, TCO isolate, or firmware reset

TCO reset — If force TCO reset is enabled in the NVM, the force TCO reset clears the data path (Rx/Tx) of the 10 GbE controller to enable the MC to transmit/receive packets through the 10 GbE controller by asserting a global reset. This command should only be used when the MC is unable to transmit receive and suspects that the 10 GbE controller is inoperable. The command also causes the LAN software device driver to unload. It is recommended to perform a system restart to resume normal operation.

TCO isolate — if TCO isolate is enabled in the NVM, the TCO Isolate command disables PCIe write operations to the LAN port. If TCO isolate is disabled in NVM, the 10 GbE controller does not execute the command but sends a response to the MC with successful completion. Following a TCO isolate, management sets *EMP_TCO_ISOLATE.EMP_TCO_ISOLATE* to 1b for all the PFs associated with the port on which this command is received.

Firmware reset — This command causes re-initialization of all the embedded controller functions and re-load of related NVM words (like a firmware patch code). Applying this command resets the entire device as well as having an effect on TCO reset. A firmware reset is achieved by setting the *GSCR.SET_FWRST* aux bit.

Note: A firmware reset causes a global reset of the entire device (GLOBR).

The 10 GbE controller considers the Force TCO Reset command as an indication that the operating system is unavailable. The Force TCO command format is as follows:

Function	Command	Byte Count	Data 1
Force TCO Reset	0xCF	1	TCO Mode

Where TCO mode is:

Field	Bit(s)	Description
DO_TCO_RST	0	Perform TCO Reset. 0b = Do nothing. 1b = Perform TCO reset.
DO_TCO_ISOLATE[1]	1	Do TCO Isolate. 0b = Enable PCIe write access to LAN port. 1b = Isolate host PCIe write operation to the port. Should be used for debug only.
RESET_MGMT	2	Reset Manageability; re-load manageability NVM words. 0b = Do nothing. 1b = Issue firmware reset to manageability. Setting this bit generates a one-time firmware reset. Following the reset, management related data from NVM is loaded.
Reserved	7:3	Reserved (set to 0x00).

Notes:

1. TCO isolate host write operation enabled in NVM.

Note: Only one of the fields should be set in a given command. Setting more than one field might yield unexpected results.

Management Control

This command is used to set generic manageability parameters. The parameters are listed in Table 38-530. The command is 0xC1 stating that it is a Management Control command. The first data byte is the parameter number and the data afterwards (length and content) are parameter specific as listed in Table 38-530.

Note: If the parameter that the MC sets is not supported by the 10 GbE controller. The 10 GbE controller does not NACK the transaction. After the transaction ends, the 10 GbE controller discards the data and asserts a transaction abort status.

The Management Control command format is as follows:

Function	Command	Byte Count	Data 1	Data 2	...	Data N
Management Control	0xC1	N	Parameter Number	Parameter Dependent		

Table 38-530.Management control command parameters/content

Parameter	#	Parameter Data
Keep PHY Link Up	0x00	A single byte parameter: Data 2: Bit 0 = Set to indicate that the PHY link for this port should be kept up throughout system resets. This is useful when the server is reset and the MC needs to keep connectivity for a manageability session. Bit [7:1] = Reserved. 0b = Disabled. 1b = Enabled.

Update Management Receive Filter Parameters

This command is used to set the manageability receive filters parameters. The command is 0xCC. The first data byte is the parameter number and the data that follows (length and content) are parameter specific as listed in management RCV filter parameters.

If the parameter that the MC sets is not supported by the 10 GbE controller, then the 10 GbE controller does not NACK the transaction. After the transaction ends, the 10 GbE controller discards the data and asserts a transaction abort status.

The update management RCV receive filter parameters command format is as follows:

Function	Command	Byte Count	Data 1	Data 2	...	Data N
Update Manageability Filter Parameters	0xCC	N	Parameter Number	Parameter Dependent		

Table 38-531 lists the different parameters and their content.

Table 38-531.Management Receive Filter Parameters

Parameter	Number	Parameter Data
Filters Enables	0x1	Defines the generic filters configuration. The structure of this parameter is four bytes as the Manageability Control (PRT_MNG_MANC) register. The general filter enable is in the Receive Enable command that enables receive filtering.
MNGONLY Configuration	0xF	This parameter defines which of the packets types identified as manageability packets in the receive path will never be directed to the host memory. Data 2:5 = PRT_MNG_MNGONLY register bytes - Data 2 is the MSB.
Flex Filter 0 Enable Mask and Length	0x10	Flex Filter 0 Mask. Data 17:2 = Mask. Bit 0 in data 2 is the first bit of the mask. Data 19:18 = Reserved. Should be set to 00b. Date 20 = Flexible filter length.

Table 38-531.Management Receive Filter Parameters

Parameter	Number	Parameter Data
Flex Filter 0 Data	0x11	Data 2 — Group of flex filter's bytes: 0x0 = bytes 0-29. 0x1 = bytes 30-59. 0x2 = bytes 60-89. 0x3 = bytes 90-119. 0x4 = bytes 120-127. Data 3:32 = Flex filter data bytes. Data 3 is LSB. Group's length is not a mandatory 30 bytes; it might vary according to filter's length and must NOT be padded by zeros.
Decision Filters	0x61	This command is obsolete and should not be used. Use 0x68 instead.
VLAN Filters	0x62	Three bytes are required to load the VLAN tag filters. Data 2: VLAN filter number. Data 3: MSB of VLAN filter. Data 4: LSB of VLAN filter.
Flex Port Filters	0x63	Three to four bytes are required to load the manageability flex port filters. Data 2 = Flex port filter number. Data 3 = MSB of flex port filter. Data 4 = LSB of flex port filter. Data 5: Bit 0 = Match UDP ports. Bit 1 = Match TCP ports. Bit 2 = Match destination port (0) or source port (1). If Data 5 is not present, the match is done on TCP and UDP destination ports (legacy behavior).
IPv4 Filters	0x64	Five bytes are required to load the IPv4 address filter. Data 2 = IPv4 address filter number (3:0). Data 3 = LSB of IPv4 address filter. … Data 6 = MSB of IPv4 address filter.
IPv6 Filters	0x65	17 bytes are required to load the IPv6 address filter. Data 2 = IPv6 address filter number (3:0). Data 3 = LSB of IPv6 address filter. … Data 18 = MSB of IPv6 address filter.
MAC Filters	0x66	Seven bytes are required to load the MAC address filters. Data 2 = MAC address filters pair number (3:0). Data 3 = MSB of MAC address. … Data 8 = LSB of MAC address.
EtherType Filters	0x67	Five bytes to load EtherType filters (METF). Data 2 = METF filter index (valid values are 0, 1, 2, 3). Data 3 = MSB of METF. ... Data 6 = LSB of METF.
Extended Decision Filter	0x68	Nine bytes to load the extended decision filters (MDEF_EXT and MDEF). Data 2 = MDEF filter index (valid values are 0...6). Data 3 — MSB of MDEF_EXT (DecisionFilter1). .... Data 6 = LSB of MDEF_EXT (DecisionFilter1). Data 7 = MSB of MDEF (DecisionFilter0). .... Data 10 = LSB of MDEF (DecisionFilter0). The command overwrites any previously stored value.

Table 38-531.Management Receive Filter Parameters

Parameter	Number	Parameter Data
Management Special Filter Modifiers	0x69	Four bytes to load the management special filter modifiers. Data 2 = MSB of MSFM register. ... Data 5 = LSB of MSFM register.

Table 38-532.Filter enable parameters

Bit	Name	Description
16:0	Reserved	Reserved.
17	RCV_TCO_EN	Receive TCO Packets Enabled. When this bit is set it enables the receive flow to the manageability block. This bit should be set only if at least one of EN_BMC2OS or EN_BMC2NET bits are set. This bit is usually set using the receive enable command (see Section , "Receive Enable Command").
18	KEEP_PHY_LINK_UP	Block PHY reset and power state changes. When this bit is set the PHY reset and power state changes does not get to the PHY.
22:19	Reserved	Reserved.
23	Enable Xsum Filtering to MNG	When this bit is set, only packets that pass the L3 and L4 checksum are send to the manageability block.
24	Reserved	Reserved.
25	FIXED_NET_TYPE	Fixed Net Type. If set, only packets matching the net type defined by the *NET_TYPE* field passes to manageability. Otherwise, both tagged and un-tagged packets can be forwarded to the manageability engine.
26	NET_TYPE	Net Type. 0b = Pass only un-tagged packets. 1b = Pass only VLAN tagged packets. Valid only if *FIXED_NET_TYPE* is set.
31:27	Reserved	Reserved.

Set Common Filters Command

The Set Common Filters command is a single fragment command capable of configuring the most common filters.

Note: If this command is used, all the other commands that programs forwarding filters should not be used (apart from the Clear All Fitters command). When this command is received, an implied Clear All Filters command is done before the application of this command.

The Set Common Filters command has two possible formats:

IPv4 format:

Function	Command	Byte Count	Data 1	Data 2:4	5:10	Data 11	Data 12	Data 13	Data 14:17
Set Common Filters	0xC2	17	Opcode = 0	Receive Control - see Table 38-533	MAC Address	BMC Alert Address	Interface Data Byte	Alert Value Byte	IPv4 Address

IPv6 format:

Function	Command	Byte Count	Data 1	Data 2:4	5:10	Data 11	Data 12	Data 13	Data 14:29
Set Common Filters	0xC2	29	Opcode = 0	Receive Control - see Table 38-533	MAC Address	BMC Alert Address	Interface Data Byte	Alert Value Byte	IPv6 Address

Table 38-533.Set Common Filters Receive Control Bytes

Byte	Bit	Field	Description
1	0	RCV_EN	Receive TCO Packets Enabled. When this bit is set it enables the receive flow to the manageability block. This bit should be set only if at least one of EN_BMC2O or EN_BMC2NET bits are set.
	1	EN_STA	Enable Status Reporting. 0b = Disable status reporting. 1b = Enable status reporting.
	2	Auto ARP	Automatically respond to ARP packets. Ignored in IPv6 mode. If this bit is set, broadcast ARP packets are handled by the 10 GbE controller and ARP requests to the IP address set in the command are responded to. Mutually exclusive to Configure ARP/ Neighborhood Filter bit. If this bit is set, the IP address must be valid. This bit is ignored if RCV_EN is cleared.
	3	Enable Xsum Filtering to MNG	When this bit is set, only packets that pass the L3 and L4 checksum are send to the manageability block. This bit is ignored if RCV_EN is cleared.
	5:4	Reserved	Reserved.
	7:6	Notification Method	Notification Method. Define the notification method the 10 GbE controller uses. 00b = SMBus alert. 01b = Asynchronous notify. 10b = Direct receive. 11b = Not supported.

Table 38-533.Set Common Filters Receive Control Bytes

Byte	Bit	Field	Description
2	8	CBDM	Configure the BMC Dedicated MAC Address. 0b = The 10 GbE controller shares the MAC address for manageability traffic with the host MAC address, which is specified in NVM words 0x0-0x2. 1b = The 10 GbE controller uses the MC dedicated MAC address as a filter for incoming receive packets. The MC MAC address is set in bytes 5:1 in this command. This bit is ignored if RCV_EN is cleared.
	9	Configure IP Address Filter	Automatically configure an IP address filter. If this bit is set, only packets matching this IP address is forwarded. If the *CBDM* bit is set, only packets matching the MAC and IP address is forwarded. This bit is ignored if RCV_EN is cleared.
	10	Configure RMCP 0x26F Filter	Automatically configure standard IPMI port 0x26F filters. If this bit is set, only packets matching this port is forwarded. If the *CBDM/ Configure IP Address Filter* bits are set, only packets matching the MAC and IP address and this port is forwarded. The other port enable bit (11) might add additional forwarding conditions. This bit is ignored if RCV_EN is cleared.
	11	Configure RMCP 0x298 Filter	Automatically configure standard IPMI port 0x298 filter. If this bit is set, only packets matching this port is forwarded. If the *CBDM/ Configure IP Address Filter* bits are set, only packets matching the MAC and IP address and this port is forwarded. The other port enable bit (10) might add additional forwarding conditions. This bit is ignored if RCV_EN is cleared.
	12	Configure ARP/ Neighborhood Filter	Automatically configure filters to enable this traffic to the MC (mutually exclusive to *Auto ARP* bit). If this bit is set, broadcast ARP packets are forwarded to the MC. In IPv4 mode, setting this bit enables forwarding of broadcast ARP requests and responses and unicast ARP responses. If the IP address is set, only a response to this address is forwarded. In IPv6 mode, setting this bit enables forwarding of all types of neighbor discovery and MLD ICMPv6 packet types: 0x86 (134d) = Router advertisement. 0x87 (135d) = Neighbor solicitation. 0x88 (136d) = Neighbor advertisement. 0x89 (137d) = Redirect. 0x82 (130d) = MLD query. 0x83 (131d) = MLDv1 report. 0x84 (132d) = MLD done. 0x8F (143d) = MLDv2 report.
	13	Configure DHCP port 0x44 Filter	Automatically configure DHCP port 44 filter to the MC. If this bit is set, multicast packets matching this port is forwarded. Otherwise, multicast packets are not forwarded to the MC. This bit is ignored if RCV_EN is cleared or in IPv6 mode.
	15:14	Reserved	Reserved.
3	16	Disable Host ARP	Configure ARP requests and network neighborhood packets not to go to the host. This bit should be cleared during normal operation. Ignored if both bit 12 and bit 2 are cleared or if RCV_EN is cleared.
	17	Disable Host DHCP	Configure DHCP packets (port 0x44) not to go to host. This bit should be cleared in normal operation. Ignored if bit 13 is cleared, RCV_EN is cleared, or in IPv6 mode.
	24:18	Reserved	Reserved.

Clear All Filters Command

The Clear all Filters command is a single fragment command capable of clearing all the receive filters currently programmed for manageability traffic.

Function	Command	Byte Count	Data
Clear all Filters	0xC3	1	0x00

38.37.5.12.2 Read SMBus Transactions

This section details the PT read transactions that the MC can send to the 10 GbE controller over the SMBus.

SMBus read transactions lists the different SMBus read transactions supported by the 10 GbE controller. All the read transactions are compatible with SMBus read block protocol format.

Table 38-534.SMBus Read Transactions

TCO Command	Transaction	Command	Opcode	Fragments
Receive TCO Packet	Block Read	0xD0 or 0xC0	First: 0x90 Middle: 0x10 Last[1]: 0x50	Multiple
Read Status	Block Read	0xD0 or 0xC0 or 0xDE	Single: 0xDD	Single
Get System MAC Address	Block Read	0xD4	Single: 0xD4	Single
Read Management Parameters	Block Read	0xD1	Single: 0xD1	Single
Read Management RCV Filter Parameters	Block Read	0xCD	Single: 0xCD	Single
Read Receive Enable Configuration	Block Read	0xDA	Single: 0xDA	Single
Get Controller Information	Block Read	0xD5	Single: 0xD5	Single
Get Common Filters	Block Read	0xD3	Single: 0xD3	Single

Notes:

1. The last fragment of the receive TCO packet is the packet status.

0xC0 or 0xD0 commands are used for more than one payload. If the MC issues these read commands, and the 10 GbE controller has no pending data to transfer, it always returns as default opcode 0xDD with the 10 GbE controller status and does not NACK the transaction.

If an SMBus Quick Read command is received, it is handled as a Product Name Request Status command.

Receive TCO LAN Packet Transaction

The MC uses this command to read packets received on the LAN and its status. When the 10 GbE controller has a packet to deliver to the MC, it asserts the SMBus notification for the MC to read the data (or direct receive). Upon receiving notification of the arrival of a LAN receive packet, the MC begins issuing a Receive TCO packet command using the block read protocol.

A packet can be transmitted to the MC in at least two fragments (at least one for the packet data and one for the packet status). As a result, the MC should follow the *F* and *L* bit of the opcode.

The opcode can have these values:

- 0x90 — First fragment
- 0x10 — Middle fragment
- When the opcode is 0x50, this indicates the last fragment of the packet, which contains packet status.

If a notification timeout is defined (in the NVM) and the MC does not finish reading the entire packet within the timeout period, since the packet has arrived, the packet is silently discarded. The time spent in ARA cycle or in reading the packet is not counted by the timeout counter.

Following is the receive TCO packet format and the data format returned from the 10 GbE controller.

Function	Command
Receive TCO Packet	0xC0 or 0xD0

Function	Byte Count	Data 1 (Opcode)	Data 2	...	Data N
Receive TCO First Fragment	N	0x90	Packet Data Byte	...	Packet Data Byte
Receive TCO Middle Fragment		0x10			
Receive TCO Last Fragment	9 (0x9)	0x50	See the section that follows.		

Receive TCO LAN Status Payload Transaction

This transaction is the last transaction that the 10 GbE controller issues when a packet received from the LAN is transferred to the MC. The transaction contains the status of the received packet.

The format of the status transaction is as follows:

Function	Byte Count	Data 1 (Opcode)	Data 2 – Data 17 (Status Data)
Receive TCO Long Status	9 (0x9)	0x50	See as follows.

The status is 8 bytes where byte 0 (bits 7:0) is set in Data 2 of the status and byte 7 in Data 9 of the status. Table 38-535 lists the content of the status data.

Table 38-535.CO LAN Packet Status Data

Name	Bits	Description
Packet Length	13:0	Packet length including CRC, only 14 LSB bits.
Reserved	15:14	Reserved.
Packet status	31:16	See Table 38-536.
VLAN	47:32	The two bytes of the VLAN header tag.
MNG status	63:48	See Table 38-537. This field should be ignored if Receive TCO is not enabled.

Table 38-536.Packet Status Information

Field	Bit(s)	Description
Reserved	15:4	Reserved.
LAN#	3:2	Indicates the source port of the packet.
VP	1	VLAN Stripped (indicates if the VLAN is part of the packet, or was removed).
CRC stripped	0	Insertion of CRC is needed.

Table 38-537.MNG Status

Name	Bits	Description
Reserved	15:9	Reserved.
Decision Filter match	8	Set when there is a match to one of the decision filters.
Decision Filter index	7:4	Indicates which of the decision filters match the packet. (allows for up to 16 filters - although only 8 are currently supported).
MNG VLAN Address Match	3	Set when the manageability packet matches one of the manageability VLAN filters.
Pass MNG VLAN Filter Index	2:0	Indicates which of the VLAN filters match the packet.

38.37.5.12.3 Read Status Command

The MC should use this command after receiving a notification from the 10 GbE controller (such as SMBus alert). The 10 GbE controller also sends a notification to the MC in either of the following two cases:

- The MC asserts a request for reading the status.
- The 10 GbE controller detects a change in one of the Status Data 1 bits or NVM error bit in Data 2 (and was set to send status to the MC on status change) in the Receive Enable command.

Note: Commands 0xC0/0xD0 are for backward compatibility and can be used for other payloads. The 10 GbE controller defines these commands in the opcode as well as which payload this transaction is. When the 0XDE command is set, the 10 GbE controller always returns opcode 0XDD with the 10 GbE controller status. The MC reads the event causing the notification, using the Read Status command as follows.

The 10 GbE controller's response to one of the commands (0xC0 or 0xD0) in a given time as defined in the SMBus Notification Timeout and Flags word in the NVM.

Function	Command
Read Status	0XC0 or 0XD0 or 0XDE

Function	Byte Count	Data 1 (Opcode)	Data 2 (Status Data 1)	Data 3 (Status Data 2)
Receive TCO Partial Status	3	0XDD	See as follows.	

This command can also be executed using the I^2C quick read format as follows.

1	7	1	1	8	1	8	1	8	1	1
Start	Slave Address	Rd	Ack	Byte Count	Ack	Status Data 1	Ack	Status Data 2	Ack	Stop
		1	0	0000 0002	0		0		1	

Table 38-538 lists the status data byte 1 parameters.

Table 38-538.Status Data Byte 1

Bit	Name	Description
7	LAN Port LSB	LAN port LSB together with LAN Port MSB define port that sent status. See further information in the description of LAN Port MSB (bit 2).
6	TCO Command Aborted	1b = A TCO command abort event occurred since the last read status cycle. 0b = A TCO command abort event did not occur since the last read status cycle.
5	Link Status Indication	0b = LAN link down. 1b = LAN link up.
4	PHY Link Forced Up	Contains the value of the *PHY_Link_Up* bit. When set, indicates that the PHY link is configured to keep the link up.
3	Initialization Indication	0b = An NVM reload event has not occurred since the last read status cycle. 1b = An NVM reload event has occurred since the last read status cycle[1].
2	LAN Port MSB	Defines together with LAN Port LSB the port that sent the status: Lan Port MSB / Lan Port LSB 0 / 0 — Status came from LAN port 0. 0 / 1 — Status came from LAN port 1. 1 / 0 — Status came from LAN port 2. 1 / 1 — Status came from LAN port 3.
1:0	Power State	00b = Dr state. 01b = D0u state. 10b = D0 state. 11b = D3 state. When more than one function is mapped to the same port, the highest power state of the mapped functions is reported according to the following order: Dr < D3 < D0u < D0.

Notes:

1. This indication is asserted when the 10 GbE controller the manageability block reloads the NVM and its internal database is updated to the NVM default values. This is an indication that the external MC should reconfigure the 10 GbE controller, if other values other than the NVM default should be configured.

Status data byte 2 is used by the MC to indicate whether the LAN device driver is up and running.

The LAN device driver valid indication is a bit set by the LAN device driver during initialization; the bit is cleared when the LAN device driver enters a Dx state or is cleared by the hardware on a PCI reset.

Table 38-539 lists status data byte 2.

Table 38-539.Status Data Byte 2

Bit	Name	Description
7:6	Reserved	Reserved.
5	NVM Error	If set, indicates that a CRC/checksum error was detected in one of the manageability related NVM sections.
4	Reserved	Reserved.
3	Driver Valid Indication	0b = LAN driver is not up. 1b = LAN driver is up.

Get System MAC Address

The get system MAC address returns the system MAC address over to the SMBus. This command is a single-fragment read block transaction that returns the system MAC address.

When a single function is defined on the port, it returns the LAN MAC address of this function as read from the PF allocations NVM section or from the alternate RAM or as set by the Manage MAC Address Write AQ command. When more than one function is defined on the port, it returns the address of the lowest defined function on this port.

Get system MAC address format:

Function	Command
Get system MAC address	0xD4

Data returned from the 10 GbE controller:

Function	Byte Count	Data 1 (Opcode)	Data 2	...	Data 7
Get system MAC address	7	0xD4	MAC address MSB	...	MAC address LSB

Read Management Parameters

In order to read the management parameters, the MC should execute two SMBus transactions. The first transaction is a block write that sets the parameter that the MC wants to read. The second transaction is block read that reads the parameter.

Block write transaction:

Function	Command	Byte Count	Data 1
Management control request	0xC1	1	Parameter number

Following the block write, the MC should issue a block read that reads the parameter that was set in the Block Write command:

Function	Command
Read management parameter	0xD1

Data returned:

Function	Byte Count	Data 1 (Opcode)	Data 2	Data 3	...	Data N
Read management parameter	N	0xD1	Parameter number	Parameter dependent		

The returned data is in the same format of the MC command.

The returned data is as follow.

Parameter	#	Parameter Data
Keep PHY Link Up	0x00	A single byte parameter: Data 2 — Bit 0 = Set to indicate that the PHY link for this port should be kept up. Sets the keep_PHY_link_up bit. When cleared, clears the keep_PHY_link_up bit. Bit [7:1] = Reserved.
Wrong parameter request	0xFE	Returned by the 10 GbE controller only. This parameter is returned on a read transaction, if in the previous Read command the MC sets a parameter that is not supported by the 10 GbE controller.
Product Name is not ready	0xFF	Returned by the 10 GbE controller only, on a Read Parameters command when the data that should have been read is not ready. This parameter has no data. The MC should retry the read transaction. This value is also returned if the byte count is illegal or if the read command is not preceded by a Write command.

The parameter that is returned might not be the parameter requested by the MC. The MC should verify the parameter number (default parameter to be returned is 0x1).

If the parameter number is 0xFF, it means that the data that was requested from the 10 GbE controller is not ready yet. The MC should retry the read transaction.

It is responsibility of the MC to follow the procedure previously defined. When the MC sends a Block Read command (as previously described) that is not preceded by a Block Write command with bytecount=1, the 10 GbE controller sets the parameter number in the read block transaction to be 0xFF.

Read Management Receive Filter Parameters

In order to read the management receive filter parameters, the MC should execute two SMBus transactions. The first transaction is a block write that sets the parameter that the MC wants to read. The second transaction is block read that read the parameter.

Block write transaction:

Function	Command	Byte Count	Data 1	Data 2
Update MNG RCV filter parameters	0xCC	1 or 2	Parameter number	Parameter data

The different parameters supported for this command are the same as the parameters supported for the update management receive filter parameters.

Following the block write the MC should issue a block read that reads the parameter that was set in the Block Write command:

Function	Command
Request MNG RCV filter parameters	0xCD

Data returned from the 10 GbE controller:

Function	Byte Count	Data 1 (Opcode)	Data 2	Data 3	...	Data N
Read MNG RCV filter parameters	N	0xCD	Parameter number	Parameter dependent		

The parameter that is returned might not be the parameter requested by the MC. The MC should verify the parameter number (default parameter to be returned is 0x1).

If the parameter number is 0xFF, it means that the data that was requested from the 10 GbE controller should supply is not ready yet. The MC should retry the read transaction.

It is MC's responsibility to follow the procedure previously defined. When the MC sends a Block Read command (as previously described) that is not preceded by a Block Write command with bytecount=1, the 10 GbE controller sets the parameter number in the read block transaction to be 0xFF.

Parameter	#	Parameter Data
Filters Enable	0x01	None.
MNGONLY Configuration	0x0F	None.
Flex Filter Enable Mask and Length	0x10	None.

Parameter	#	Parameter Data
Flex Filter Data	0x11	Data 2 — Group of Flex Filter's Bytes: 0x0 = Bytes 0-29. 0x1 = Bytes 30-59. 0x2 = Bytes 60-89. 0x3 = Bytes 90-119. 0x4 = Bytes 120-127.
Filters Valid	0x60	None.
Decision Filters	0x61	This command is obsolete. Use 0x68 instead.
VLAN Filters	0x62	One byte to define the accessed VLAN tag filter (PRT_MNG_MAVTV). Data 2 — VLAN Filter number.
Flex Ports Filters	0x63	One byte to define the accessed manageability flex port filter (PRT_MNG_MFUTP). Data 2 — Flex port filter number.
IPv4 Filter	0x64	One byte to define the accessed IPv4 address filter (PRT_MNG_MIPAF4). Data 2 — IPv4 address filter number.
IPv6 Filters	0x65	One byte to define the accessed IPv6 address filter (PRT_MNG_MIPAF6). Data 2 — Pv6 address filter number.
MAC Filters	0x66	One byte to define the accessed MAC address filters pair (PRT_MNG_MMAL, PRT_MNG_MMAH). Data 2 — MAC address filters pair number (0-3).
EtherType Filters	0x67	1 byte to define EtherType filters (PRT_MNG_METF). Data 2 — METF filter index (valid values are 0 - 3).
Extended Decision Filter	0x68	1 byte to define the extended decisions filters (PRT_MNG_MDEF_EXT and PRT_MNG_MDEF). Data 2 — MDEF filter index (valid values are 0 - 6).
Management Special Filter Modifiers	0x69	
Wrong parameter request	0xFE	Returned by the 10 GbE controller only. This parameter is returned on a read transaction, if in the previous Read command the MC sets a parameter that is not supported by the 10 GbE controller.
Intel® C620 Series Chipset is not ready	0xFF	Returned by the 10 GbE controller only on a Read Parameters command when the data that should have been read is not ready. This parameter has no data. This value is also returned if the byte count is illegal or if the Read command is not preceded by a Write command.

Get Controller Information Command

The MC uses this command to get the controller identification. Each parameter is returned using a different parameter in the block write transaction.

In order to read the controller information, the MC should execute two SMBus transactions. The first transaction is a block write that sets the parameter that the MC wants to read. The second transaction is block read that read the parameter.

Block write transaction:

Function	Command	Byte Count	Data 1
Get Controller Information	0xD5	1	Parameter number

Following the block write, the MC should issue a block read that reads the parameter that was set in the Block Write command:

Function	Command
Get Controller Information	0xD5

Data returned from the 10 GbE controller:

Function	Byte Count	Command	Data 2 (Op-Code)	Data 3 -n
Get Controller Information	See Table 38-540.	0xD5	See Table 38-540.	See Table 38-540 for the data for each opcode.

Table 38-540.Get Controller Information Data

Parameter	Byte Count	Description	Notes
0x00	5	Data 4:3: Device ID. Data 5: Silicon Revision (RevID).	This is the hardware default value, not any value programmed via the NVM.
0x0B	4	Data 4:3 NVM Image version.	
0x0C	6	Data 6:3: Firmware ROM Internal version.	
0x0D	6	Data 6:3: Firmware Flash Internal version.	
0x0E	4	Data 4:3: PXE firmware version.	MajorVersion.MinorVersion.Build.SubBuild. If a version is not found, a value of 0xFFFF is returned.
0x0F	4	Data 4:3: iSCSI firmware version.	
0x10	4	Data 4:3: uEFI firmware version.	
0x16	4	Reserved.	
0xFE	2	Wrong parameter request.	Returned by the 10 GbE controller only. This parameter is returned on a read transaction, if in the previous Read command the MC sets a parameter that is not supported by the 10 GbE controller.
0xFF	2	Product Name is not ready.	Returned by the 10 GbE controller only, on a Read Parameters command when the data that should have been read is not ready. This parameter has no data. The MC should retry the read transaction. This value is also returned if the byte count is illegal or if the Read command is not preceded by a Write command.

Get Common Filters Command

The MC uses this command to get the common filters setting. This data can be configured when using Set Common Filters command. The first transaction is a block write that alerts that the MC wants to read the filters configuration. The second transaction is block read that read the configuration.

Block write transaction:

Function	Command	Byte count	Data
Get Common Filters	0xD3	1	0x00

Following the block write the MC should issue a block read that reads the filter settings:

Function	Command
Get Common filters	0xD3

Data returned from the 10 GbE controller:

Function	Byte Count	Command	Data 1	Data 2:4	5:10	Data 11	Data 12	Data 13	Data 14:17
Get Common Filters	18	0xD3	0	Receive Control - see Table 38-533	MAC Address	BMC Alert Address	Interface Data Byte	Alert Value Byte	IPv4 Address

Function	Byte Count	Command	Data 1	Data 2:4	5:10	Data 11	Data 12	Data 13	Data 14:29
Get Common Filters	30	0xD3	0	Receive Control - see Table 38-533	MAC Address	BMC Alert Address	Interface Data Byte	Alert Value Byte	IPv6 Address

If an error occurs, the following answers might be returned:

Function	Command	Byte Count	Data 1
Get Common Filters	0xD3	1	0xFF

This response is by the 10 GbE controller on a Read Common Filter command when the data that should have been read is not ready. This parameter has no data. The MC should retry the read transaction.

Read Receive Enable Configuration

The MC uses this command to read the receive configuration data. This data can be configured when using the Receive Enable command or through the NVM.

The Read Receive Enable Configuration command format (SMBus read block) is as follows:

Function	Command
Read Receive Enable	0xDA

Data returned from the 10 GbE controller:

Function	Byte Count	Data 1 (Opcode)	Data 2	Data 3	...	Data 8	Data 9	...	Data 12	Data 13	Data 14	Data 15
Read Receive Enable	15 (0x0F)	0xDA	Receive Control Byte	MAC Addr MSB	...	MAC Addr LSB	IP Addr MSB	...	IP Addr LSB	BMC SMBus Addr	I/F Data Byte	Alert Value Byte

The details of each field is specified in sub-section Section “Receive Enable Command”.

38.37.5.13 Example Configuration Steps

This section provides sample configuration settings for common filtering configurations. Four examples are presented. The examples are in pseudo code format, with the name of the SMBus command followed by the parameters for that command and an explanation.

Example 1 - Shared MAC, RMCP Only Ports

This example is the most basic configuration. The MAC address filtering is shared with the host operating system and only traffic directed the RMCP ports (0x26F and 0x298) is filtered. For this example, the MC must issue gratuitous ARPs because no filter is enabled to pass ARP requests to the MC.

Step 1: Disable existing filtering

`Receive Enable[00]`

Using the basic form of the Receive Enable command, this prevents any packets from reaching the MC by disabling filtering:

Receive Enable Control 0x00:

- Bit 0 [0] – Disable Receiving of packets

Step 2: Configure MDEF[0]

`Update Manageability Filter Parameters [68, 0, C0000000, 00000000]`

Use the Update Manageability Filter Parameters command to update Decision Filters (MDEF) (parameter 0x68). This updates MDEF[0], as indicated by the second parameter (0).

MDEF[0] value of 0xC0000000:

- Bit 30 [1] – port 0x298
- Bit 31 [1] – port 0x26F

MDEF_EXT[0] value of 0x0000000:

Step 3: Configure *MNGONLY*

`Update Manageability Filter Parameters [F, 0, 00000001]`

Use the Update Manageability Filter Parameters command to update Manageability Only (*MNGONLY*) (parameter 0xF) so that port 0x298 and 0x26F would not be sent to the host.

- Bit [0] - *MDEF[0]* is exclusive to the MC.

Step 4: - Enable Filtering

`Receive Enable [05]`

Using the basic form of the Receive Enable command:

Receive Enable Control 0x05:

- Bit 0 [1] = Enable receiving of packets
- Bit 2 [1] = Enable status reporting (such as link lost)
- Bit 5:4 [00] = Notification method = SMB alert
- Bit 7 [0] = Use shared MAC

The resulting MDEF filters are as follows:

Table 38-541.Example 1 MDEF Results

Manageability Decision Filter (MDEF)									
Filter		0	1	2	3	4	5	6	7
L2 Unicast Address[3:0]	AND								
Broadcast	AND								
Manageability VLAN[7:0]	AND								
IPv6 Address[3:0]	AND								
IPv4 Address[3:0]	AND								
L2 Unicast Address[3:0]	OR								
Broadcast	OR								
Multicast	AND								
ARP Request	OR								
ARP Response	OR								
Neighbor Discovery	OR								
Port 0x298	OR	X							
Port 0x26F	OR	X							
Flex Port 7:0	OR								
Flex TCO	OR								

Example 2 - Dedicated MAC, auto ARP Response and RMCP Port Filtering

This example shows a common configuration; the MC has a dedicated MAC and IP address. Automatic ARP responses are enabled as well as RMCP port filtering. By enabling automatic ARP responses, the MC is not required to send the gratuitous ARPs as it did in Example 1.

For demonstration purposes, the dedicated MAC address is calculated by reading the system MAC address and adding a one to it; assume the system MAC is AABBCCDC. The IP address for this example is 1.2.3.4. Additionally, the XSUM filtering is enabled.

Note that not all Intel Ethernet controllers support automatic ARP responses, refer to product specific documentation.

Example 2 - Pseudo Code

Step 1: Disable existing filtering

```
Receive Enable[00]
```

Using the basic form of the Receive Enable command, this prevents any packets from reaching the MC by disabling filtering:

Receive Enable Control 0x00:

- Bit 0 [0] – Disable Receiving of packets

Step 2: Read System MAC Address

```
Get System MAC Address []
```

Reads the system MAC address. Assume a returned AABBCCDC for this example.

Step 3: Configure XSUM Filter

```
Update Manageability Filter Parameters [01, 00800000]
```

Use the Update Manageability Filter Parameters command to update Filters Enable settings (parameter 1). This sets the Manageability Control (*PRT_MNG_MANC*) register.

PRT_MNG_MANC Register 0x00800000:

- Bit 23 [1] - XSUM Filter Enable

  Note that some of the following configuration steps manipulate the *PRT_MNG_MANC* register indirectly, this command sets all bits except XSUM to 0b. It is important to either do this step before the others, or to read the value of the *PRT_MNG_MANC* and then write it back with only bit 32 changed. Also note that the XSUM enable bit might differ between Ethernet controllers, refer to product specific documentation.

Step 4: Configure MDEF[0]

```
Update Manageability Filter Parameters [68, 0, C0000000, 00000000]
```

Use the Update Manageability Filter Parameters command to update Decision Filters (MDEF) (parameter 0x68). This updates MDEF[0], as indicated by the second parameter (0).

MDEF value of 0x00000C00:

- Bit 30 [1] – port 0x298
- Bit 31 [1] – port 0x26F

MDEF_EXT[0] value of 0x0000000:

Step 5: Configure MDEF[1]

```
Update Manageability Filter Parameters [68, 1, 10000000, 00000000]
```

Use the Update Manageability Filter Parameters command to update Decision Filters (MDEF) (parameter 0x61). This updates MDEF[1], as indicated by the second parameter (1).

MDEF value of 0x10000000:

- Bit 28 [1] – ARP Requests

MDEF_EXT[1] value of 0x0000000:

When enabling automatic ARP responses, the ARP requests still go into the manageability filtering system and as such need to be designated as also needing to be sent to the host. For this reason a separate MDEF is created with only ARP request filtering enabled.

Refer to the next step for more details.

Step 6: Configure Manageability only

```
Update Manageability Filter Parameters [F, 0, 00000001]
```

Use the Update Manageability Filter Parameters command to update Manageability Only (MNGONLY) (parameter 0xF) so that port 0x298 and 0x26F would not be sent to the host.

- Bit [0] - MDEF[0] is exclusive to the MC.

  This enables ARP requests to be passed to both manageability and to the host. Specified separate MDEF filter for ARP requests. If ARP requests had been added to *MDEF[0]* and then *MDEF[0]* specified in management only configuration then not only would RMCP traffic (ports 0x26F and 0x298) be sent only to the MC, ARP requests would have also been sent to the MC only.

Step 7: Enable Filtering

```
Receive Enable [8D, AABBCCDD, 01020304, 00, 00, 00]
```

Using the advanced version Receive Enable command, the first parameter:
Receive Enable Control 0x8D:

- Bit 0 [1] – Enable receiving of packets
- Bit 2 [1] – Enable status reporting (such as link lost)
- Bit 3 [1] – Enable automatic ARP responses
- Bit 5:4 [00] – Notification method = SMB alert
- Bit 7 [1] - Use dedicated MAC
  Second parameter is the MAC address (AABBCCDD).
  Third parameter is the IP address(01020304).

The last three parameters are zero when the notification method is SMB alert.

The resulting MDEF filters are as follows:

Table 38-542.Example 2 MDEF Results

Manageability Decision Filter (MDEF)									
Filter		0	1	2	3	4	5	6	7
L2 Unicast Address[3:0]	AND								
Broadcast	AND								
Manageability VLAN[7:0]	AND								
IPv6 Address[3:0]	AND								
IPv4 Address[3:0]	AND								
L2 Unicast Address[3:0]	OR								
Broadcast	OR								
Multicast	AND								
ARP Request	OR		X						
ARP Response	OR								
Neighbor Discovery	OR								
Port 0x298	OR	X							
Port 0x26F	OR	X							
Flex Port [7:0]	OR								
Flex TCO	OR								

38.37.5.13.1 Example 3 - Dedicated MAC and IP Address

This example provided, the MC with a dedicated MAC and IP address enables it to receive ARP requests. The MC is then responsible for responding to ARP requests.

For demonstration purposes, the dedicated MAC address is calculated by reading the system MAC address and adding a one to it; assume the system MAC is AABBCCDC. The IP address for this example is 1.2.3.4. For this example, the Receive Enable command is used to configure the MAC address filter.

In order for the MC to be able to receive ARP requests, it needs to specify a filter for this, and that filter needs to be included in the manageability-to-host filtering so that the host operating system can also receive ARP requests.

Example 3 - Pseudo Code

Step 1: Disable existing filtering

`Receive Enable[00]`

Using the basic form of the Receive Enable command, this prevents any packets from reaching the MC by disabling filtering:

Receive Enable Control 0x00:

- Bit 0 [0] – Disable receiving of packets

Step 2: Read System MAC Address

`Get System MAC Address []`

Reads the system MAC address. Assume a returned AABBCCDC for this example.

Step 3: Configure IP Address Filter

`Update Manageability Filter Parameters [64, 00, 01020304]`

Use the update manageability filter parameters to configure an IPv4 filter.

The first parameter (0x64) specifies that we are configuring an IPv4 filter.

The second parameter (0x00) indicates which IPv4 filter is being configured; in this case filter 0.

The third parameter is the IP address – 1.2.3.4.

Step 4: Configure MAC Address Filter

`Update Manageability Filter Parameters [66, 00, AABBCCDD]`

Use the update manageability filter parameters to configure a MAC address filter.

The first parameter (0x66) specifies that we are configuring a MAC address filter.

The second parameter (0x00) indicates which MAC address filter is being configured; in this case filter 0.

The third parameter is the MAC address - AABBCCDD

Step 5: Configure MDEF[0] for IP and MAC Filtering

`Update Manageability Filter Parameters [68, 0, 00002001, 00000000]`

Use the Update Manageability Filter Parameters command to update Decision Filters (MDEF) (parameter 0x68). This updates MDEF[0], as indicated by the second parameter (0).

MDEF value of 00002001:

- Bit 0 [1] – MAC[0] address filtering
- Bit 13 [1] – IP[0] address filtering

MDEF_EXT[0] value of 0x0000000:

Step 6: Configure MDEF[1]

`Update Manageability Filter Parameters [68, 1, 10000000]`

Use the Update Manageability Filter Parameters command to update Decision Filters (MDEF) (parameter 0x68). This updates MDEF[1], as indicated by the second parameter (1).

MDEF value of 10000000:

- Bit 28 [1] – ARP requests

MDEF_EXT[1] value of 0x0000000:

Step 7: Configure the Management to Host Filter

`Update Manageability Filter Parameters [F, 0, 00000001]`

Use the Update Manageability Filter Parameters command to update Manageability Only (*MNGONLY*) (parameter 0xF) so that the dedicated MAC/IP traffic would not be sent to the host. Note that given the host does not program this address in it's L2 filtering, this step is not a must, unless the host chooses to work in promiscuous mode.

- Bit [0] - MDEF[0] is exclusive to the MC.

Step 8: Enable Filtering

`Receive Enable [05]`

Using the basic form of the Receive Enable command,:

Receive Enable Control 0x05:

- Bit 0 [1] – Enable receiving of packets
- Bit 2 [1] – Enable status reporting (such as link lost)
- Bit 5:4 [00] – Notification method = SMB alert

The resulting MDEF filters are as follows:

Table 38-543.Example 3 MDEF Results

Manageability Decision Filter (MDEF)									
Filter		**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
L2 Unicast Address[3:0]	AND	0001							
Broadcast	AND								
Manageability VLAN[7:0]	AND								
IPv6 Address[3:0]	AND								
IPv4 Address[3:0]	AND	0001							
L2 Unicast Address[3:0]	OR								
Broadcast	OR								
Multicast	AND								
ARP Request	OR		X						
ARP Response	OR								
Neighbor Discovery	OR								
Port 0x298	OR								
Port 0x26F	OR								
Flex Port [7:0]	OR								
Flex TCO	OR								

38.37.5.13.2 Example 4 - Dedicated MAC and VLAN Tag

This example shows an alternate configuration; the MC has a dedicated MAC and IP address, along with a VLAN tag of 0x32 are required for traffic to be sent to the MC. This means that all traffic with a VLAN and matching tag is sent to the MC.

For demonstration purposes, the dedicated MAC address is calculated by reading the system MAC address and adding a one to it; assume the system MAC is AABBCCDC. The IP address for this example is 1.2.3.4 and the VLAN tag is 0x0032.

Additionally, the XSUM filtering is enabled.

Example 4 - Pseudo Code

Step 1: Disable existing filtering

`Receive Enable[00]`

Using the basic form of the Receive Enable command, this prevents any packets from reaching the MC by disabling filtering:

Receive Enable Control 0x00:

- Bit 0 [0] – Disable receiving of packets

Step 2: - Read System MAC Address

`Get System MAC Address []`

Reads the system MAC address. Assume a returned AABBCCDC for this example.

Step 3: Configure XSUM Filter

`Update Manageability Filter Parameters [01, 00800000]`

Use the Update Manageability Filter Parameters command to update filters enable settings (parameter 1). This sets the Manageability Control (*PRT_MNG_MANC*) register.

PRT_MNG_MANC Register 0x00800000:

- Bit 23 [1] – XSUM filter enable

Note that some of the following configuration steps manipulate the *PRT_MNG_MANC* register indirectly. This command sets all bits except XSUM to 0b. It is important to either do this step before the others, or to read the value of the *PRT_MNG_MANC* and then write it back with only bit 32 changed. Also note that the XSUM enable bit might differ between Ethernet controllers, refer to product specific documentation.

Step 4: Configure VLAN 0 Filter

`Update Manageability Filter Parameters [62, 0, 0032]`

Use the Update Manageability Filter Parameters command to configure VLAN filters. Parameter 0x62 indicates an update to the VLAN filter. The second parameter indicates which VLAN filter (0 in this case). The last parameter is the VLAN ID (0x0032).

Step 5: Configure MDEF[0]

`Update Manageability Filter Parameters [68, 0, 00000020, 00000000]`

Use the Update Manageability Filter Parameters command to update Decision Filters (MDEF) (parameter 0x68). This updates MDEF[0], as indicated by the second parameter (0).

MDEF value of 0x00000020:

- Bit 5 [1] – VLAN[0] AND

MDEF_EXT[0] value of 0x0000000:

Step 6: Enable Filtering

`Receive Enable [85, AABBCCDD, 01020304, 00, 00, 00]`

Using the advanced version Receive Enable command, the first parameter:

Receive Enable Control 0x85:

- Bit 0 [1] – Enable receiving of packets
- Bit 2 [1] – Enable status reporting (such as link lost)

- Bit 5:4 [00] – Notification method = SMB alert
- Bit 7 [1] – Use dedicated MAC
  Second parameter is the MAC address: AABBCCDD.
  The third parameter is the IP address: 01020304.
  The last three parameters are zero when the notification method is SMBus alert.

The resulting MDEF filters are as follows:

Table 38-544.Example 4 MDEF Results

Manageability Decision Filter (MDEF)									
Filter		0	1	2	3	4	5	6	7
L2 Unicast Address[3:0]	AND								0001
Broadcast	AND								
Manageability VLAN[7:0]	AND	X							
IPv6 Address[3:0]	AND								
IPv4 Address[3:0]	AND								
L2 Unicast Address[3:0]	OR								
Broadcast	OR								
Multicast	AND								
ARP Request	OR								
ARP Response	OR								
Neighbor Discovery	OR								
Port 0x298	OR								
Port 0x26F	OR								
Flex Port [7:0]	OR								
Flex TCO	OR								

38.37.5.14 SMBus Troubleshooting

This section outlines the most common issues found while working with PT using the SMBus sideband interface.

38.37.5.14.1 TCO Alert Line Stays Asserted After a Power Cycle

After the 10 GbE controller resets, all its ports indicates a status change. If the MC only reads status from one port (slave address), the other one continues to assert the TCO alert line.

Ideally, the MC should use the ARA transaction (see Section 38.37.5.10) to determine which slave asserted the TCO alert. Many customers only want to use one port for manageability, thus using ARA might not be optimal.

An alternative to using ARA is to configure part of the ports to not report status and to set its SMBus timeout period. In this case, the SMBus timeout period determines how long a port asserts the TCO alert line awaiting a status read from a MC; by default this value is zero (indicates an infinite timeout).

The SMBus configuration section of the NVM has a *SMBus Notification Timeout* (ms) field that can be set to a recommended value of 0xFF (for this issue). Note that this timeout value is for all slave addresses. Along with setting the *SMBus Notification*

Timeout to 0xFF, it is recommended that the other ports be configured in the NVM to disable status alerting. This is accomplished by having the *Enable Status Reporting* bit set to 0b for the desired ports in the LAN configuration section of the NVM.

The third solution for this issue is to have the MC hard-code the slave addresses to always read from all ports. As with the previous solution, it is recommend that the other ports have status reporting disabled.

38.37.5.14.2 When SMBus Commands are Always NACK'd

There are several reasons why all commands sent to the 10 GbE controller from a MC could be NACK'd. The following are most common:

- Invalid NVM Image — The image itself might be invalid or it could be a valid image and is not a PT image, as such SMBus connectivity is disabled.
- The MC is not using the correct SMBus address — Many MC vendors hard-code the SMBus address(es) into their firmware. If the incorrect values are hard-coded, the 10 GbE controller does not respond.
- The SMBus address(es) can be dynamically set using the SMBus ARP mechanism.
- The MC is using the incorrect SMBus interface — The NVM might be configured to use one physical SMBus port; however, the MC is physically connected to a different one.
- Bus Interference — The bus connecting the MC and the 10 GbE controller might be unstable.

38.37.5.14.3 SMBus Clock Speed is 16.6666 kHz

This can happen when the SMBus connecting the MC and the 10 GbE controller is also tied into another device (such as an ICH) that has a maximum clock speed of 16.6666 kHz. The solution is to not connect the SMBus between the 10 GbE controller and the MC to this device.

38.37.5.14.4 A network Based Host Application is Not Receiving Any Network Packets

Reports have been received about an application not receiving any network packets. The application in question was NFS under Linux. The problem was that the application was using the RMPC/RMCP+ IANA reserved port 0x26F (623) and the system was also configured for a shared MAC and IP address with the operating system and the MC.

The management control to host configuration, in this situation, was setup not to send RMCP traffic to the operating system (this is typically the correct configuration). This means that no traffic send to port 623 was being routed.

The solution in this case is to configure the problematic application NOT to use the reserved port 0x26F.

38.37.5.14.5 Unable to Transmit Packets From the MC

If the MC has been transmitting and receiving data without issue for a period of time and then begins to receive NACKs from the 10 GbE controller when it attempts to write a packet, the problem is most likely due to the fact that the buffers internal to the 10 GbE controller are full of data that has been received from the network but has yet to be read by the MC.

Being an embedded device, the 10 GbE controller has limited buffers that are shared for receiving and transmitting data. If a MC does not keep the incoming data read, the 10 GbE controller can be filled up. This prevents the MC form transmitting more data, resulting in NACKs.

If this situation occurs, the recommended solution is to have the MC issue a Receive Enable command to disable more incoming data, read all the data from the 10 GbE controller, and then use the Receive Enable command to enable incoming data.

38.37.5.14.6 SMBus Fragment Size

The SMBus specification indicates a maximum SMBus transaction size of 32 bytes. Most of the data passed between the 10 GbE controller and the MC over the SMBus is RMCP/RMCP+ traffic, which by its very nature (UDP traffic) is significantly larger than 32 bytes in length. Multiple SMBus transactions might therefore be required to move data from the 10 GbE controller to the MC or to send a data from the MC to the 10 GbE controller.

Recognizing this bottleneck, the 10 GbE controller handles up to 240 bytes of data in a single transaction. This is a configurable setting in the NVM. The default value in the NVM images is 32, per the SMBus specification. If performance is an issue, increase this size.

During initialization, firmware within the 10 GbE controller allocates buffers based upon the SMBus fragment size setting within the NVM. Product Name firmware has a finite amount of RAM for its use: the larger the SMBus fragment size, the fewer buffers it can allocate. Because this is true, MC implementations must take care to send data over the SMBus efficiently.

For example, the 10 GbE controller firmware has 3 KB of RAM it can use for buffering SMBus fragments. If the SMBus fragment size is 32 bytes then the firmware could allocate 96 buffers of size 32 bytes each. As a result, the MC could then send a large packet of data (such as KVM) that is 800 bytes in size in 25 fragments of size 32 bytes apiece.

However, this might not be the most efficient way because the MC must break the 800 bytes of data into 25 fragments and send each one at a time.

If the SMBus fragment size is changed to 240 bytes, the 10 GbE controller firmware can create 12 buffers of 240 bytes each to receive SMBus fragments. The MC can now send that same 800 bytes of KVM data in only four fragments, which is much more efficient.

The problem of changing the SMBus fragment size in the NVM is if the MC does not also reflect this change. If a programmer changes the SMBus fragment size in the 10 GbE controller to 240 bytes and then wants to send 800 bytes of KVM data, the MC can still only send the data in 32 byte fragments. As a result, firmware runs out of memory.

This is because firmware created the 12 buffers of 240 bytes each for fragments; however, the MC is only sending fragments of size 32 bytes. This results in a memory waste of 208 bytes per fragment. Then when the MC attempts to send more than 12 fragments in a single transaction, the 10 GbE controller NACKs the SMBus transaction due to not enough memory to store the KVM data.

In summary, if a programmer increases the size of the SMBus fragment size in the NVM (recommended for efficiency purposes) take care to ensure that the MC implementation reflects this change and uses that fragment size to its fullest when sending SMBus fragments.

38.37.5.14.7 Losing Link

Normal behavior for the Ethernet controller when the system powers down or performs a reset is for the link to temporarily go down and then back up again to re-negotiate the link speed. This behavior can have adverse affects on manageability.

For example, if there is an active FTP or Serial Over LAN (SoL) session to the MC, this connection can be lost. In order to avoid this possible situation, the MC can use the Management Control command to ensure the link stays active at all times.

This command is available when using the NC-SI sideband interface as well.

Care should be taken with this command, if the software device driver negotiates the maximum link speed, the link speed remains the same when the system powers down or resets. This can have undesirable power consumption consequences. Currently, when using NC-SI, the MC can re-negotiate the link speed. That functionality is not available when using the SMBus interface.

38.37.5.14.8 Enable Checksum Filtering

If checksum filtering is enabled, the MC does not need to perform the task of checking this checksum for incoming packets. Only packets that have a valid checksum is passed to the MC. All others are silently discarded.

This is a way to offload some work from the MC.

38.37.5.14.9 Still Having Problems?

If problems still exist, contact your field representative. Be prepared to provide the following:

- A SMBus trace if possible.
- A dump of the NVM image. This should be taken from the actual Product Name, rather than the NVM image provided by Intel. Parts of the NVM image are changed after writing (such as the physical NVM size).

38.37.6 Network Controller Sideband Interface (NC-SI) PT Interface

The NC-SI is a DMTF industry standard protocol for the sideband interface. NC-SI uses a modified version of the industry standard RMII interface for the physical layer as well as defining a new logical layer.

The NC-SI specification can be found at:

http://www.dmtf.org/

38.37.6.1 Overview

38.37.6.1.1 Terminology

The terminology in this section is taken from the NC-SI specification.

Table 38-545.NC-SI Terminology

Term	Definition
Frame Versus Packet	Frame is used in reference to Ethernet, whereas packet is used everywhere else.
External Network Interface	The interface of the network controller that provides connectivity to the external network infrastructure (port).
Internal Host Interface	The interface of the network controller that provides connectivity to the host operating system running on the platform.
Management Controller (BMC)	An intelligent entity comprising of hardware/firmware/software, that resides within a platform and is responsible for some or all management functions associated with the platform (BMC, service processor, etc.).
Network Controller (NC)	The component within a system that is responsible for providing connectivity to the external Ethernet network world.

Table 38-545.NC-SI Terminology

Term	Definition
Remote Media	The capability to allow remote media devices to appear as if they were attached locally to the host.
Network Controller Sideband Interface	The interface of the network controller that provides connectivity to a management controller. It can be shorten to sideband interface as appropriate in the context.
Interface	This refers to the entire physical interface, such as both the transmit and receive interface between the management controller and the network controller.
Integrated Controller	The term integrated controller refers to a network controller device that supports two or more channels for NC-SI that share a common NC-SI physical interface. For example, a network controller that has two or more physical network ports and a single NC-SI bus connection.
Multi-Drop	Multi-drop commonly refers to the case where multiple physical communication devices share an electrically common bus and a single device acts as the master of the bus and communicates with multiple slave or target devices. In NC-SI, a management controller serves the role as the master, and the network controllers are the target devices.
Point-to-Point	Point-to-point commonly refers to the case where only two physical communication devices are interconnected via a physical communication medium. The devices might be in a master/slave relationship, or could be peers. In NC-SI, point-to-point operation refers to the situation where only a single management controller and single network controller package are used on the bus in a master/slave relationship where the management controller is the master.
Channel	The control logic and data paths supporting NC-SI pass-through operation on a single network interface (port). A network controller that has multiple network interface ports can support an equivalent number of NC-SI channels.
Package	One or more NC-SI channels in a network controller that share a common set of electrical buffers and common buffer control for the NC-SI bus. Typically, there will be a single, logical NC-SI package for a single physical network controller package (chip or module). However, the specification allows a single physical chip or module to hold multiple NC-SI logical packages.
Control Traffic/Messages/Packets	Command, response and notification packets transmitted between BMC and the 10 GbE controller for the purpose of managing NC-SI.
Pass-Through Traffic/Messages/ Packets	Non-control packets passed between the external network and the BMC through the 10 GbE controller.
Channel Arbitration	Refer to operations where more than one of the network controller channels can be enabled to transmit pass-through packets to the BMC at the same time, where arbitration of access to the RXD, CRS_DV, and RX_ER signal lines is accomplished either by software of hardware means.
Logically Enabled/Disabled NC	Refers to the state of the network controller wherein pass-through traffic is able/unable to flow through the sideband interface to and from the management controller, as a result of issuing Enable/Disable Channel command.
NC RX	Defined as the direction of ingress traffic on the external network controller interface
NC TX	Defined as the direction of egress traffic on the external network controller interface
NC-SI RX	Defined as the direction of ingress traffic on the sideband enhanced NC-SI Interface with respect to the network controller.
NC-SI TX	Defined as the direction of egress traffic on the sideband enhanced NC-SI Interface with respect to the network controller.

38.37.6.1.2 System Topology

In NC-SI each physical endpoint (NC package) can have several logical slaves (NC channels).

NC-SI defines that one MC and up to four network controller packages can be connected to the same NC-SI link.

Figure 38-133 shows an example topology for a single MC (also know as BMC) and a single NC package. In this example, the NC package has two NC channels.

Figure 38-133.Single NC Package, Two NC Channels

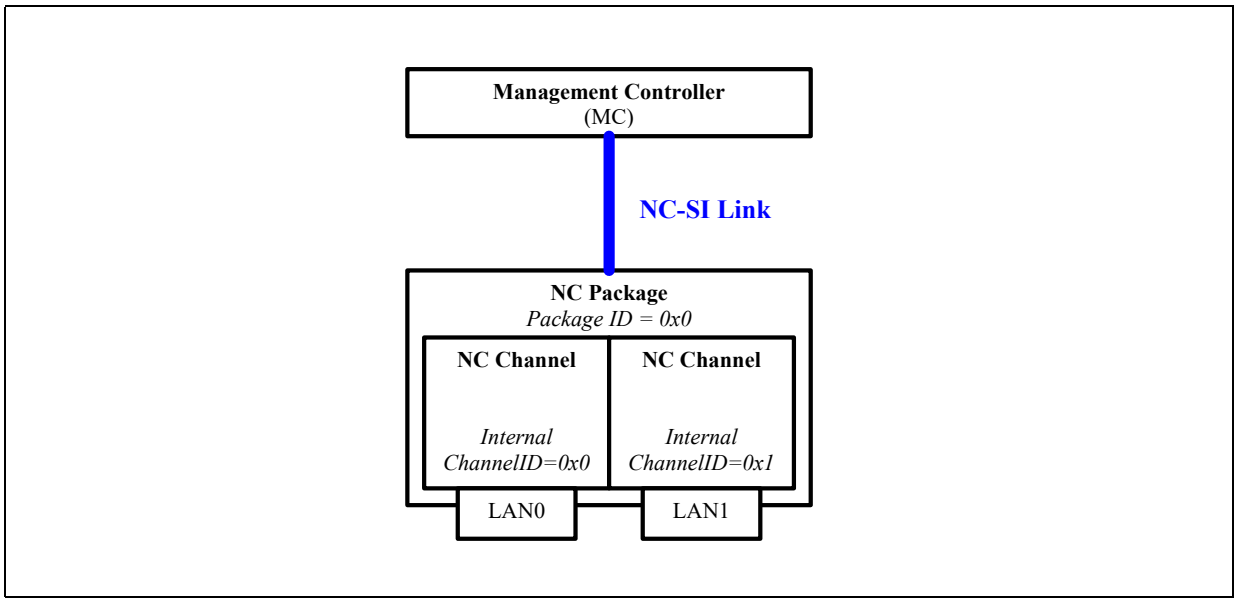
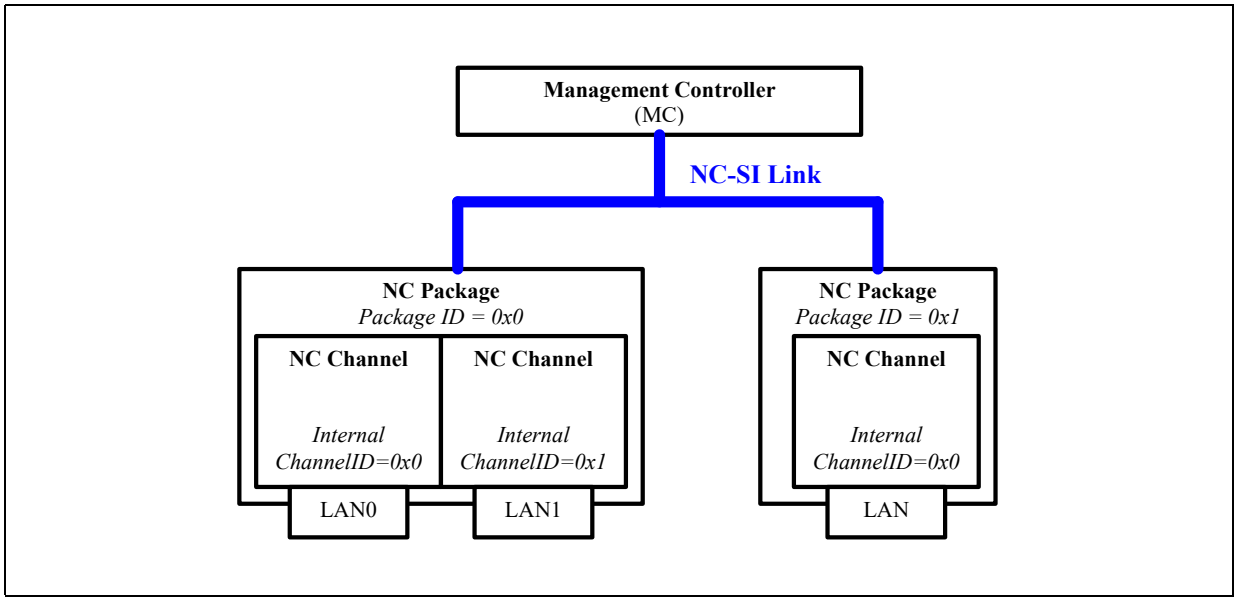


Figure 38-134 shows an example topology for a single MC and two NC packages. In this example, one NC package has two NC channels and the other has only one NC channel. Scenarios in which the NC-SI lines are shared by multiple NCs (Figure 38-134) mandate an arbitration mechanism.

The arbitration mechanism is described in Section 38.37.6.9.1.

Figure 38-134.Two NC Packages (Left, With Two NC Channels and Right, With One NC Channel)



38.37.6.1.3 Data Transport

Since NC-SI is based upon the RMII transport layer, data is transferred in the form of Ethernet frames.

NC-SI defines two types of transmitted frames:

1. Control frames:
   a. Configures and control the interface.
   b. Identified by a unique EtherType in their L2 header.
2. PT frames:
   a. Actual LAN pass-through frames transferred from/to the MC.
   b. Identified as not being a control frame.
   c. Attributed to a specific NC channel by their source MAC address (as configured in the NC by the MC).

38.37.6.1.4 Control Frames

NC-SI control frames are identified by a unique NC-SI EtherType (0x88F8).

Control frames are used in a single-threaded operation, meaning commands are generated only by the MC and can only be sent one at a time. Each command from the MC is followed by a single response from the NC (command-response flow), after which the MC is allowed to send a new command.

The only exception to the command-response flow is the Asynchronous Event Notification (AEN). These control frames are sent unsolicited from the NC to the MC.

AEN functionality by the NC must be disabled by default, until activated by the MC using the Enable AEN commands.

In order to be considered a valid command, a control frame must:

1. Comply with the NC-SI header format.
2. Be targeted to a valid channel in the package via the *Package ID* and *Channel ID* fields. For example, to target a NC channel with package ID of 0x2 and internal channel ID of 0x5, the MC must set the channel ID inside the control frame to 0x45. The channel ID is composed of three bits of package ID and five bits of internal channel ID.
3. Contain a correct payload checksum (if used).
4. Meet any other condition defined by NC-SI.

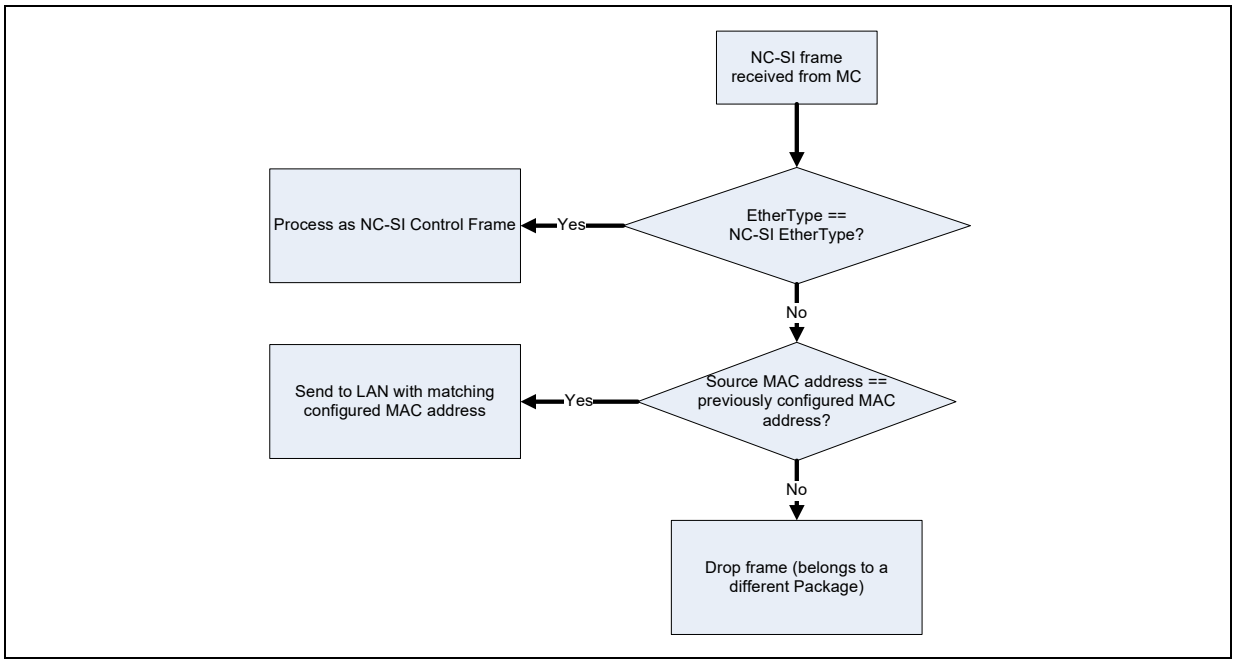
There are also commands (such as select package) targeted to the package as a whole. These commands must use an internal channel ID of 0x1F.

For details, refer to the NC-SI specification.

NC-SI Frames Receive Flow

Figure 38-135 shows the flow for frames received on the NC from the MC.

Figure 38-135.NC-SI Frames Receive Flow for the NC



38.37.6.2 NC-SI Standard Support

38.37.6.2.1 Supported Features

The 10 GbE controller supports all the mandatory features of the NC-SI specification (rev 1.0.1). Table 38-546 lists the supported commands.

Table 38-547 lists optional features supported and the level of support for partially supported commands.

Table 38-546.Supported NC-SI Commands

Command	Supported Over RMII	Supported Over MCTP With Pass Through	Supported Over MCTP Without Pass Through
Clear Initial State	Yes	Yes	Yes
Get Version ID	Yes	Yes	Yes
Get Parameters	Yes	Yes	Yes
Get Controller Packet Statistics	Yes, partially	Yes, partially	Yes, partially
Get Link Status[6]	Yes	Yes	Yes
Enable Channel	Yes	Yes	Yes
Disable Channel	Yes	Yes	Yes
Reset Channel	Yes	Yes	Yes
Enable VLAN	Yes[1,2]	Yes	No[3]
Disable VLAN	Yes	Yes	No-
Enable Broadcast Filter	Yes	Yes	No
Disable Broadcast Filter	Yes	Yes	No

Table 38-546.Supported NC-SI Commands

Command	Supported Over RMII	Supported Over MCTP With Pass Through	Supported Over MCTP Without Pass Through
Set MAC Address	Yes	Yes	No
Get NC-SI Statistics	Yes	Yes	Yes
Set NC-SI Flow Control	Yes	No	No
Set Link Command	Yes	Yes	Yes
Enable Global Multicast Filter	Yes	Yes	No
Disable Global Multicast Filter	Yes	Yes	No
Get Capabilities	Yes	Yes	Yes[4]
Set VLAN Filters	Yes	Yes	No
AEN Enable	Yes	Yes	Yes
Get NC-SI Pass-through Statistics	Yes, partially	Yes, partially	No
Select Package	Yes	Yes	Yes
Deselect Package	Yes	Yes	Yes
Enable Channel Network Tx	Yes	Yes	No
Disable Channel Network Tx	Yes	Yes	No
OEM Command[5]	Yes	Yes	Yes

Notes:

1. In cases that one of the LAN devices is assigned for the sole use of the manageability and its LAN PCIe function is disabled, using the NC-SI Set Link command while advertising multiple speeds and enabling auto-negotiation, results in the lowest possible speed chosen. To enable link of higher a speed, the MC should not advertise speeds that are below the desired link speed. When doing it, changing the power state of the LAN device has no effect and the link speed is not re-negotiated.
2. The 10 GbE controller does not support filtering of *User Priority/CFI* bits of VLAN.
3. In MCTP without PT mode, only control commands are supported and not PT traffic. Thus many of the regular NC-SI commands are not supported or are supported in a limited manner, only to enable control and status reporting for the device.
4. When PT is disabled, the Get Capabilities command does not expose all the filtering capabilities of the device.
5. See Section 38.37.6.3.2 for details.

6.Parallel Detection flag in this command response is not supported.

Table 38-547.Optional NC-SI features support

Feature	Implement	Details
AENs	Yes	The software device driver state AEN might be emitted up to one minute after actual driver change if the driver was taken down unexpectedly. When more than one function is associated with a channel, the driver status is enabled if at least one driver is up.
Get Controller Packet Statistics Command	Yes, partially	Supports the following counters[1]: 0-8,11-16, 21-36[2]. The statistics are not cleared between reads. ***Note:*** The packets counted by counter #35 (1523 - 9022 byte frames transmitted) are up to 9522 bytes and not 9022 as requested in the specification.
Get NC-SI Statistics	Yes	Support all the counters[3].
Get NC-SI Pass-through Statistics	Yes, partially	Support the following counters: 1, 6, 7.
VLAN Modes	Yes, partially	Support only modes 1, 3.
Buffering Capabilities	Yes	8 Kb.
MAC Address Filters	Yes	Supports 2 MAC addresses per port.
Channel Count	Yes	Supports 4 channels.

Table 38-547.Optional NC-SI features support

Feature	Implement	Details
VLAN Filters	Yes	Support 8 VLAN filters per port. Filtering is ignoring the *CFI* bit and the 802.1P priority bits.
Broadcast Filters	Yes	Support the following filters: ARP. DHCP. Net BIOS.
Multicast Filters	Yes	Supports the following filters: IPv6 neighbor advertisement. IPv6 router advertisement. DHCPv6 relay and server multicast.
Hardware Arbitration	Yes	Supports NC-SI hardware arbitration.

Notes:

1. *TCTL.EN* should be set to 1b to activate Tx-related counters and *RCTL.RXEN*, *PRT_MNG_MANC.RCV_EN* or *WUC.APME* should be set to enable Rx-related counters.
2. As described in the Get Controller Packet Statistics Counter Numbers table in NC-SI specification.
3. The 10 GbE controller does not increment the NC-SI control packets dropped counter when packets with checksum errors are dropped. In this case, only the NC-SI command checksum errors counter is updated.

38.37.6.2.2 AEN Handling

Asynchronous events might occur when the device is not allowed to send them. The following rules defines the behavior of the 10 GbE controller in these cases:

1. While the device is disabled, for each type of AEN only the last event is kept.
2. Outstanding AENs that occurred while a package was deselected is transmitted when a package is selected.
3. On a transition from channel disabled to channel enabled, all outstanding events are erased to prevent stale event notifications.
4. If the AEN becomes outdated before being sent (for example a link down, link up sequence occurring before the AEN is sent), then no AEN is sent.

Table 38-548.Get NC-SI PT Statistics Generation

Counter Number	Statistic Name	Source
1	Total PT Tx Packets Received	Sum of GLV_UPTCL, GLV_MPTCL and GLV_BPTCL for relevant VSI.
6	Total PT Rx Packets	Sum of GLV_UPRCL, GLV_MPRCL and GLV_BPRCT
7	Total PT Rx Packets Dropped	GLV_RDPC.

38.37.6.3 External Link Control via NC-SI

38.37.6.3.1 NC-SI Link State Control

In NC-SI mode, the device might dynamically change the PHY power mode according to the NC-SI channel state assuming no other functionality requires the PHY to be active (host or wake up).

The following algorithm is used to define if PHY activity is required:

- At initialization time, the PHY is required to be active only if the *EMP_LINK_ON* bit in *Common Firmware Parameters 2* NVM word is set.
- Once a channel is enabled via a Enable Channel NC-SI command, The PHY is powered up.

- If the channel is disabled via a Disable Channel command with the *ALD* bit set, the PHY is disabled.
- If the channel is disabled via a Reset Channel command, the PHY power state is set back to the initial value as define by the *EMP_LINK_ON* bit.

Note: Before a transition to D3 it is the responsibility of the software device driver to request the PHY to be active for wake-up activities.

38.37.6.3.2 Set Link Error Codes

The following rules are used to define the error code returned for a Set Link command in case an invalid configuration is requested:

1. Host driver check — If a host device driver is present, return a Command Specific Response (0x9) with a Set Link Host OS/Driver Conflict Reason (0x1).
2. Speed present check — If no speed is selected, return a General Reason Code for a failed command (0x1) with Parameter Is Invalid, Unsupported, or Out-of-Range Reason (0x2).
3. Parameter Validity:
   a. Auto-negotiation Parameter Validation — If auto-negotiation is requested and none of the selected parameters are valid for the device, return a General Reason Code for a failed command (0x1) with a Parameter Is Invalid, Unsupported, or Out-of-Range Reason (0x2).

Note: This means, for example, a command requesting 10 GbE on a 1 GbE device succeeds provided that the command requests at least one other supported speed.

For the 10 GbE controller, setting the auto-negotiation enabled field is ignored. For speed and connection types that only accept force mode, force mode is used. For modes that only support auto-negotiation, auto-negotiation is used to enforce a speed by only negotiating this speed. There are no modes supported by the 10 GbE controller that supports both auto-negotiation and force modes.

The same goes for an unsupported duplex setting (a device with no HD support accepts a command with both FD and HD set), and also for HD being requested with speeds of 1 GbE and higher as long as a speed below 1 GbE is also requested (and is supported in HD). The device simply ignores the unsupported parameters.

   a. Force mode parameter validation:
      - If more than one link speed is being forced, then return a General Reason Code for a failed command (0x1) and a Command Specific Reason with a Set Link Speed Conflict Error (0x0905).
      - If more than one duplex setting is being forced, then return a General Reason Code for a failed command (0x1) with Parameter Is Invalid, Unsupported, or Out-of-Range Reason (0x2).
      - If 1 GbE and above is requested with HD, then return a General Reason Code for a failed command (0x1) and a Command Specific Reason with Set Link Parameter Conflict Reason (0x0903).

4. Media type compatibility check — If current media type is not compatible for the requested link parameters, return a General Reason Code for a failed command (0x1) and a Command Specific Reason with Set Link Media Conflict Error (0x0902).
5. Power state compatibility check — If current power state does not allow for the requested link parameters, return a General Reason Code for a failed command (0x1) and a Command Specific Reason with Set Link Power Mode Conflict Reason (0x0904).

6. If for some reason the hardware cannot perform the flow required for the command, return a General Reason Code for a failed command (0x1) and a Command Specific Response (0x9) with Link Command Failed-Hardware Access Error (0x6).

38.37.6.4 NC-SI Mode — Intel Specific Commands

In addition to regular NC-SI commands, the following Intel vendor specific commands are supported. The purpose of these commands is to provide a means for the MC to access some of the Intel-specific features present in the 10 GbE controller.

38.37.6.4.1 Overview

The following features are available via the NC-SI OEM specific commands:

- Receive filters.
- Packet addition decision filters 0x0…0x4.
- Packet reduction decision filters 0x5…0x7.
- *PRT_MNG_MNGONLY* register (controls the forwarding of manageability packets to the host).
- Flex 128 filters.
- Flex TCP/UDP port filters 0x0...0x2.
- IPv4/IPv6 filters.
- Get system MAC address — This command enables the MC to retrieve the system MAC address used by the MC. This MAC address can be used for a shared MAC address mode.
- Keep PHY link up (*Veto* bit) enable/disable — This feature enables the MC to block PHY reset, which might cause session loss.
- TCO reset — Enables the MC to reset the 10 GbE controller.
- Checksum offloading — Offloads IP/UDP/TCP checksum checking from the MC.
- OS2BMC control commands.
- Firmware version commands.
- Shared MAC and shared IP commands.

These commands are designed to be compliant with their corresponding SMBus commands (if existing). All of the commands are based on a single DMTF defined NC-SI command, known as OEM Command. This command is as follows.

OEM Command (0x50)

The OEM command can be used by the MC to request the sideband interface to provide vendor-specific information. The Vendor Enterprise Number (VEN) is the unique MIB/SNMP private enterprise number assigned by IANA per organization. Vendors are free to define their own internal data structures in the vendor data fields.

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...	Intel Command Number	Optional Data		

	Bits	
...	...	
...	Optional Data	Padding To 32 Bits (0x00)
...	Checksum	

OEM Response (0xD0)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...27	Intel Command Number	Optional Return Data		
...	...			
...	Optional Return Data		Padding To 32 Bits (0x00)	
...	Checksum			

Note: Responses have no command-specific reason code, unless otherwise specified within the command.

Note: The commands/responses described as follows includes only the part up to the data. The padding and checksum are implied.

38.37.6.4.2 OEM Commands Summary

Table 38-549.OEM Specific Command Response Reason Codes

Response Code		Reason Code	
Value	**Description**	**Value**	**Description**
0x1	Command Failed	0x5081	Invalid Intel command number.
		0x5082	Invalid Intel command parameter number.
		0x5085	Internal network controller error.
		0x5086	Invalid vendor enterprise code.
		0x508D	Returned when one of the shared IP commands is received with an out of range resource (IP, port, binding) index.
		0x508E	Returned when a request to disable a port or an IP address used in a active binding is received.
		0x5090	Returned when a binding of a non enabled resource (MAC, VLAN, IP address, port) is required.
		0x5091	Returned when the *Set Port* command is received with an unsupported protocol.
		0x5092	Not is shared mode. Returned when shared mode commands are used while not in shared MAC/IP mode.
		0x008E	Returned when a request to disable a VLAN or a MAC address used in a active binding is received.

Table 38-550.OEM Commands Summary

Intel Command	Parameter	Command Name	Supported in MCTP Without PT
0x00	0x00	Set IP Filters Control	No
0x01	0x00	Get IP Filters Control	No
0x02	0x0F	Set Manageability Only	No
	0x10	Set Flexible 128 Filter Mask and Length	
	0x11	Set Flexible 128 Filter Data	
	0x63	Set Flex TCP/UDP Port Filters	
	0x64	Set Flex IPv4 Address Filters	
	0x65	Set Flex IPv6 Address Filters	
	0x67	Set EtherType Filter	
	0x68	Set Packet Addition Extended Filter	
	0x69	Set Special Filter Modifiers	
0x03	0x0F	Get Manageability Only	No
	0x10	Get Flexible 128 Filter Mask and Length	
	0x11	Get Flexible 128 Filter Data	
	0x63	Get Flex TCP/UDP Port Filters	
	0x64	Get Flex IPv4 Address Filters	
	0x65	Get Flex IPv6 Address Filters	
	0x67	Get EtherType Filter	
	0x68	Get Packet Addition Extended Filter	
	0x69	Get Special Filter Modifiers	
0x04	0x10	Set Extended Unicast Packet Reduction	No
	0x11	Set Extended Multicast Packet Reduction	
	0x12	Set Extended Broadcast Packet Reduction	
0x5	0x10	Get Extended Unicast Packet Reduction	No
	0x11	Get Extended Multicast Packet Reduction	
	0x12	Get Extended Broadcast Packet Reduction	
0x06	N/A	Get System MAC Address	Yes
0x20	N/A	Set Intel Management Control	No
0x21	N/A	Get Intel Management Control	No
0x22	N/A	TCO Reset	Yes
0x23	N/A	Enable IP/UDP/TCP Checksum Offloading	No
0x24	N/A	Disable IP/UDP/TCP Checksum Offloading	No

Table 38-550.OEM Commands Summary

Intel Command	Parameter	Command Name	Supported in MCTP Without PT
0x25	0x0	Set IP Address	No
	0x1	Get IP Address	
	0x2	Set Port	
	0x3	Get Port	
	0x4	Enable Unicast Infrastructure Filter	
	0x5	Get Shared IP Capabilities Command	
	0x6	Shared IP Enable Broadcast Filtering	
	0x7	Shared IP Enable Global Multicast Filtering	
	0x8	Get Shared IP Parameters	
	0x9	Set Binding	
	0xA	Get Binding	
	0xB	Set Shared Mode	
0x40	0x01	Enable OS2BMC Flow	No
	0x02	Enable Network to BMC Flow	
	0x03	Enable Both Network-to-BMC and Host-to-BMC Flow	
0x40	0x04	Set BMC IP Address	No
0x41	N/A	Get OS2BMC Parameters	No
0x48	0x1	Get Controller Information	Yes

Note: All the commands are supported both over RMII NC-SI and over MCTP.

38.37.6.4.3 Set Intel Filters Control — IP Filters Control Command (Intel Command 0x00, Filter Control Index 0x00)

This command controls different aspects of the Intel filters.

Set Intel Filters Control — IP Filters Control Command

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...23	0x00	0x00	IP Filters Control (3-2)	
24...27	IP Filters Control (1-0)			

Where IP filters control has the following format.

Bit #	Name	Description	Default Value
0	IPv4/IPv6 Mode	IPv6 (0b) = There are zero IPv4 filters and four IPv6 filters. IPv4 (1b) = There are four IPv4 filters and four IPv6 filters.	1b
1...31	Reserved		

Note: This command is kept for compatibility with other projects and has no effect in Product Name.

Set Intel Filters Control — IP Filters Control Response

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...27	0x00	0x00		

38.37.6.4.4 Get Intel Filters Control Commands (Intel Command 0x01)

Get Intel Filters Control — IP Filters Control Command (Intel Command 0x01, Filter Control Index 0x00)

This command controls different aspects of the Intel filters.

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...21	0x01	0x00		

Get Intel Filters Control — IP Filters Control Response (Intel Command 0x01, Filter Control Index 0x00)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...27	0x01	0x00	IP Filters Control (3-2)	
28...29	IP Filters Control (1-0)			

Note: This command is kept for compatibility with other projects and returns always 0x1 in the 10 GbE controller.

38.37.6.4.5 Set Intel Filters Formats

Set Intel Filters Command (Intel Command 0x02)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...21	0x02	Parameter Number	Filters Data (optional)	

Set Intel Filters Response (Intel Command 0x02)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...	0x02	Filter Control Index	Return Data (Optional)	

Set Intel Filters — Manageability Only Command (Intel Command 0x02, Filter Parameter 0x0F)

This command sets the PRT_MNG_MNGONLY register. The PRT_MNG_MNGONLY register controls whether PT packets destined to the MC are not forwarded to the Host operating system. The PRT_MNG_MNGONLY register is listed in Table 38-523.

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...23	0x02	0x0F	Manageability Only (3-2)	
24...25	Manageability Only (1-0)			

Set Intel Filters — Manageability Only Response (Intel Command 0x02, Filter Parameter 0x0F)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...25	0x02	0x0F		

Set Intel Filters — Flex Filter Enable Mask and Length Command (Intel Command 0x02, Filter Parameter 0x10)

The following command sets the Intel flex filters mask and length.

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...23	0x02	0x10	Mask Byte 1	Mask Byte 2
24...27	...	...	...	...
28...31	...	...	...	...
32...35	...	...	...	...
36...37	Mask Byte 15	Mask Byte 16	Reserved	Reserved
38	Length			

Set Intel Filters — Flex Filter Enable Mask and Length Response (Intel Command 0x02, Filter Parameter 0x10)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...25	0x02	0x10		

Set Intel Filters — Flex Filter Data Command (Intel Command 0x02, Filter Parameter 0x11)

Table 38-551.Filter data group

Code	Bytes Programmed	Filter Data Length
0x0	Bytes 0-29	1 - 30
0x1	Bytes 30-59	1 - 30
0x2	Bytes 60-89	1 - 30
0x3	Bytes 90-119	1 - 30
0x4	Bytes 120-127	1 - 8

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...	0x02	0x11	Filter Data Group	Filter Data 1
	...	Filter Data N		

Note: Using this command to configure the filters data must be done after the flex filter mask command is issued and the mask is set.

Set Intel Filters — Flex Filter Data Response (Intel Command 0x02, Filter Parameter 0x11)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...25	0x02	0x11		

Set Intel Filters — Packet Addition Decision Filter Command (Intel Command 0x02, Filter Parameter 0x61)

This command is no longer supported. Use the Section "Set Intel Filters - Packet Addition Extended Decision Filter Command (Intel Command 0x02, Filter Parameter 0x68)" instead.

Set Intel Filters — Flex TCP/UDP Port Filter Command

(Intel Command 0x02, Filter Parameter 0x63)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...23	0x02	0x63	Port Filter Index	TCP/UDP Port MSB
24..27	TCP/UDP Port LSB	Port Flags		

Filter index range: 0x0...0xA.

Port flags are as follows:

- Bit 0: Match UDP ports
- Bit 1: Match TCP ports
- Bit 2: Match destination port (0) or source port (1).
- Bit 7:3: Reserved

If flags are not present (payload length = 9), the match is done on TCP and UDP destination ports (legacy behavior).

If the filter index is larger than 10, a command failed response code is returned with Invalid Intel Parameter Number reason (0x5082).

Set Intel Filters — Flex TCP/UDP Port Filter Response

(Intel Command 0x02, Filter Parameter 0x63)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			

	Bits			
24...25	0x02	0x63		

Set Intel Filters — Ipv4 Filter Command (Intel Command 0x02, Filter Parameter 0x64)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...23	0x02	0x64	IP Filter Index	IPv4 Address (3)
24...26	IPv4 Address (2-0)			

Filter index range: 0x0...0x3.

Set Intel Filters — Ipv4 Filter Response (Intel Command 0x02, Filter Parameter 0x64)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...25	0x02	0x64		

If the IP filter index is larger than three, a command failed response code is returned Invalid Intel Parameter Number reason (0x5082).

Set Intel Filters — Ipv6 Filter Command (Intel Command 0x02, Filter Parameter 0x65)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...23	0x02	0x65	IP Filter Index	...IPv6 Address (MSB, Byte 15)
24...27	...	...	...	...
28...31	...	...	...	...
32...35	...	...	...	...
36...37	...		IPv6 Address (LSB, Byte 0)	

Note: The filters index range can vary according to the IPv4/IPv6 mode setting in the Filters Control command.

IPv4 Mode: Filter index range: 0x1...0x3.

IPv6 Mode: Filter index range: 0x0...0x3.

Set Intel Filters — Ipv6 Filter Response (Intel Command 0x02, Filter Parameter 0x65)

	Bits			
Bytes	**31:24**	**23:16**	**15:08**	**07:00**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...25	0x02	0x65		

If the IP filter index is larger than three, a command failed Response Code is returned, Invalid Intel Parameter Number reason (0x5082).

Set Intel Filters - EtherType Filter Command (Intel Command 0x02, Filter Parameter 0x67)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...23	0x02	0x67	EtherType Filter Index	EtherType Filter MSB
24...27	...	...	EtherType Filter LSB	

Where the EtherType filter has the format previously described.

38.37.6.4.6 Set Intel Filters - EtherType Filter Response (Intel Command 0x02, Filter Parameter 0x67)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...25	0x02	0x67		

If the EtherType filter index is different than two or three, a command failed Response Code is returned Invalid Intel Parameter Number reason (0x5082).

Set Intel Filters - Packet Addition Extended Decision Filter Command (Intel Command 0x02, Filter Parameter 0x68)

See Figure 38-129 for description of the decision filters structure.

The command must overwrite any previously stored value. The value set is not checked.

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...23	0x02	0x68	Extended Decision Filter Index	Extended Decision Filter 1 MSB
24...27	...	...	Extended Decision Filter 1 LSB	Extended Decision Filter 0 MSB
28...30	...	...	Extended Decision Filter 0 LSB	

Extended decision filter index range: 0...4

Filter 0: See Table 38-552.

Filter 1: See Table 38-553.

Table 38-552.Filter Values

Bit #	Name	Description
3:0	Unicast (AND)	If set, packets must match unicast filter 0 to 3, respectively.
4	Broadcast (AND)	If set, packets must match the broadcast filter.
12:5	VLAN (AND)	If set, packets must match VLAN filter 0 to 7, respectively.
16:13	IPv4 Address (AND)	If set, packets must match IPv4 filter 0 to 3, respectively
20:17	IPv6 Address (AND)	If set, packets must match IPv4 filter 0 to 3, respectively
24:21	Unicast (OR)	If set, packets can pass if match unicast filter 0 to 3, respectively or a different OR filter.
25	Broadcast (OR)	If set, packets can pass if match the broadcast filter or a different OR filter.
26	Multicast (AND)	If set, packets must match the multicast filter.
27	ARP Request (OR)	If set, packets can pass if match the ARP request filter or a different OR filter.
28	ARP Response (OR)	If set, packets can pass if match the ARP response filter or a different OR filter.
29	Neighbor Discovery - 134 (OR)	If set, packets can pass if match the neighbor discovery filter (type134 - router advertisement) or a different OR filter.
30	Port 0x298 (OR)	If set, packets can pass if match a fixed TCP/UDP port 0x298 filter or a different OR filter.
31	Port 0x26F (OR)	If set, packets can pass if match a fixed TCP/UDP port 0x26F filter or a different OR filter.

Table 38-553. Extended Filter 1 Values (Sheet 1 of 2)

Bit #	Name	Description
3:0	EtherType 0 -3 (AND)	If set, packets must match the EtherType filter 0 to 3, respectively.
7:4	EtherType 0 -3 (OR)	If set, packets must match the EtherType filter 0 to 3, respectively or a different OR filter.
18:8	Flex Port 10:0 (OR)	If set, packets can pass if match the TCP/UDP port filter 10:0.
19	DHCPv6 (OR)	If set, packets can pass if match the DHCPv6 port (0x0223).
20	DHCP Client (OR)	If set, packets can pass if match the DHCP server port (0x0043).
21	DHCP Server (OR)	If set, packets can pass if match the DHCP client port (0x0044).
22	NetBIOS Name Service (OR)	If set, packets can pass if match the NetBIOS name service port (0x0089).
23	NetBIOS Datagram Service (OR)	If set, packets can pass if match the NetBIOS datagram service port (0x008A).

Table 38-553. Extended Filter 1 Values (Sheet 2 of 2)

Bit #	Name	Description
24	Flex TCO (OR)	If set, packets can pass if match the flex 128 TCO filter.
25	Neighbor Discovery - 135 (OR)	If set, packets must also match the neighbor discovery filter (type135 - neighbor solicitation). or a different OR filter.
26	Neighbor Discovery - 136 (OR)	If set, packets must also match the neighbor discovery filter (type136 - neighbor advertisement) or a different OR filter.
27	Neighbor Discovery - 137 (OR)	If set, packets must also match the neighbor discovery filter (type137 - Redirect) or a different OR filter.
28	ICMPv4 (OR)	Controls the inclusion of ICMPv4 filtering in the manageability filter decision (OR section).
29	MLD	If set, packets must also match one of the MLD ICMPv6 types or a different OR filter.
31:30	Reserved	Reserved

Set Intel Filters – Packet Addition Extended Decision Filter Response (Intel Command 0x02, Filter Parameter 0x68)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...25	0x02	0x68		

If the extended decision filter index is larger than five, a command failed Response Code is returned Invalid Intel Parameter Number reason (0x5082).

Set Intel Filters - Special Modifier Command (Intel Command 0x02, Filter Parameter 0x69)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...23	0x02	0x69	Special Modifier Register MSB	
24...27	Special Modifier Register LSB		Padding	

Where the special modifier filter has the format previously described.

Set Intel Filters - Special Modifier Response (Intel Command 0x02, Filter Parameter 0x69)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...25	0x02	0x69		

Get Intel Filters Formats

Get Intel Filters Command (Intel Command 0x03)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...21	0x03	Parameter Number		

Get Intel Filters Response (Intel Command 0x03)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...25	0x03	Parameter Number	Optional Return Data	

Get Intel Filters — Manageability Only Command (Intel Command 0x03, Filter Parameter 0x0F)

This command retrieves the PRT_MNG_MNGONLY register. The PRT_MNG_MNGONLY register controls whether PT packets destined to the MC are also be forwarded to the host operating system.

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...21	0x03	0x0F		

Get Intel Filters — Manageability Only Response (Intel Command 0x03, Filter Parameter 0x0F)

The PRT_MNG_MNGONLY register structure is listed in Table 38-523.

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...27	0x03	0x0F	Manageability to Host (3-2)	
28...29	Manageability to Host (1-0)			

Get Intel Filters — Flex Filter 0 Enable Mask And Length Command (Intel Command 0x03, Filter Parameter 0x10)

The following command retrieves the Intel flex filters mask and length. See Section 38.37.3.3.6 for details of the values returned by this command.

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...21	0x03	0x10		

Get Intel Filters — Flex Filter 0 Enable Mask And Length Response (Intel Command 0x03, Filter Parameter 0x10)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...27	0x03	0x10	Mask Byte 1	Mask Byte 2
28...31	...	...	...	...
32...35	...	...	...	...
36...39	...	...	...	...
40...43	...	Mask Byte 16	Reserved	Reserved
44	Flexible Filter Length			

Get Intel Filters — Flex Filter 0 Data Command (Intel Command 0x03, Filter Parameter 0x11)

The following command retrieves the Intel flex filters data.

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...21	0x03	0x11	Filter Data Group 0...4	

The filter data group parameter defines which bytes of the flex filter are returned by this command:

Table 38-554.Filter Data Group

Code	Bytes Returned
0x0	bytes 0-29
0x1	bytes 30-59
0x2	bytes 60-89
0x3	bytes 90-119
0x4	bytes 120-127

Get Intel Filters — Flex Filter 0 Data Response (Intel Command 0x03, Filter Parameter 0x11)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...	0x03	0x11	Filter Group Number	Filter Data 1
	...	Filter Data N		

Get Intel Filters — Packet Addition Decision Filter Command (Intel Command 0x03, Filter Parameter 0x61)

This command is no longer supported. Use the Section "Get Intel Filters – Packet Addition Extended Decision Filter Command (Intel Command 0x03, Filter Parameter 0x68)" instead.

Get Intel Filters — Flex TCP/UDP Port Filter Command (Intel Command 0x03, Filter Parameter 0x63)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...22	0x03	0x63	TCP/UDP Filter Index	

Filter index range: 0x0...0x2.

Get Intel Filters — Flex TCP/UDP Port Filter Response (Intel Command 0x03, Filter Parameter 0x63)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...27	0x03	0x63	TCP/UDP Filter Index	TCP/UDP Port (1)
28..29	TCP/UDP Port (0)	Port flags		

Filter index range: 0x0...0x2.

Get Intel Filters — Ipv4 Filter Command (Intel Command 0x03, Filter Parameter 0x64)

	Bits			
Bytes	**31...24**	**23...16**	**15...08**	**07...00**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...22	0x03	0x64	IPv4 Filter Index	

Note: The filters index range can vary according to the IPv4/IPv6 mode setting in the Filters Control command.

IPv4 Mode: Filter index range: 0x0...0x3.

IPv6 Mode: This command should not be used in IPv6 mode.

Get Intel Filters — Ipv4 Filter Response (Intel Command 0x03, Filter Parameter 0x64)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...27	0x03	0x64	IPv4 Filter Index	IPv4 Address (3)
28...29	IPv4 Address (2-0)			

Get Intel Filters — Ipv6 Filter Command (Intel Command 0x03, Filter Parameter 0x65)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...22	0x03	0x65	IPv6 Filter Index	

Note: The filters index range can vary according to the IPv4/IPv6 mode setting in the Filters Control command.

IPv4 Mode: Filter index range: 0x0...0x2.

IPv6 Mode: Filter index range: 0x0...0x3.

Get Intel Filters — Ipv6 Filter Response Intel Command 0x03, Filter Parameter 0x65)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...27	0x03	0x65	IPv6 Filter Index	IPv6 Address (MSB, Byte 16)
28...31	...	...	...	...
32...35	...	...	...	...
36...39	...	...	...	...
40...42	...	...	IPv6 Address (LSB, Byte 0)	

Get Intel Filters - EtherType Filter Command (Intel Command 0x03, Filter Parameter 0x67)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...22	0x03	0x67	EtherType Filter Index	

Valid indices: 0...3

Get Intel Filters - EtherType Filter Response (Intel Command 0x03, Filter Parameter 0x67)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...27	0x03	0x67	EtherType Filter Index	EtherType Filter MSB
28...30	..	..	EtherType Filter LSB	

If the EtherType filter index is larger than three, a command failed Response Code is returned Invalid Intel Parameter Number reason (0x5082).

Get Intel Filters – Packet Addition Extended Decision Filter Command (Intel Command 0x03, Filter Parameter 0x68)

This command enables the MC to retrieve the extended decision filter.

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...22	0x03	0x68	Extended Decision Filter Index	

Get Intel Filters – Packet Addition Extended Decision Filter Response (Intel Command 0x03, Filter Parameter 0x68)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...27	0x03	0x68	Decision Filter Index	Decision Filter 1 MSB
28...31	..	..	Decision Filter 1 LSB	Decision Filter 0 MSB
32...34	..	..	Decision Filter 0 LSB	

Where decision filter 0 and decision filter 1 have the structure as detailed in the respective Set commands.

If the extended decision filter index is larger than 4, a command failed Response Code is returned Invalid Intel Parameter Number reason (0x5082).

Get Intel Filters – Special Modifier Command (Intel Command 0x03, Filter Parameter 0x69)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...23	0x03	0x69	Padding	

Where the special modifier filter has the format previously described.

Get Intel Filters - Special Modifier Response (Intel Command 0x02, Filter Parameter 0x69)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...27	0x03	0x69	Special Modifier Register MSB	
28...29	Special Modifier Register LSB		Padding	

38.37.6.4.7 Set Intel Packet Reduction Filters Formats

The non-extended commands are obsolete. The extended commands should be used instead.

Set Intel Packet Reduction Filters Command (Intel Command 0x04)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...23	0x04	Packet Reduction Index	Packet Reduction Data...	

Note: It is advised that the MC only use the extended packet reduction commands.

The *Packet Reduction* data field has the following structure:

Table 38-555.Packet Reduction Field Description

Bit #	Name	Description
12:0	Reserved	Reserved
16:13	IPv4 Address (AND)	If set, packets must match IPv4 filter 0 to 3, respectively.
20:17	IPv6 Address (AND)	If set, packets must match IPv4 filter 0 to 3, respectively.
27:21	Reserved	Reserved.
28	ARP Response (OR)	If set, packets can pass if match the ARP response filter or a different OR filter.
29	Reserved	Reserved.
30	Port 0x298	If set, packets can pass if match a fixed TCP/UDP port 0x298 filter.
31	Port 0x26F	If set, packets can pass if match a fixed TCP/UDP port 0x26F filter.

Table 38-556.Extended Packet Reduction Field Description (Sheet 1 of 2)

Bit #	Name	Description
3:0	EtherType 0 -3 (AND)	If set, packets must match the EtherType filter 0 to 3, respectively.
7:4	EtherType 0-3 (OR)	If set, packets can pass if match the EtherType filter 0 to 3, respectively.
15:12	Reserved	Reserved.
8:18	Flex port 10:0 (OR)	If set, packets can pass if match the TCP/UDP Port filter 10:0.
23:19	Reserved	Reserved.
24	Flex TCO (OR)	If set, packets can pass if match the Flex 128 TCO filter.
27:25	Reserved	Reserved.

Table 38-556.Extended Packet Reduction Field Description (Sheet 2 of 2)

Bit #	Name	Description
28	ICMPv4	Is set, ICMPv4 packets can pass.
31:29	Reserved	Reserved.

The filtering is divided into two decisions:

- Bit 20:13 in Table 38-555 and bits 3:2 in Table 38-556 works in an AND manner; it must be true in order for a packet to pass (if was set).

Bits 28 in Table 38-555 and bits 24:10 in Table 38-556 work in an OR manner; at least one of them must be true for a packet to pass (if any were set).

Set Intel Packet Reduction Filters Response (Intel Command 0x04)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...	0x04	Packet Reduction Index		

Set Unicast Extended Packet Reduction Command (Intel Command 0x04, Reduction Filter Index 0x10)

The command has the following format:

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00..15	NC-SI Header			
16..19	Manufacturer ID (Intel 0x157)			
20..23	0x04	0x10	Extended Unicast Reduction Filter MSB	..
24..27	..	Extended Unicast Reduction Filter LSB	Unicast Reduction Filter MSB	..
28..29	..	Unicast Reduction Filter LSB		

The command overwrites any previously stored value.

Note: See Table 38-555 and Table 38-556 for description of the unicast extended packet reduction format.

Set Unicast Extended Packet Reduction Response (Intel Command 0x04, Reduction Filter Index 0x10)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..25	0x04	0x10		

Set Multicast Extended Packet Reduction Command (Intel Command 0x04, Reduction Filter Index 0x11)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00..15	NC-SI Header			
16..19	Manufacturer ID (Intel 0x157)			
20..23	0x04	0x11	Extended Multicast Reduction Filter MSB	..
24..27	..	Extended Multicast Reduction Filter LSB	Multicast Reduction Filter MSB	..
28..29	..	Multicast Reduction Filter LSB		

Note: See Table 38-555 and Table 38-556 for description of the multicast extended packet reduction format.

The command overwrites any previously stored value.

Set Multicast Extended Packet Reduction Response (Intel Command 0x04, Reduction Filter Index 0x11)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..25	0x04	0x11		

Set Broadcast Extended Packet Reduction Command (Intel Command 0x04, Reduction Filter Index 0x12)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00..15	NC-SI Header			
16..19	Manufacturer ID (Intel 0x157)			
20..23	0x04	0x12	Extended Broadcast Reduction Filter MSB	..
24..27	..	Extended Broadcast Reduction Filter LSB	Broadcast Reduction Filter MSB	..
28..29	..	Broadcast Reduction Filter LSB		

Note: See Table 38-555 and Table 38-556 for description of the broadcast extended packet reduction format.

The command overwrites any previously stored value.

Set Broadcast Extended Packet Reduction Response (Intel Command 0x04, Reduction Filter Index 0x12)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..25	0x04	0x12		

38.37.6.4.8 Get Intel Packet Reduction Filters Formats

Note: The non-extended commands are obsolete. Use the extended commands: Section "Get Unicast Extended Packet Reduction Command (Intel Command 0x05, Reduction Filter Index 0x10)" to Section "Get Broadcast Extended Packet Reduction Response (Intel Command 0x05, Reduction Filter Index 0x12)" instead.

Get Unicast Extended Packet Reduction Command (Intel Command 0x05, Reduction Filter Index 0x10)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...21	0x05	0x10		

Get Unicast Extended Packet Reduction Response (Intel Command 0x05, Reduction Filter Index 0x10)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...27	0x05	0x00	Extended Unicast Packet Reduction (3-2)	
28...29	Extended Unicast Packet Reduction (1-0)		Unicast Packet Reduction (3-2)	
30...31	Unicast Packet Reduction (1-0)			

Get Multicast Extended Packet Reduction Command (Intel Command 0x05, Reduction Filter Index 0x11)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...21	0x05	0x11		

Get Multicast Extended Packet Reduction Response (Intel Command 0x05, Reduction Filter Index 0x11)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...27	0x05	0x11	Extended Multicast Packet Reduction (3-2)	
28...29	Extended Multicast Packet Reduction (1-0)		Multicast Packet Reduction (3-2)	
30...31	Multicast Packet Reduction (1-0)			

Get Broadcast Extended Packet Reduction Command (Intel Command 0x05, Reduction Filter Index 0x12)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...21	0x05	0x12		

Get Broadcast Extended Packet Reduction Response (Intel Command 0x05, Reduction Filter Index 0x12)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...27	0x05	0x12	Extended Broadcast Packet Reduction (3-2)	
28...29	Extended Broadcast Packet Reduction (1-0)		Broadcast Packet Reduction (3-2)	
30...31	Broadcast Packet Reduction (1-0)			

38.37.6.4.9 System MAC Address

Get System MAC Address Command (Intel Command 0x06)

In order to support a system configuration that requires the NC to hold the MAC address for the MC (such as shared MAC address mode), the following command is provided to enable the MC to query the NC for a valid MAC address.

The NC must return the system MAC addresses. The MC should use the returned MAC addressing as a shared MAC address by setting it using the Set MAC Address command as defined in NC-SI 1.0.

When a single function is defined on the port, it returns the LAN MAC address of this function as read from the PF allocations NVM section or from the alternate RAM or as set by the Manage MAC address Write AQ command. When more than one function is defined on the port, it returns the address of the lowest defined function on this port.

It is also recommended that the MC use packet reduction and the Manageability-to-Host command to set the proper filtering method.

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20	0x06			

Get System MAC Address Response (Intel Command 0x06)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...27	0x06	MAC Address		
28...30	MAC Address			

38.37.6.4.10 Set Intel Management Control Formats

Set Intel Management Control Command (Intel Command 0x20)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...22	0x20	0x00	Intel Management Control 1	

Where Intel Management Control 1 is as follows:

Bit #	Default value	Description
0	0b	Enable Critical Session Mode (Keep PHY Link Up and Veto Bit). 0b = Disabled. 1b = Enabled. When critical session mode is enabled, the following behaviors are disabled: • The PHY is not reset on PE_RST# and PCIe resets (in-band and link drop). Other reset events are not affected — Internal_Power_On_Reset, device disable, Force TCO, and PHY reset by software. • The PHY does not change its power state. As a result, link speed does not change. • The device does not initiate configuration of the PHY to avoid losing link.
7:1	0x0	Reserved.

Set Intel Management Control Response (Intel Command 0x20)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...25	0x20	0x00		

38.37.6.4.11 Get Intel Management Control Formats

Get Intel Management Control Command (Intel Command 0x21)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...21	0x21	0x00		

Where Intel Management Control 1 is as described in Section , "Set Intel Management Control Response (Intel Command 0x20)".

Get Intel Management Control Response (Intel Command 0x21)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...26	0x21	0x00	Intel Management Control 1	

38.37.6.4.12 TCO Reset

Depending on the bit set in the TCO mode field this command causes the 10 GbE controller to perform either:

1. TCO reset — If force TCO reset is enabled in the NVM. The force TCO reset clears the data path (Rx/Tx) of the 10 GbE controller to enable the MC to transmit/receive packets through the 10 GbE controller.
   - If the MC has detected that the operating system is hung and has blocked the Rx/Tx path, the force TCO reset clears the data-path (Rx/Tx) of the NC to enable the MC to transmit/receive packets through the NC.
   - When this command is issued to a channel in a package, it applies only to the specific channel.
   - After successfully performing the command, the NC considers the Force TCO command as an indication that the operating system is hung and clears the internal driver up indication. If TCO reset is disabled in the NVM, the 10 GbE controller does not reset the data path and notifies the MC on successful completion.
2. TCO isolate — If TCO isolate is enabled in the NVM. The TCO Isolate command disables PCIe write operations to the LAN port.
   - If TCO isolate is disabled in NVM, the 10 GbE controller does not execute the command but sends a response to the MC with successful completion.
   - Following a TCO isolate, management sets *EMP_TCO_ISOLATE.EMP_TCO_ISOLATE* to 1b for all PFs associated with the port on which this command is received.
3. Firmware reset — This command causes re-initialization of all the manageability functions and re-loads of manageability related NVM words (such as firmware patch code).
   - When the MC loads a new management related NVM image (like a firmware patch) the Firmware Reset command loads the management related NVM information without the need to power down the system.
   - This command is issued to the package and affects all channels. After the firmware reset, the FW Semaphore register (FWSM) is re-initialized.

Note: Applying this command resets the entire device and also has an effect on TCO reset. TCO isolate affects only the channel (port) that the command was issued to. Force TCO resets the entire device (all channels in the package).

Following firmware reset, the MC needs to re-initialize all ports. A firmware reset causes a global reset of the entire device (GLOBR).

Note: Only one of the fields should be set in a given command. Setting more than one field might yield unexpected results.

Perform Intel TCO Reset Command (Intel Command 0x22)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20	0x22	TCO Mode		

Where TCO mode is:

Field	Bit(s)	Description
DO_TCO_RST	0	Do TCO Reset. 0b = Do nothing. 1b = Perform TCO reset.
DO_TCO_ISOLATE[1]	1	Do TCO Isolate. 0b = Enable PCIe write access to LAN port. 1b = Isolate Host PCIe write operation to the port ***Note:*** Should be used for debug only. ***Note:*** The TCO Isolate do not impact MCTP traffic ***Note:*** When isolate is set, the OS2BMC flow is also disabled.
RESET_MGMT	2	Reset Manageability; Re-load Manageability NVM Words. 0b = Do nothing. 1b = Issue firmware reset to manageability. Setting this bit generates a one-time firmware reset. Following the reset, management related data from the NVM is loaded. ***Note:*** A reset of the internal firmware causes a reset of the entire device.
Reserved	7:3	Reserved (set to 0x00).

Note: For compatibility, the TCO Reset command without the TCO mode parameter is accepted (TCO reset is done).

Notes:

1. TCO isolate host write operation enabled in the NVM.

Perform Intel TCO Reset Response (Intel Command 0x22)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...26	0x22			

38.37.6.4.13 Checksum Offloading

This command enables the checksum offloading filters in the NC.

When enabled, these filters block any packets that did not pass IP, UDP or TCP checksum from being forwarded to the MC.

Enable Checksum Offloading Command (Intel Command 0x23)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20	0x23			

Enable Checksum Offloading Response (Intel Command 0x23)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...26	0x23			

Disable Checksum Offloading Command (Intel Command 0x24)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20	0x24			

Disable Checksum Offloading Response (Intel Command 0x24)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...26	0x24			

38.37.6.4.14 Shared MAC and Shared IP Support Commands (Intel Command 0x25)

To meet the requirements introduced by sharing IP addresses, modifications and additions to the NC-SI command set are required. These changes include the new commands in this section and the modifications described in Section 38.37.3.6.

Note: All indexes in this command set starts at one to match the NC-SI methodology.

Set IP Address Command (Intel Command 0x25, Index = 0x0)

The *Set IP Address* command is used by the MC to communicate its IP address to a NC. The format of a *Set IP Address* command packet is listed in Table 38-557.

If at least one IP address filter is enabled, only unicast packets that match one of the enabled filters are forwarded through the NC-SI interface. Otherwise, the IP address is ignored in the unicast filtering process.

This command does not impact the forwarding results. It is used as a preliminary stage to the *Set Binding* command.

Table 38-557.Set IP Address Command Packet Format

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20..23	0x25	0x0	Reserved	
24..27	Management Controller IP Address			
28..31				
32..35				
36..39				
40..43	Reserved		IP Address Number	Set IP Flags
44..47	Checksum			

- MC IP Address — An IP address that is used by the MC. If the *IP Version* bit of the *Flags* field is 0b (IPv4), this is a 4-byte unicast IPv4 address in network byte order. In this case, the address occupies bytes 24-27 of the packet, and bytes 28-39 are ignored. If the *IP Version* bit of the *Flags* field is 1b (IPv6), this is a 16-byte unicast IPv6 address in network byte order. In this case, the address occupies the full field (bytes 24-39 of the packet).
- IP Address Number — Indicates which IP address filter is configured by the command. The value can relate to one of three pools of filters according to Table 38-558.

Table 38-558.IP filters Pools

Set IP Flag.IP Version	Set IP Flag.Mixed Index	Pool to Use	Allowed Values
0	0	IPv4	One to the number of IPv4 only addresses.
1	0	IPv6	One to the number of IPv6 only addresses.
X (0/1)	1	Mixed	One to the number of mixed IP addresses.

Note: The values shown in the allowed values column refers to the Get Shared IP Capabilities Response.

- Table 38-559 lists the bits fields in the *Set IP Flags* field.

Table 38-559.Set IP Flag Field

Bit Position	Field Description	Value Description
0	Enable	0b = Disable the filter. 1b = Enable the filter.
1	IP Version	0b = IPv4. 1b = IPv6.
2	Mixed Index	0b = Index relates to the IPv4 or IPv6 only IP filter sets according to the IP version field. 1b = Index relates to the mixed IP filter set.
3	MAC Based IP	This flags define if the Ipv6 address is derived from a MAC address and thus only the 24 LSB should be used for the comparison. This flag is relevant only if the IP version = IPv6. 0b = Filter according to the full 128 bits of IPv6 address. 1b = Filter according to the 24 LS bits of the IPv6 address.
7:4	Reserved	Reserved.

Set IP Address Response

The NC must, in the absence of a checksum error or identifier mismatch, always accept the Set IP Address command and send a response using the format listed in Table 38-560.

Table 38-560.Set IP Address Response Packet Format

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..27	0x25	0x0	Reserved	
28..31	Checksum			

Get IP Address Command (Intel Command 0x25, Index = 0x1)

An MC uses the *Get* IP Address command to determine the IP address programmed in one of the IP address filters in a NC. The format of a Get IP Address command packet is listed in Table 38-561.

Table 38-561.Get IP Address Command Packet Format

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Manufacturer ID (Intel 0x157)			
20..23	0x25	0x1	Reserved	
24..27	Reserved		IP Address Number	IP Filter Pool
28..31	Checksum			

- IP address number. Defines the index of the IP address in the pool defined by the IP filter pool. The allowed values are listed in Table 38-558.
- IP filter pool:
  - 0x0: Mixed IP filters
  - 0x1: IPv4 filters
  - 0x2: IPv6 filters
  - 0x3 - 0xFF: Reserved

Get IP Address Response

The NC must, in the absence of a checksum error or identifier mismatch, always accept the *Get IP Address* command and send a response using the format listed in Table 38-562.

Table 38-562.Get IP Address Response Packet Format

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..27	0x25	0x1	IP Address Number	Get IP Flags
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
28..31	Management Controller IP Address			
32..35				
36..39				
40..43				
44..47	Checksum			

- MC IP Address — An IP address that is used by the MC. If the *IP Version* bit of the *Flags* field is 0b (IPv4), this is a 4-byte unicast IPv4 address in network byte order. In this case, the address occupies bytes 28-31 of the packet, and bytes 32-43 are ignored. If the *IP Version* bit of the *Flags* field is 1b (IPv6), this is a 16-byte unicast IPv6 address in network byte order. In this case, the address occupies the full field (bytes 28-43 of the packet).
- IP Address Number — Indicates which IP address filter is described in the response. Should be equal to the IP address number in the command.
- Table 38-563 lists the bits fields in the *Get IP Flags* field.

Table 38-563.Get IP Flag Field

Bit Position	Field Description	Value Description
0	Enable	0b = Filter is disabled. 1b = Filter is enabled.
1	IP Version	0b = IPv4. 1b = IPv6.
2	Mixed Index	0b = Index relates to the IPv4 or IPv6 only IP filter sets according to the IP version field. 1b = Index relates to the mixed IP filter set.

Table 38-563.Get IP Flag Field

Bit Position	Field Description	Value Description
3	MAC Based IP	This flag defines if the IPpv6 address is derived from a MAC address and thus only the 24 LSB should be used for the comparison. This flag is relevant only if the IP version = IPv6. 0b = Filter according to the full 128 bits of IPv6 address. 1b = Filter according to the 24 LS bits of the IPv6 address.
7:4	Reserved	Reserved.

Set Port Command (Intel Command 0x25, Index = 0x2)

An MC uses the Set Port command to communicate one of its TCP or UDP ports to a NC. The format of a Set Port command packet is listed in Table 38-564.

This command does not impact the forwarding results. It is used as a preliminary stage to the Set Binding command.

If the *Ignore Protocol* flag is cleared, the protocol should also match the *Protocol* field; otherwise, the *Protocol* field is ignored.

Table 38-564.Set Port Command Packet Format

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI header			
16..19	Manufacturer ID (Intel 0x157)			
20..23	0x25	0x2	Set Port Flags	Reserved
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
24..27	Port Index	Protocol	Port	
28..31	Checksum			

- Protocol — The value to match in the IPv4 header *Protocol* field or IPv6 header *Next Header* field. These values are defined by IANA. Allowed values are 0x6 (TCP) and 0x11 (UDP).
- Port — The value to match in the *Destination Port* or *Source Port* field of the TCP or UDP header. The legal port range for both TCP and UDP is 0-65,535. The compared field is defined by the port type flag.
- Port Index — Indicates which port filter is configured by the command. Allowed values are 1 to n, where n is the number of port filters supported by the network controller.

Table 38-565 lists the fields in the *Set Port Flags* field.

Table 38-565.Set Port Flags Field Descriptions

Bit Position	Field Description	Value Description
0	Enable	0b = Disable the filter. 1b = Enable the filter.
1	Ignore Protocol	0b = Filter by port and protocol. 1b = Filter by port only.
2	Port Type	0b = Compare destination port. 1b = Compare source port.
7:3	Reserved	Reserved.

Set Port Response

The NC must, in the absence of a checksum error or identifier mismatch, always accept the Set Port command and send a response using the format listed in Table 38-566.

Table 38-566.Set Port Response Packet Format

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
12..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..27	0x25	0x2	Reserved	
28..31	Checksum			

Get Port Command (Intel Command 0x25, Index = 0x3)

An MC uses the Get Port command to determine the TCP or UDP port programmed in one of the port filters in a NC. The format of a Get Port command packet is listed in Table 38-567.

Table 38-567.Get Port Command Packet Format

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
16..19	Manufacturer ID (Intel 0x157)			
20..23	0x25	0x3	Reserved	
24..27	Reserved		Port Index	Reserved
28..31	Checksum			

Table 38-568 lists the fields in the Get Port command.

Table 38-568.Get Port Command Field Descriptions

Field	Field Description	Value Description
Port Index	Indicates which port filter is requested by the command.	1 to *n*, where *n* is the number of port filters supported by the NC.

Get Port Response

The NC must, in the absence of a checksum error or identifier mismatch, always accept the Get Port command and send a response using the format listed in Table 38-569.

Table 38-569.Get Port Response Packet Format

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..27	0x25	0x3	Get Port Flags	Reserved
28..31	Port Index	Protocol	Port	
32..35	Checksum			

- Protocol — The value compared in the IPv4 header *Protocol* field or IPv6 header *Next Header* field. Possible values are 0x6 (TCP) and 0x11 (UDP).
- Port — The value compared in the *Destination Port or Source Port* field of the TCP or UDP header.
- Port Index — Indicates which port filter is reported by the response. Should match the Port Index in the command.

Table 38-570 lists the fields in the *Get Port Flags* field.

Table 38-570.Get Port Flags Field Descriptions

Bit Position	Field Description	Value Description
0	Enable	0b = Filter is disabled. 1b = Filter is enabled.
1	Ignore Protocol	0b = Filter by port and protocol. 1b = Filter by port only.
2	Port Type	0b = Compare destination port. 1b = Compare source port.
7:3	Reserved	Reserved.

Enable Unicast Infrastructure Filter Command (Intel Command 0x25, Index = 0x4)

A MC uses the Enable Unicast Infrastructure Filter command to configure a NC to forward copies of network infrastructure packets to it. Network infrastructure packets contain messages that are necessary for operating the network infrastructure layers (such as DHCP, ARP, and DNS messages). This is required when the MC shares an IP address with the host. In this case, both the host and the MC need to process the messages. As a result, the NC must forward the packets to both the MC and the host.

This command should be applied only after a MAC address is added using the Set MAC Address NC-SI command.

All the IP addresses added through the Set IP command before this command is given are considered as IP addresses of the MC for the purpose of this command.

If a Set IP command is received after this command was received, the list of IP address is not updated and this command should be given again.

The format of an Enable Unicast Infrastructure Filter command packet is listed in Table 38-571.

Table 38-571.Enable Unicast Infrastructure Filter Command

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..27	0x25	0x4	Reserved	
28..31	Unicast Infrastructure Filter Settings			
32..35	Checksum			
36..29	Padding			

Table 38-572 lists the sub fields of the *Unicast Infrastructure Filter Settings* field.

Table 38-572.Unicast Infrastructure Packet Filter Settings Field

Bit Position	Field Description	Value Description
0	ARP Response Packets Received From Wire	0x1 = Forward this packet type to both the host and the MC. 0x0 = Forward this packet type to the host only. For the purposes of this filter, an ARP response packet is defined to be any packet that meets all of the following requirements: The Ethernet *Destination Address* field contains the MAC address assigned to the MC. The Ethernet *Type* field contains 0x0806 (ARP). The ARP *Opcode* field is set to 0x0002 (response). The ARP *Target Protocol Address* field contains the IP address assigned to the MC.
1	ICMPv4 Request Packets Received From Wire	0x1 = Forward this packet type to both the host and the MC. 0x0 = Forward this packet type to the host only. For the purposes of this filter, an ICMP request packet is defined to be any packet that meets all of the following requirements: The Ethernet *Destination Address* field contains the MAC address assigned to the MC. The Ethernet *Type* field contains 0x0800 (IPv4). The IP *Destination Address* field contains the IPv4 address assigned to the MC. The IP *Protocol* field contains 1 (ICMP).
2	ICMPv6 Request Packets Received From Wire	0x1 = Forward this packet type to both the host and the MC. 0x0 = Forward this packet type to the host only. For the purposes of this filter, an ICMPv6 request packet is defined to be any packet that meets all of the following requirements: The Ethernet *Destination Address* field contains the MAC address assigned to the MC. The Ethernet *Type* field contains 0x86DD (IPv6). The IP *Destination Address* field contains the IPv6 address assigned to the MC. The IP *Next Header* field contains 58 (ICMPv6). This filter is not supported by the 10 GbE controller.

Table 38-572.Unicast Infrastructure Packet Filter Settings Field

Bit Position	Field Description	Value Description
3	DHCP Server Unicast Packets Received From Wire	0x1 = Forward this packet type to both the host and the MC. 0x0 = Forward this packet type to the host only. For the purposes of this filter, a DHCP server unicast packet is defined to be any packet that meets all of the following requirements: The Ethernet *Destination Address* field contains the MAC address assigned to the MC. The Ethernet *Type* field contains 0x0800 (IPv4). The IP *Destination Address* field contains either 255.255.255.255 (the local broadcast address) or the IPv4 address assigned to the MC. The IP *Protocol* field contains 17 (UDP). The UDP *Destination Port* field contains 68 (bootstrap protocol client).
4	DNS Server Packets Received From Wire	0x1 = Forward this packet type to both the host and the MC. 0x0 = Forward this packet type to the host only. For the purposes of this filter, a DNS server unicast packet is defined to be any packet that meets all of the following requirements: The Ethernet *Destination Address* field contains the MAC address assigned to the MC. The Ethernet *Type* field contains 0x0800 (IPv4). The IP *Destination Address* field contains the IPv4 address assigned to the MC. The IP *Protocol* field contains 17 (UDP). The UDP *Source Port* field contains 53 (domain name server).
5	DHCP Client Packets Transmitted By Host	0x1 = Forward this packet type to both the wire and the MC. 0x0 = Forward this packet type to the wire only. For the purposes of this filter, a DHCP client unicast packet is defined to be any packet that meets all of the following requirements: The Ethernet *Source Address* field contains the MAC address assigned to the MC. The Ethernet *Type* field contains 0x0800 (IPv4). The IP *Protocol* field contains 17 (UDP). The UDP *Destination Port* field contains 67 (bootstrap protocol server).
6	DHCPv6 Server Unicast Packets Received From Wire	0x1 = Forward this packet type to both the host and MC. 0x0 = Forward this packet type to the host only. For the purposes of this filter, a DHCPv6 server unicast packet is defined to be any packet that meets all of the following requirements: The Ethernet *Destination Address* field contains the MAC address assigned to the MC. The Ethernet *Type* field contains 0x86DD (IPv6). The IPv6 *Destination Address* field contains the IPv6 address assigned to the MC. The IP *Protocol* field contains 17 (UDP). The UDP *Destination Port* field contains 546 (DHCPv6 protocol client).
7	RMCP Primary Port - UDP	0x1 = Forward this packet type to the MC only. 0x0 = Forward this packet type to the host. For the purposes of this filter, a RMCP primary UDP packet is defined to be any packet that meets all of the following requirements: The Ethernet *Destination Address* field contains the MAC address assigned to the MC. The Ethernet *Type* field contains 0x86DD (IPv6) 0r 0x0800 (IPv4). The IP *Destination Address* field contains the one of the IP address assigned to the MC. The IP *Protocol* field contains 17 (UDP). The UDP *Destination Port* field contains 623 [aux bus shunt (primary RMCP port)].

Table 38-572.Unicast Infrastructure Packet Filter Settings Field

Bit Position	Field Description	Value Description
8	RMCP Primary Port - TCP	0x1 = Forward this packet type to the MC only. 0x0 = Forward this packet type to the host. For the purposes of this filter, a RMCP primary TCP packet is defined to be any packet that meets all of the following requirements: The Ethernet *Destination Address* field contains the MAC address assigned to the MC. The Ethernet *Type* field contains 0x86DD (IPv6) 0r 0x0800 (IPv4). The IP *Destination Address* field contains the one of the IP address assigned to the MC. The IP *Protocol* field contains 6 (TCP). The UDP *Destination Port* field contains 623 [aux bus shunt (primary RMCP port)].
9	RMCP Secondary Port - UDP	0x1 = Forward this packet type to the MC only. 0x0 = Forward this packet type to the host For the purposes of this filter, a RMCP secondary UDP packet is defined to be any packet that meets all of the following requirements: The Ethernet *Destination Address* field contains the MAC address assigned to the MC. The Ethernet *Type* field contains 0x86DD (IPv6) or 0x0800 (IPv4). The IP *Destination Address* field contains the one of the IP address assigned to the MC. The IP *Protocol* field contains 17 (UDP). The UDP *Destination Port* field contains 664 [secure aux bus (secondary RMCP port)].
10	RMCP Secondary Port - TCP	0x1 = Forward this packet type to the MC only. 0x0 = Forward this packet type to the host. For the purposes of this filter, a RMCP secondary TCP packet is defined to be any packet that meets all of the following requirements: The Ethernet *Destination Address* field contains the MAC address assigned to the MC. The Ethernet *Type* field contains 0x86DD (IPv6) 0r 0x0800 (IPv4). The IP *Destination Address* field contains the one of the IP address assigned to the MC. The IP *Protocol* field contains 6 (TCP). The TCP *Destination Port* field contains 664 [secure aux bus (secondary RMCP port)].
31:11	Reserved	None.

Enable Unicast Infrastructure Filter Response

The NC, in the absence of a checksum error or identifier mismatch, always accept the Enable Unicast Infrastructure Filter command and send a response using the format listed in Table 38-573. Currently no command-specific reason codes are identified for this response.

Table 38-573.Enable Unicast Infrastructure Filter Response Packet Format

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..27	0x25	0x4	Reserved	
28..29	Checksum			

38.37.6.4.15 Get Shared IP Capabilities Command (Intel Command 0x25, Index = 0x5)

An MC uses the Get Shared IP Capabilities command to determine the level of support of shared IP of the device. The format of a Get Shared IP Capabilities command packet is listed in Table 38-574.

Table 38-574. Get Shared IP Capabilities Command Packet Format

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Manufacturer ID (Intel 0x157)			
20..23	0x25	0x5	Reserved	
24..27	Checksum			

Get Shared IP Capabilities Response

The NC must, in the absence of a checksum error or identifier mismatch, always accept the Get Shared IP Capabilities command and send a response, using the format listed in Table 38-575. Currently no command-specific reason codes are identified for this response.

Table 38-575. Get Shared IP Capabilities Response Packet Format

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..27	0x25	0x5	Number of Mixed IP Address	Number of IPv4 Only Addresses
28..31	Number of IPv6 only addresses	Number of Ports	Number Of Bindings	Filtering Capabilities
32..35	Unicast Infrastructure Filter Capabilities			
36..30	Checksum			

- Number of mixed IP addresses — The number of supported IP filters that can be used for IPv4 or IPv6. The 10 GbE controller does not support mixed IP address filters.
- Number of IPv4 only addresses — The number of supported IP filters that can be used for IPv4 only. The 10 GbE controller supports three IPv4 address filters.
- Number of IPv6 only addresses — The number of supported IP filters that can be used for IPv6 only. The 10 GbE controller supports four IPv6 address filters.
- Number of ports — The number of supported port filters.
- Number of bindings — Defines the number of IP addresses that can be bound with different ports.
- Unicast infrastructure filter capabilities — Defines the optional unicast infrastructure filter capabilities that the channel supports. The bit definitions for this field correspond directly with the bit definitions for the *Unicast Infrastructure Filter Settings* field defined for the Unicast Infrastructure Filter command listed in Table 38-572. A bit set to 1b indicates that the channel supports the filter associated with that bit position; otherwise, the channel does not support that

filter. The 10 GbE controller supports all filters but ICMPv6 filtering, so the returned value is 0x7FB.

- Table 38-576 lists the bits fields in the *Filtering Capabilities* field.

Table 38-576.Filtering Capabilities Field

Bit Position	Field Description	Value Description
0	IPv4 support	0b = IPv4 filtering is not supported. 1b = IPv4 filtering is supported.
1	IPv6 support	0b = IPv6 filtering is not supported. 1b = IPv6 filtering is supported.
2	Protocol filtering support	0b = Filtering by protocol is not supported. 1b = Filtering by protocol is supported.
3	Source port filtering support	0b = Port filtering is supported only for destination port. 1b = Port filtering is supported for destination port or source port.
7:4	Reserved	Reserved.

Shared IP Enable Broadcast Filtering Command (Intel Command 0x25, Index = 0x6)

A new *shared IP enable broadcast filtering* is defined to enable the MC to limit the flow of ARP requests to those that contain a target IP address value that matches the MC IP address.

This command should be used instead of the regular NC-SI Enable Broadcast Filtering command.

Note: Receiving a standard NC-SI Enable Broadcast Filtering command enables the matching bits in this command. Receiving a standard NC-SI Disable Broadcast Filter Command clears the settings in this command.

The format of an Shared IP Enable Broadcast Filtering command packet is listed in Table 38-571.

Table 38-577.Shared IP Enable Broadcast Filtering Command

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..27	0x25	0x6	Reserved	
28..31	Shared IP Broadcast Packet Filter Settings			
32..35	Checksum			
36..29	Padding			

The content of the *Shared IP Broadcast Packet Filter Settings* field is listed in Table 38-578. Bit 4 has been added to the standard enable broadcast filtering command limit ARP broadcast packets to the MC IP add*ress*.

Table 38-578.Shared IP Broadcast Packet Filter Settings Field

Bit Position	Field Description	Value Description
3:0	As defined in DSP0222	As defined in DSP0222 in 8.4.33 Enable Broadcast Filter command (0x10) - table 68.
4	Limit ARP Broadcast Packets to Management Controller IP Address	When bit 0 is set, it limits the flow of ARP packets to the MC as follows: 0x1 = Forward only ARP broadcast packets that are targeted at IP addresses bound to the MC. 0x0 = Forward all ARP broadcast packets to the MC. All the IPs set by the *Set IP* command before this command is given will be included in forwarding. This field is optional. If unsupported, the behavior for ARP packets is set according to bit 1 in this structure. The value must be set to 0b if unsupported.
31:5	Reserved	None.

Shared IP Enable Broadcast Filtering Response

The NC must, in the absence of a checksum error or identifier mismatch, always accept the Shared IP Enable Broadcast Filtering command and send a response using the format listed in Table 38-573. Currently no command-specific reason codes are identified for this response.

Table 38-579.Shared IP Enable Broadcast Filtering Packet Format

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..27	0x25	0x6	Reserved	
28..29	Checksum			

Shared IP Enable Global Multicast Filtering Command (Intel Command 0x25, Index = 0x7)

A new shared IP enable global multicast filtering is defined to enable the MC to enable the forwarding of IEEE 802.1X Extensible Authentication Protocol over LAN (EAPOL) frames to the MC IP address. IEEE 802.1X defines methods for port-based network access control.

This command should be used instead of the regular NC-SI Enable Global Multicast Filtering command.

Note: Receiving a standard NC-SI Enable Global Multicast Filtering command enables the matching bits in this command. Receiving a standard NC-SI Disable Global Multicast Filter command clears the settings in this command.

The format of an Shared IP Enable Global Multicast Filtering command packet is listed in Table 38-580.

Table 38-580.Shared IP Enable Global Multicast Filtering Command

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
20..23	Manufacturer ID (Intel 0x157)			
24..27	0x25	0x7	Reserved	
28..31	Shared IP Multicast Packet Filter Settings			
32..35	Checksum			
36..29	Padding			

The content of the *Shared IP Multicast Packet Filter Settings* field is listed in Table 38-581. Bit 4 has been added to the standard enable broadcast filtering command limit ARP broadcast packets to MC IP address.

Table 38-581.Shared IP Multicast Packet Filter Settings Field

Bit Position	Field Description	Value Description
2:0	As defined in DSP0222	As defined in DSP0222 in 8.4.37 Enable Global Multicast Filter Command (0x12) - table 74.
3	IEEE 802.1X EAPOL	0x1 = Forward this packet type to the MC. 0x0 = Filter out this packet type. For the purposes of this filter, a IEEE 802.1X multicast packet is defined to be any packet that meets all of the following requirements: The destination MAC address field is set to the layer 2 multicast address 01:80:c2:00:00:03. The EtherType field is set to 0x888E (802.1X PAE). This field is optional. If unsupported, multicast 802.1X packets are blocked when multicast filtering is enabled, unless they are matched by an address filter configured using the Set MAC Address command. The value must be set to 0b if unsupported.
31:4	Reserved	None.

Shared IP Enable Global Multicast Filtering Response

The NC must, in the absence of a checksum error or identifier mismatch, always accept the Shared IP Enable Global Multicast Filtering command and send a response using the format listed in Table 38-582. Currently no command-specific reason codes are identified for this response.

Table 38-582.Shared IP Enable Global Multicast Filtering Packet Format

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..27	0x25	0x7	Reserved	
28..29	Checksum			

Get Shared IP Parameters Command (Intel Command 0x25, Index = 0x8).

The Get Shared IP parameters command can be used by the MC to request that the channel send the MC a copy of part of the currently stored parameter settings that have been put into effect by the MC related to shared IP filtering. The format of a Get Shared IP Capabilities command packet is listed in Table 38-583.

Table 38-583.Get Shared IP Parameters Command Packet Format

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Manufacturer ID (Intel 0x157)			
20..23	0x25	0x8	Reserved	
24..27	Checksum			

Get Shared IP Parameters Response

The NC must, in the absence of a checksum error or identifier mismatch, always accept the Get Shared IP parameters command and send a response using the format listed in Table 38-584. Currently no command-specific reason codes are identified for this response.

Table 38-584.Get Shared IP Parameters Response Packet Format

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..27	0x25	0x5	Reserved	
28..31	Number of IP Addresses	IP addresses Flags		
32..35	Number of ports	Ports Flags		
36..39	Unicast Infrastructure Filter Settings			
40..43	Broadcast Filtering Settings			
44..47	Multicast Filtering Settings			
48..51	Checksum			

- Number of IP addresses — The number of supported IP filters including all the types of IP addresses (IPv4 only, IPv6 only and mixed).
- IP address flags — The enable/disable state for each supported IP address. See Table 38-585.

Table 38-585.IP Address Flags Field

Bit Position	Field Description	Value Description
0	IP Address 1 Status	0b = Default or unsupported or disabled. 1b = Enabled.
1	IP Address 2 Status or Reserved	0b = Default or unsupported or disabled. 1b = Enabled.
2	IP Address 3 Status or Reserved	0b = Default or unsupported or disabled. 1b = Enabled.

Table 38-585.IP Address Flags Field

Bit Position	Field Description	Value Description
...		
23	IP Address 24 Status or Reserved	0b = Default or unsupported or disabled. 1b = Enabled.

Note: IP address flags are organized in the following order: IPv4 addresses first, followed by IPv6 addresses, followed by mixed addresses, with the number of each corresponding to those reported through the Get Shared IP Capabilities command.

For example, if the interface reports four IPv4 filters, two IPv6 filters, and two mixed filters, then IP addresses 1 through 4 are those currently configured through the interface's IPv4 filters, IP addresses 5 and 6 are those configured through the IPv6 filters, and 7 and 8 are those configured through the mixed filters.

The actual settings of each enabled IP address can be found using the Get IP address command.

- Number of ports — The number of supported port filters.
- Port flags — The enable/disable state for each supported ports. See Table 38-585.

Table 38-586.Port Flags Field

Bit Position	Field Description	Value Description
0	Port 1 Status	0b = Default or unsupported or disabled. 1b = Enabled.
1	Port 2 Status or Reserved	0b = Default or unsupported or disabled. 1b = Enabled.
2	Port 3 Status or Reserved	0b = Default or unsupported or disabled. 1b = Enabled.
...		
23	Port 24 Status or Reserved	0b = Default or unsupported or disabled. 1b = Enabled.

Note: The actual settings of each enabled port can be found using the Get Port command

- Unicast Infrastructure Filter Settings — Defines the optional unicast infrastructure filter capabilities settings. The bit definitions for this field correspond directly with the bit definitions for the *Unicast Infrastructure Filter Settings* field defined for the Unicast Infrastructure Filter command in Table 38-572. A bit set to 1b indicates that the filter associated with that bit position is enabled; otherwise, the filter is not enabled.
- Broadcast Filter Settings — Defines the optional broadcast filter settings. The bit definitions for this field correspond directly with the bit definitions for the *Broadcast Filter Settings* field defined for the Shared IP Broadcast Filtering command in Table 38-578. A bit set to 1b indicates that the filter associated with that bit position is enabled; otherwise, the filter is not enabled.
- Global Multicast Filter Settings — Defines the optional multicast filter capabilities settings. The bit definitions for this field correspond directly with the bit definitions for the *Multicast Filter Settings* field defined for the Shared IP Global Multicast Filtering command in Table 38-578. A bit set to 1b indicates that the filter associated with that bit position is enabled; otherwise, the filter is not enabled.

Set Binding Command (Intel Command 0x25, Index = 0x9)

The Set Binding command is used by the MC to define which combination of MAC addresses, VLAN tags, IP addresses and TCP/UDP ports should be forwarded to the MC. The format of a Set Binding command packet is listed in Table 38-587.

Once a Set Binding command is activated, all the previous forwarding rules based on the Set MAC Address or Set VLAN filter commands are disabled and should be re-enabled using the Set Binding command. Subsequent Set MAC Address or Set VLAN filter commands are used to enable MAC or VLAN addresses for the Set Binding command but does not impact the forwarding rules.

Table 38-587.Set Binding Command Packet Format

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
16...19	Manufacturer ID (Intel 0x157)			
20..23	0x25	0x9	Binding Index	Set Binding Flags
24..27	Enabled MAC Addresses			
28..31	Enabled VLAN			
32..35	Enabled IP addresses			
36..39	Enabled Ports (MSB)			
40..43	Enabled Ports (LSB)			
44..47	Checksum			

Table 38-588 lists the fields in the *Set Binding Flags* field.

Table 38-588.Set Binding Flags Field Descriptions

Bit Position	Field Description	Value Description
0	Enable	0b = Disable the binding. 1b = Enable the binding.
1	Exclusive to MC	0b = Traffic matching this filter is sent to the MC and to the host. 1b= Traffic matching this filter is sent to the MC only.
2	Apply to Network[1]	0b = Do not compare traffic received from the network when checking this binding. 1b = Compare traffic received from the network when checking this binding.
3	Apply to Host	0b = Do not compare traffic received from the host when checking this binding. 1b = Compare traffic received from the host when checking this binding.
7:4	Reserved	Reserved.

Notes:

1. At least one of the apply to network/host flags should be set for enabled bindings. Clearing both of them is equivalent to disabling the filter.

- Binding Index — Indicates which binding is configured by the command.The value should be smaller than the number of supported bindings as reported in the get shared IP capabilities response in the *Number of Bindings* field.
- Enabled MAC Addresses — The MAC addresses participating in this binding. The numbering of the MAC addresses is similar to the one used in the MAC address flags in the get parameters response. Namely, MAC addresses are returned in the following order: unicast filtered addresses first, followed by multicast filtered

addresses, followed by mixed filtered addresses, with the number of each corresponding to those reported through the Get Capabilities command. A MAC address can be added to a binding only if previously enabled through a Set MAC Address NC-SI command.

- Enabled VLAN — The VLAN IDs participating in this binding. The numbering of the VLAN IDs. A VLAN tag can be added to a binding only if previously enabled through a Set VLAN Filter NC-SI command.
- Enabled IP Addresses — The IP addresses participating in this binding. The numbering of the IP addresses is similar to the one used in Section "Get Shared IP Parameters Command (Intel Command 0x25, Index = 0x8)." An IP address can be added to a binding only if previously enabled through a Set IP Address Intel OEM command.
- Enabled Ports — The ports participating in this binding. A port can be added to a binding only if previously enabled through a Set Port Intel OEM command.

Set Binding Address Response

The NC must, in the absence of a checksum error or identifier mismatch, always accept the Set Binding command and send a response using the format listed in Table 38-589.

Table 38-589.Set binding response packet format

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..27	0x25	0x9	Reserved	
28..31	Checksum			

Get Binding Command (Intel Command 0x25, Index = 0xA)

A MC uses the Get Binding command to determine the current programming of one of the bindings in a NC. The format of a Get Binding command packet is listed in Table 38-590.

Table 38-590.Get Binding Command Packet Format

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Manufacturer ID (Intel 0x157)			
20..23	0x25	0xA	Binding Number	Reserved
24..27	Checksum			

- Binding Index — Indicates which binding is requested by the command. The value should be smaller than the number of supported bindings as reported in the Get Shared IP Capabilities Response in the *Number of Bindings* field.

Get Binding Response

The NC must, in the absence of a checksum error or identifier mismatch, always accept the Get Binding command and send a response using the format listed in Table 38-591.

Table 38-591.Get Binding Response Packet Format

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..27	0x25	0xA	Binding Number	Get Binding Flags
28..31	Enabled MAC Addresses			
32..35	Enabled VLAN			
36..39	Enabled IP Addresses			
40..43	Enabled Ports (MSB)			
44..47	Enabled Ports (LSB)			
48..51	Checksum			

The fields in the Get Binding response are equivalent to their counterparts in the Set Binding command.

Set Shared Mode Command (Intel Command 0x25, Index = 0xB)

An MC uses the Set Shared Mode command to indicate to the NIC it intends to operate in shared MAC/IP mode or in dedicated MAC mode.

If used, this command should be sent before any of the regular or OEM NC-SI commands used to set forwarding filters. When this command is received, all the filters are cleared.

This command is only needed when the Intel OEM commands with command ID 0x25 are used to configure the shared behavior. If other commands are used, users should take care of the right configuration of the filters.

When shared mode is activated, the Set MAC and Set VLAN NC-SI commands do not impact the receive filtering until a Set Binding or Enable Unicast Infrastructure Filter command is received.

Any other command from this section (38.37.6.4.14) received before shared mode is set fails with a Not is Shared Mode (0x5092) reason.

The format of a Set Shared Mode command packet is listed in Table 38-592.

Table 38-592.Set Shared Mode Command Packet Format

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Manufacturer ID (Intel 0x157)			
20..23	0x25	0xB	Shared Mode	Reserved
24..27	Checksum			

- Shared mode:
  — 0x0: Dedicated MAC mode.
  — 0x1: Shared MAC/IP mode.

Set Shared Mode Response

The NC must, in the absence of a checksum error or identifier mismatch, always accept the Set Shared Mode command and send a response using the format listed in Table 38-593.

Table 38-593.Set Shared Mode Response Packet Format

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..27	0x25	0xB	Shared Mode	Reserved
48..51	Checksum			

38.37.6.4.16 OS2BMC Configuration

These commands control enabling of the OS2BMC flow.

EnableOS2BMC Flow Command (Intel Command 0x40, Index 0x1)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...21	0x40	0x01		

EnableOS2BMC Flow Response (Intel Command 0x40, Index 0x1)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...25	0x40	0x01		

Enable Network-to-BMC Flow Command (Intel Command 0x40, Index 0x2)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...21	0x40	0x02		

Enable Network-to-BMC Flow Response (Intel Command 0x40, Index 0x2)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...25	0x40	0x02		

Enable Both Host and Network-to-BMC Flows Command (Intel Command 0x40, Index 0x3)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...21	0x40	0x03		

Enable Both Host and Network-to-BMC Flows Response (Intel Command 0x40, Index 0x3)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...25	0x40	0x03		

Set MC IP Address Command (Intel Command 0x40, Index 0x4)

This command is used to expose the MC IP address to the host. This command is supported by the 10 GbE controller, but no action is taken upon reception. The IP address is not stored and when a Get OS2BMC Parameters command is received, the IP valid flag is cleared.

The IP type entry indicate whether the IP address is an IPv4 or an IPv6 address:

0 = IPv4.

1 = IPv6.

2 = No IP address, then the command should not include an IP address.

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Manufacturer ID (Intel 0x157)			
20..23	0x40	0x04	IP type	IPv6 Address (MSB, byte 15)/IPv4 Address (MSB, byte 3)
24..27	IPv6 Address (byte 14)/ IPv4 Address (byte 2)	IPv6 Address (byte 13)/ IPv4 Address (byte 1)	IPv6 Address (byte 12)/ IPv4 Address (LSB, byte 0)	IPv6 Address (byte 11)/ Reserved
28..31	..	..	..	..
32..35	..	..	..	..
36..38	..	..	IPv6 Address (LSB, byte 0)/Reserved	

Set BMC IP Address Response (Intel Command 0x40, Index 0x4)

	Bits			
Bytes	**31...24**	**23...16**	**15...08**	**07...00**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...25	0x40	0x04		

Get OS2BMC Parameters Command (Intel Command 0x41)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20	0x41			

Get OS2BMC Parameters Response (Intel Command 0x41)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...27	0x41	Status	IPv6 Address (MSB, byte 15)/IPv4 Address (MSB, byte 3)	IPv6 Address (byte 14)/ IPv4 Address (byte 2)
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
28..31	IPv6 Address (byte 13)/ IPv4 Address (byte 1)	IPv6 Address (byte 12)/ IPv4 Address (LSB, byte 0)	IPv6 Address (byte 11)/ Reserved	..
32..35	..	..	..	..
36..39	..	..	..	..
39..40	..	IPv6 Address (LSB, byte 0)/Reserved		

Where the status byte partition is as follows:

Table 38-594.Status Byte Description

Bits	Content
0	0b = IPv4. 1b = IPv6. Relevant only if the IP address valid bit is set.
1	IP address valid. Never valid for the 10 GbE controller.
1:0	Reserved.
2	Network to BMC Status. 0b = Network-to-BMC flow is disabled. 1b = Network-to-BMC flow is enabled.
3	OS2BMC Status. 0b = OS2BMC flow is disabled. 1b = OS2BMC flow is enabled.
7:4	Reserved.

38.37.6.4.17 Get Controller Information Command (Intel Command 0x48, Index 0x1)

This command gathers the controller identification information and return it back to the MC.

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Manufacturer ID (Intel 0x157)			
20...23	0x48	0x1		

Get Controller Information Response (Intel Command 0x48, Index 0x1)

	Bits			
Bytes	**[31:24]**	**[23:16]**	**[15:08]**	**[07:00]**
00...15	NC-SI Header			
16...19	Response Code		Reason Code	
20...23	Manufacturer ID (Intel 0x157)			
24...27	0x48	0x01	Reserved	Number of Inventory Entries
28...31	Controller Info Item 1 ID	Controller Info Item 1 Length	Controller Info Item 1 Data	
...	....			
...	Controller Info Item 2 ID	Controller Info Item 2 Length	Controller Info Item 2 Data	
...	....			
...	Controller Info Item n ID	Controller Info Item n Length	Controller Info Item n Data	
...	....			

The possible inventory items are described as follows. Note that not all the inventory items would be present in all the implementations of this command.

Table 38-595.Controller Information Items

ID	Length (In Bytes)	Data	Notes
0x00	3	Device ID (2 bytes) + RevID	This is the hardware default value (no value programmed via the NVM).
0x0B	2	NVM Image Version	
0x0C	4	EMP ROM Internal Version	
0x0D	4	EMP Flash Internal Version	Same version as in Get Version Admin command.
0x0E	2	PXE Firmware Version	MajorVersion.MinorVersion.Build.
0x0F	2	iSCSI Firmware Version	
0x10	2	uEFI Firmware Version	

38.37.6.4.18 NVM Error AEN (Intel AEN 0x82)

The following is the AEN that might be sent by the NC following a detection of a wrong CRC/checksum on a firmware related section in the NVM. NC is required to store this AEN internally until a connection to the MC is established so the report could be issued in all cases.

This AEN must be enabled using the NC-SI AEN Enable command, using bit 18 (0x40000) of the AEN enable mask.

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00...15	NC-SI AEN Header			
20...23	Reserved			0x82
24...27	Index of module on which the error was found. The encoding is as defined in *GL_MNG_FWSM.EXT_ERR_IND* field.			

38.37.6.5 Basic NC-SI Workflows

38.37.6.5.1 Package States

A NC package can be in one of the following two states:

1. Selected — The package is allowed to use the NC-SI lines, meaning the NC package might send data to the MC.
2. De-selected — The package is not allowed to use the NC-SI lines, meaning, the NC package cannot send data to the MC.

The MC must select no more than one NC package at any given time. Package selection can be accomplished in one of two methods:

1. Select Package command — This command explicitly selects the NC package.
2. Any other command targeted to a channel in the package also implicitly selects that NC package.

Package de-select can be accomplished only by issuing the De-Select Package command. The MC should always issue the Select Package command as the first command to the package before issuing channel-specific commands. For further details on package selection, refer to the NC-SI specification.

38.37.6.5.2 Channel states

A NC channel can be in one of the following states:

1. Initial State — The channel only accepts the Clear Initial State command (the package also accepts the Select Package and De-Select Package commands).
2. Active State — This is the normal operational mode. All commands are accepted.

For normal operation mode, the MC should always send the Clear Initial State command as the first command to the channel.

38.37.6.5.3 Discovery

After interface power-up, the MC should perform a discovery process to discover the NCs that are connected to it. This process should include an algorithm similar to the following:

1. For package_id=0x0 to MAX_PACKAGE_ID:
   a. Issue a Select Package command to package ID package_id.
   b. If a response was received:
   c. For internal_channel_id = 0x0 to MAX_INTERNAL_CHANNEL_ID.
   d. Issue a Clear Initial State command for package_id | internal_channel_id (the combination of package_id and internal_channel_id to create the channel ID).
   e. If a response was received:
   f. Consider internal_channel_id as a valid channel for the package_id package.
   g. The MC can now optionally discover channel capabilities and version ID for the channel.
   h. Else, if a response was not received, then issue a Clear Initial State command three times.
   i. Issue a De-Select Package command to the package (and continue to the next package).
   j. Else, if a response was not received, issue a Select Packet command three times.

38.37.6.5.4 Configurations

This section details different configurations that should be performed by the MC.

It is good practice that the MC not consider any configuration valid unless the MC has explicitly configured it after every reset (entry into the initial state). As a result, it is recommended that the MC re-configure everything at power-up and channel/package resets.

NC Capabilities Advertisement

NC-SI defines the Get Capabilities command. It is recommended that the MC use this command and verify that the capabilities match its requirements before performing any configurations. For example, the MC should verify that the NC supports a specific AEN before enabling it.

Receive Filtering

In order to receive traffic, the BMC must configure the NC with receive filtering rules. These rules are checked on every packet received on the LAN interface (such as from the network). Only if the rules matched, will the packet be forwarded to the BMC.

MAC Address Filtering

NC-SI defines three types of MAC address filters: unicast, multicast and broadcast. To be received (not dropped) a packet must match at least one of these filters. The MC should set one MAC address using the Set MAC Address command and enable broadcast and global multicast filtering.

Unicast/Exact Match (Set MAC Address command)

This filter filters on specific 48-bit MAC addresses. The MC must configure this filter with a dedicated MAC address.

The NC might expose three types of unicast/exact match filters (such as MAC filters that match on the entire 48 bits of the MAC address): unicast, multicast and mixed. The 10 GbE controller exposes two mixed filters, which might be used both for unicast and multicast filtering. The MC should use one mixed filter for its MAC address.

Refer to NC-SI specification — Set MAC Address for further details.

Broadcast (Enable/Disable Broadcast Filter command)

NC-SI defines a broadcast filtering mechanism that has the following states:

1. Enabled — All broadcast traffic is blocked (not forwarded) to the BMC, except for specific filters (such as ARP request, DHCP, and NetBIOS).
2. Disabled — All broadcast traffic is forwarded to the BMC, with no exceptions.

Refer to NC-SI specification Enable/Disable Broadcast Filter command.

Global Multicast (Enable/Disable Global Multicast Filter)

NC-SI defines a multicast filtering mechanism which has the following states:

1. Enabled — All multicast traffic is blocked (not forwarded) to the BMC.
2. Disabled — All multicast traffic is forwarded to the BMC, with no exceptions.

The recommended operational mode is Enabled, with specific filters set. Not all multicast filtering modes are necessarily supported. Refer to NC-SI specification Enable/Disable Global Multicast Filter command for further details.

VLAN

NC-SI defines the following VLAN work modes:

Mode	Command and Name	Descriptions
Disabled	Disable VLAN command.	In this mode, no VLAN frames are received.
Enabled #1	Enable VLAN command with VLAN only.	In this mode, only packets that matched a VLAN filter are forwarded to the MC.
Enabled #2	Enable VLAN command with VLAN only + non-VLAN.	In this mode, packets from mode 1 + non-VLAN packets are forwarded.
Enabled #3	Enable VLAN command with Any-VLAN + non-VLAN.	In this mode, packets are forwarded regardless of their VLAN state.

Refer to NC-SI specification — Enable VLAN command for further details.

The 10 GbE controller only supports modes #1 and #3. Recommendation:

1. Modes:
   a. If VLAN is not required — Use the disabled mode.
   b. If VLAN is required — Use the enabled #1 mode.
2. If enabling VLAN, The MC should also set the active VLAN ID filters using the NC-SI Set VLAN Filter command prior to setting the VLAN mode.

38.37.6.5.5 PT Traffic States

The MC has independent, separate controls for enablement states of the receive (from LAN) and of the transmit (to LAN) PT paths.

38.37.6.5.6 Channel Enable

This mode controls the state of the receive path:

1. Disabled — The channel does not pass any traffic from the network to the MC.
2. Enabled — The channel passes any traffic from the network (that matched the configured filters) to the MC.

This state also affects AENs: AENs is only sent in the enabled state.

The default state is disabled.

It is recommended that the MC complete all filtering configuration before enabling the channel.

38.37.6.5.7 Network Transmit Enable

This mode controls the state of the transmit path:

1. Disabled — the channel does not pass any traffic from the MC to the network.
2. Enabled — the channel passes any traffic from the MC (that matched the source MAC address filters) to the network.

The default state is disabled.

The NC filters PT packets according to their source MAC address. The NC tries to match that source MAC address to one of the MAC addresses configured by the Set MAC Address command. As a result, the MC should enable network transmit only after configuring the MAC address.

It is recommended that the MC complete all filtering configuration (especially MAC addresses) before enabling the network transmit.

This feature can be used for fail-over scenarios. See Section 38.37.6.9.3.

38.37.6.6 Asynchronous Event Notifications (AENs)

AENs are unsolicited messages sent from the NC to the MC to report status changes (such as link change, operating system state change, etc.).

Recommendations:

- The MC firmware designer should use AENs. To do so, the designer must take into account the possibility that a NC-SI response frame (such as a frame with the NC-SI EtherType), arrives out-of-context (not immediately after a command, but rather after an out-of-context AEN).
- To enable AENs, the MC should first query which AENs are supported, using the Get Capabilities command, then enable desired AEN(s) using the Enable AEN command, and only then enable the channel using the Enable Channel command.

38.37.6.7 Querying Active Parameters

The MC can use the Get Parameters command to query the current status of the operational parameters.

38.37.6.8 Resets

In NC-SI there are two types of resets defined:

1. Synchronous entry into the initial state.
2. Asynchronous entry into the initial state.

Recommendations:

- It is very important that the MC firmware designer keep in mind that following any type of reset, all configurations are considered as lost and thus the MC must re-configure both the synchronous and asynchronous entries.
- As an asynchronous entry into the initial state might not be reported and/or explicitly noticed, the MC should periodically poll the NC with NC-SI commands (such as Get Version ID, Get Parameters, etc.) to verify that the channel is not in the initial state. Should the NC channel respond to the command with a Clear Initial State Command Expected reason code, the MC should consider the channel (and most probably the entire NC package) as if it underwent a (possibly unexpected) reset event. Thus, the MC should re-configure the NC. See the NC-SI specification section on Detecting Pass-through Traffic Interruption.
- The Intel recommended polling interval is 2-3 seconds.

For exact details on the resets, refer to NC-SI specification.

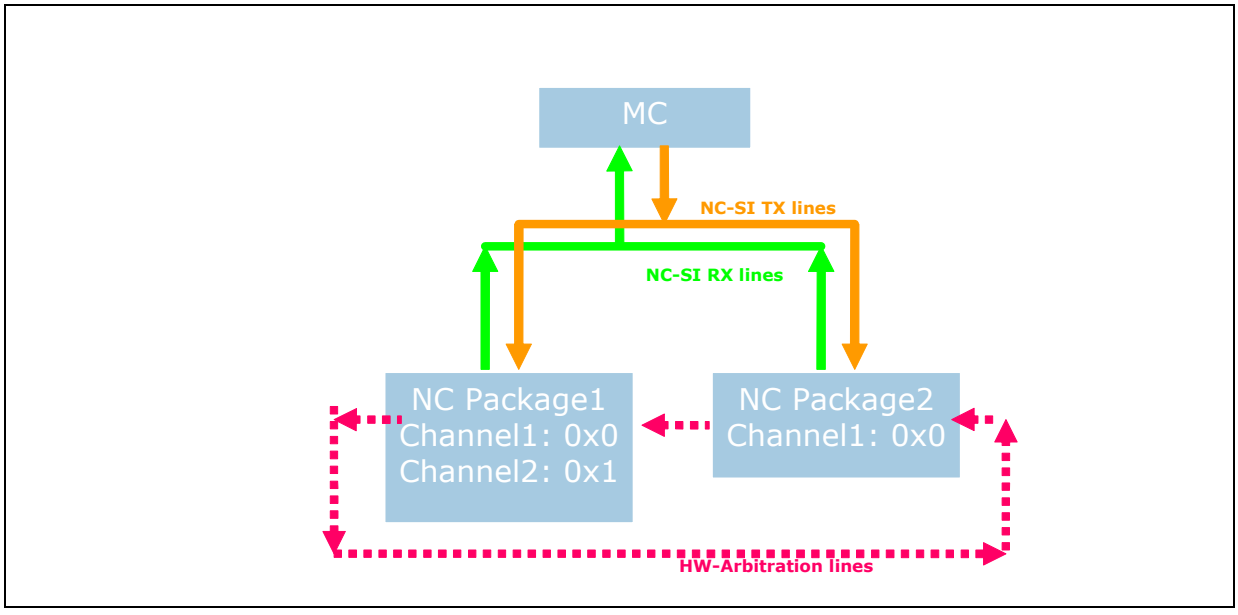
38.37.6.9 Advanced Workflows

38.37.6.9.1 Multi-NC Arbitration

As described in Section 38.37.6.1.2, in a multi-NC environment, there is a need to arbitrate the NC-SI lines.

Figure 38-136 shows the system topology of such an environment.

Figure 38-136.Multi-NC Environment



See Figure 38-136. The NC-SI Rx lines are shared between the NCs. To enable sharing of the NC-SI Rx lines, NC-SI has defined an arbitration scheme.

The arbitration scheme mandates that only one NC package can use the NC-SI Rx lines at any given time. The NC package that is allowed to use these lines is defined as selected. All the other NC packages are de-selected.

NC-SI has defined two mechanisms for the arbitration scheme:

1. Package selection by the MC. In this mechanism, the MC is responsible for arbitrating between the packages by issuing NC-SI commands (Select/De-Select Package). The MC is responsible for having only one package selected at any given time.
2. Hardware arbitration. In this mechanism, two additional pins on each NC package are used to synchronize the NC package. Each NC package has an ARB_IN and ARB_OUT line and these lines are used to transfer tokens. A NC package that has a token is considered selected.

Note: Hardware arbitration is enabled by the NC-SI *HW Arbitration Enable* configuration bit in the NC-SI Configuration 1 NVM word.

For details, refer to the NC-SI specification.

38.37.6.9.2 Package Selection Sequence Example

Following is an example work flow for a MC and occurs after the discovery, initialization, and configuration.

Assuming the MC needs to share the NC-SI bus between packages, the MC should:

1. Define a time-slot for each device.
2. Discover, initialize, and configure all the NC packages and channels.
3. Issue a De-Select Package command to all the channels.
4. Set active_package to 0x0 (or the lowest existing package ID).
5. At the beginning of each time slot the MC should:
   a. Issue a De-Select Package command to the active_package. The MC must then wait for a response and then an additional timeout for the package to become de-selected (200 us). See the NC-SI specification table 10 — parameter NC Deselect to Hi-Z Interval.
   b. Find the next available package (typically active_package = active_package + 1).
   c. Issue a Select Package command to active_package.

38.37.6.9.3 Multiple Channels (Fail-Over)

In order to support a fail-over scenario, it is required from the MC to operate two or more channels. These channels might or might not be in the same package.

The key element of a fault-tolerance fail-over scenario is having two (or more) channels identifying to the switch with the same MAC address, but only one of them being active at any given time (such as switching the MAC address between channels). To accomplish this, NC-SI provides the following commands:

1. Enable Network Tx command — This command enables shutting off the network transmit path of a specific channel. This enables the MC to configure all the participating channels with the same MAC address but only enable one of them.
2. Link Status Change AEN or Get Link Status command.

38.37.6.9.4 Fail-Over Algorithm Example

The following is a sample work flow for a fail-over scenario for the 10 GbE controller (one package and four channels):

1. The MC initializes and configures all channels after power-up. However, the MC uses the same MAC address for all of the channels.

2. The MC queries the link status of all the participating channels. The MC should continuously monitor the link status of these channels. This can be accomplished by listening to AENs (if used) and/or periodically polling using the Get Link Status command.
3. The MC then only enables channel 0 for network transmission.
4. The MC then issues a gratuitous ARP (or any other packet with its source MAC address) to the network. This packet informs the switch that this specific MAC address is registered to channel 0's specific LAN port.
5. The MC begins normal work flow.
6. Should the MC receive an indication (AEN or polling) that the link status for the active channel (channel 0) has changed, the MC should:
   a. Disable channel 0 for network transmission.
   b. Check if a different channel is available (link is up).
   c. If found:
      - Enable network Tx for that specific channel.
      - Issue a gratuitous ARP (or any other packet with its source MAC address) to the network. This packet informs the switch that this specific MAC address is registered to channel 0's specific LAN port.
      - Resume normal work flow.
      - If not found, report the error and continue polling until a valid channel is found.

The previous algorithm can be generalized such that the start-up and normal work flow are the same. In addition, the MC might need to use a specific channel (such as channel 0). In this case, the MC should switch the network transmit to that specific channel as soon as that channel becomes valid (link is up).

Recommendations:

- Wait for a link-down-tolerance timeout before a channel is considered invalid. For example, a link re-negotiation might take a few seconds (normally 2 to 3 or might be up to 9). Thus, the link must be re-established after a short time.
- Typically, this timeout is recommended to be three seconds.
- Even when enabling and using AENs, periodically poll the link status, as dropped AENs might not be detected.

38.37.6.9.5 Statistics

The MC might use the statistics commands as defined in NC-SI. These counters are intended for debug purposes and are not all supported.

The statistics are divided into three commands:

1. Controller statistics — These are statistics on the network interface (to the host operating system and the PT traffic). See the NC-SI specification for details.
2. NC-SI statistics — These are statistics on the NC-SI control frames (such as commands, responses, AENs, etc.). See the NC-SI specification for details.
3. NC-SI PT statistics — These are statistics on the NC-SI PT frames. See the NC-SI specification for details.

38.37.6.10 External Link Control

The MC can use the NC-SI Set Link command to control the external interface link settings. This command enables the MC to set the auto-negotiation, link speed, duplex, and other parameters.

This command is only available when the host operating system is not present. Indicating the host operating system status can be obtained via the Get Link Status command and/or Host OS Status Change AEN command.

Recommendation:

- Unless explicitly needed, it is not recommended to use this feature. The NC-SI Set Link command does not expose all the possible link settings and/or features. This might cause issues under different scenarios. Even if you decided to use this feature, use it only if the link is down (trust the 10 GbE controller until proven otherwise).
- It is recommended that the MC first query the link status using the Get Link Status command. The MC should then use this data as a basis and change only the needed parameters when issuing the Set Link command.

For details, refer to the NC-SI specification.

38.37.6.10.1 Set Link While LAN PCIe Functionality is Disabled

In cases where the 10 GbE controller is used solely for manageability and its LAN PCIe function is disabled, using the NC-SI Set Link command while advertising multiple speeds and enabling auto-negotiation results in the lowest possible speed chosen.

To enable a higher link speed, the MC should not advertise speeds that are below the desired link speed, as the lowest advertised link speed is chosen.

When the 10 GbE controller is only used for manageability and the link speed advertisement is configured by the MC, changes in the power state of the LAN device is not effected and the link speed is not re-negotiated by the LAN device.

38.37.7 Management Component Transport Protocol (MCTP)

38.37.7.1 MCTP Overview

MCTP defines a communication model intended to facilitate communication between:

- MCs and other MCs
- MCs and management devices

The communication model includes a message format, transport description, message exchange patterns, and configuration and initialization messages.

The basic MCTP specification is described in DMTF's DSP0236 document.

MCTP is designed so that it can potentially be used on many bus types. The protocol is intended to be used for intercommunication between elements of platform management subsystems used in computer systems, and is suitable for use in mobile, desktop, workstation, and server platforms.

Currently, specifications exists for MCTP over PCIe (DMTF's DSP0238) and over SMBus (DMTF's DSP0237). A specification for MCTP over USB is also planned.

MCs such as a Baseboard Management Controller (BMC) can use this protocol for communication between one another, as well as for accessing management devices within the platform.

38.37.7.1.1 NC-SI Over MCTP

MCTP is a transport layer protocol that does not include the functionality required to control the PT traffic required for an MC connection to the network. This functionality is provided by encapsulating NC-SI traffic as defined in DMTF's DSP0222 document.

The details of NC-SI over MCTP protocol are defined in the DMTF's DSP0261 - NC-SI Over MCTP Specification.

An NC-SI over MCTP implementation guide can be found in the DMTF's DSP0219 white paper.

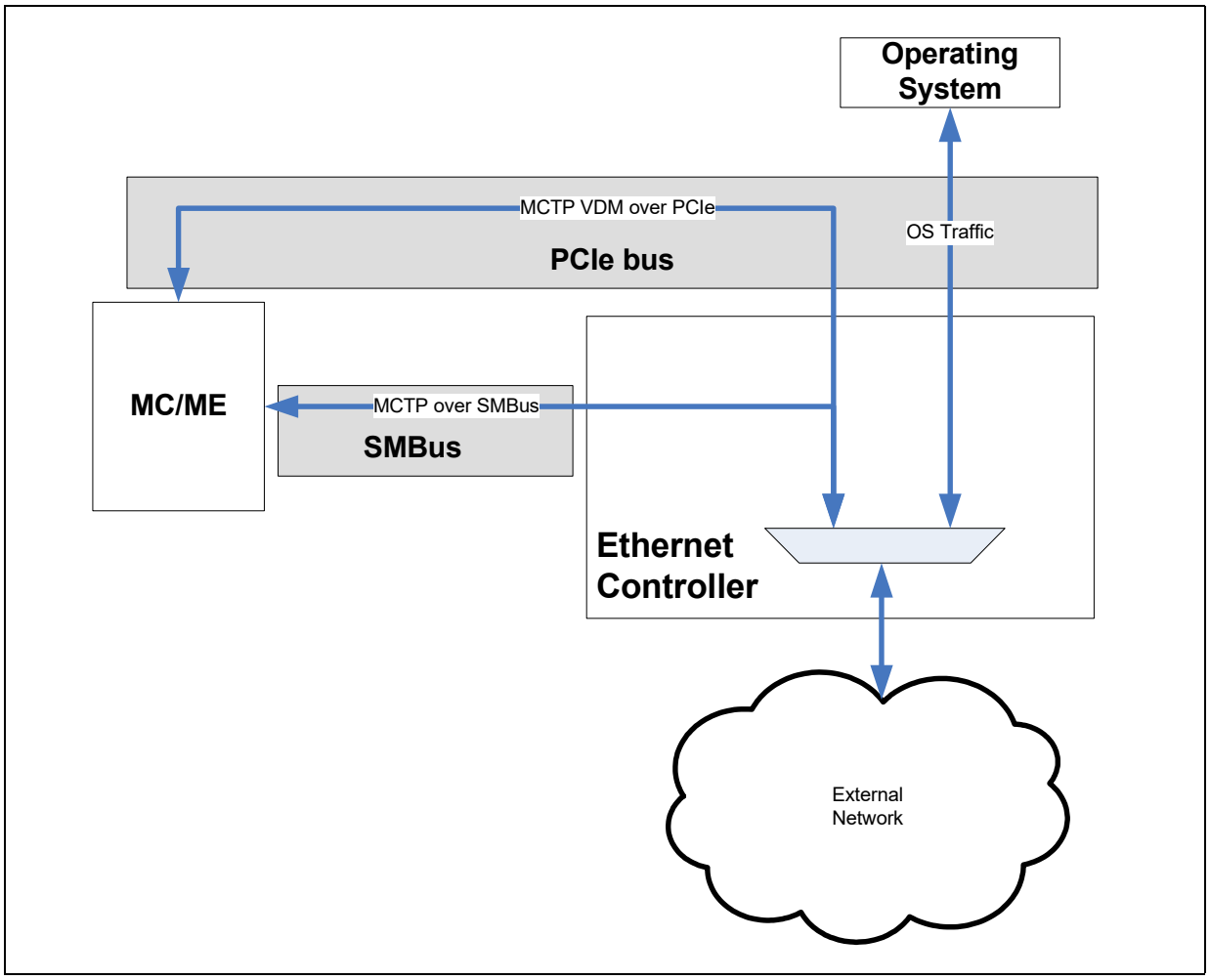
The NC-SI over MCTP specification defines two types of MCTP message types: NC-SI (0x2) and Ethernet (0x3). The 10 GbE controller supports both messages. When used only for control, then only the NC-SI (0x2) message type is supported.

Refer to the section about NC-SI over MCTP.

38.37.7.1.2 MCTP Usage Model

The 10 GbE controller supports NC-SI over MCTP protocol over the PCIe and SMBus busses. The 10 GbE controller can connect through MCTP to a MC or the Intel ME engine in the chipset as described in Figure 38-137.

Figure 38-137.10 BGBE Controller MCTP Connections



38.37.7.2 NC-SI to MCTP Mapping

The four network ports of the 10 GbE controller (mapped to two NC-SI channels) are mapped to a single MCTP endpoint on SMBus and to another endpoint over PCIe.

The PCIe endpoint is mapped to a PCIe requester ID according to the following flow:

1. If the *Bus Master Enable* bit of at least one of the functions is set, the endpoint is mapped to function the first available function.
2. If the *Bus Master Enable* bits of all functions are cleared, the MCTP endpoint on PCIe is not exposed and the MCTP traffic is routed through the SMBus endpoint.

The slave address used for the SMBus endpoint is the slave address of the first port.

Section 38.37.7.2.1 describes the transition between the two busses.

Both endpoints (SMBus and PCIe) might be active concurrently. However, PT traffic might be transferred only through one of them. If the PCIe endpoint is active, it is used for PT traffic; otherwise, the SMBus endpoint is used. The Set EID command can be used to force the transition for the PCIe endpoint to the SMBus endpoint if the bus owner determines the PCIe channel is not functional.

For each channel (SMBus or PCIe), the 10 GbE controller should expect MCTP commands from two sources: the bus owner and the MC. In addition, it should expect PT traffic through one interface only. Thus, it should be able to process up to five interleaved commands/data:

- An MCTP control/OEM command from the PCIe bus owner (single packet message).
- An MCTP control/OEM command from the SMBus bus owner (single packet message).
- An MCTP control/OEM command from the MC over SMBus (single packet message).
- An MCTP control/OEM command from the MC over PCIe (single packet message).
- An NC-SI command or Ethernet packet from the MC over the active channel.

A single source should not interleave packets it sends.

The topology used for MCTP connection is shown in Figure 38-138.

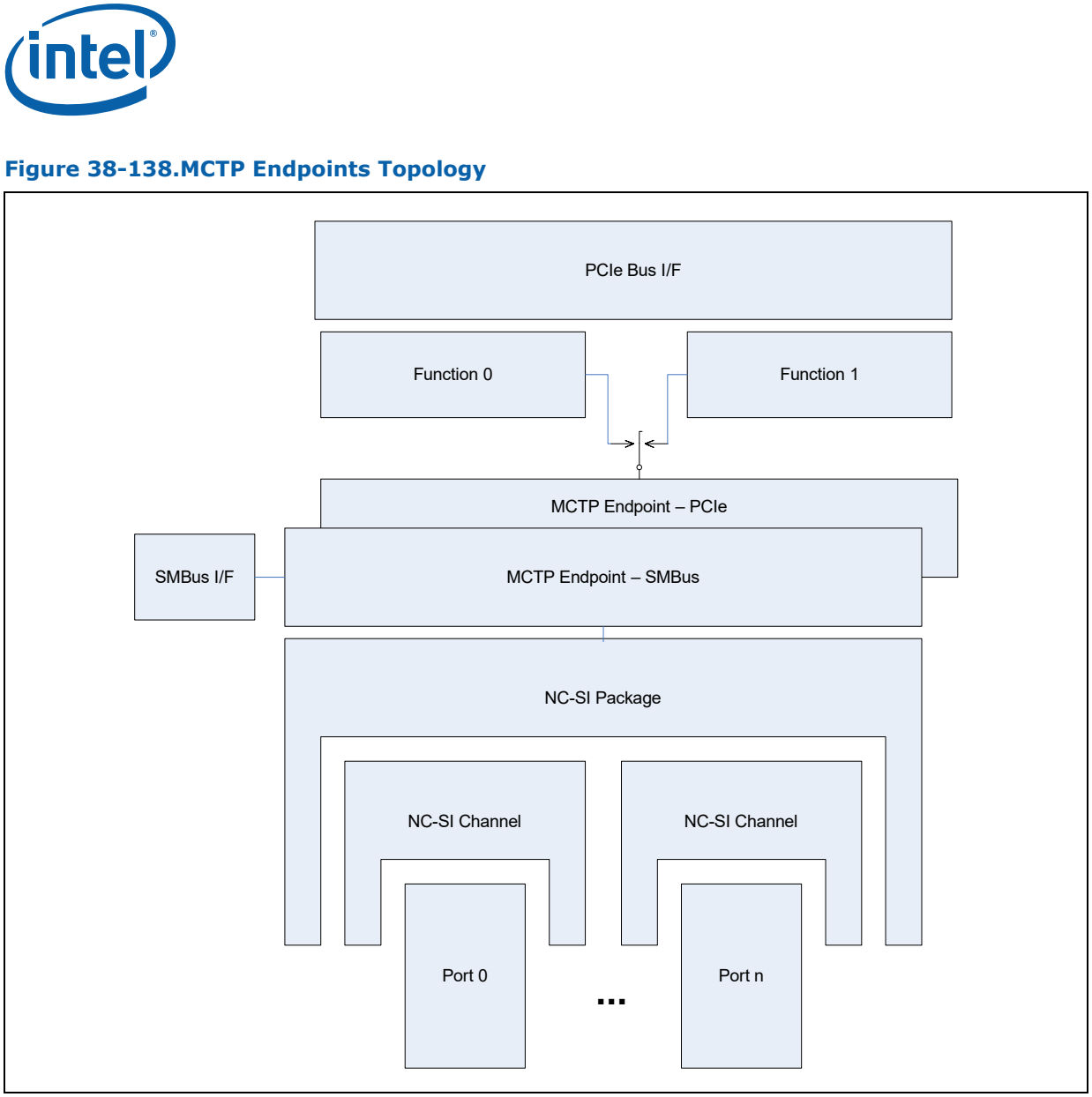


Figure 38-138.MCTP Endpoints Topology

38.37.7.2.1 Detecting an MC EID and Physical Address

In order to enable transactions between the MC and the NIC, the bus physical address (SMBus or PCIe) and the EID of the partner needs to be discovered. NICs do not try to discover the MC and assume the MC initiates the connection. If the NIC is in an NC-SI initial state, then the EID and the physical address of the MC are extracted from the Clear Initial State command parameters or any other NC-SI command received later with a channel ID of the 10 GbE controller. Subsequent PT traffic is received from or sent to this address only.

If the EID or the physical address of the NIC changes, it indicates the changes to bus owner so that the routing tables can be updated. There is no attempt to directly send an indication to the MC about the change.

See more details in next section.

38.37.7.2.2 Bus Transition

The following section defines the transition flow between PCIe and SMBus as the bus on which MCTP flows. Figure 38-139 describes the flow to transition between PCIe and SMBus. The following parameters are used to define the flow:

- NIC EID on PCIe
- NIC EID on SMBus
- NIC PCIe Target ID
- Bus Owner EID on PCIe
- Bus Owner EID on SMBus
- Bus Owner PCIe Target ID
- Bus Owner SMBus Address
- MC EID on PCIe
- MC EID on SMBus
- MC PCIe Target ID
- MC SMBus Address
- NIC SMBus Address

All these variables are initialized to zero at power on apart from the SMBus address of the endpoint (NIC), which might be initialized from a NVM value.

Initial Assignment Flow

- At power on, the NIC or MC MCTP channel is connected to the SMBus, is not assigned an EID and is in an undiscovered state.
- The bus owner might preform an SMBus ARP cycle to assign an SMBus address to the NIC or to the MC. Otherwise, a fixed address might be used. It is assumed that the SMBus address does not change after initialization time.
- The bus owner performs an EID assignment using a Set Endpoint ID MCTP command. The NIC or the MC captures the SMBus address of the bus owner from the SMBus Source Slave address field, the bus owner EID from the Source Endpoint ID field and the NIC/MC EID from the Destination Endpoint ID field in the MCTP header as described in Section 10.3 of *DSP0236*. The NIC/MC is now in a discovered state.
- The MC might detect the NIC EID using one of the two following modes:
  - Static configuration of the NIC SMBus address in the MC database and Get Routing Table Entries command to find the EID matching the SMBus address.
  - Get all endpoints through a Get Routing Table Entries command and find endpoints supporting NC-SI using the Get Message Type Support command for each endpoint.
- Once the NIC is found, the MC might send a Clear Initial State command to the NIC to start the NC-SI configuration. The NIC captures the MC SMBus address and MC EID from any NC-SI command received.
- After the NC-SI channels are enabled, traffic might be sent using the MC and NIC addresses previously discovered.
- The MC might also send a Get UUID command to get a unique identifier of the NIC that might be used later for re-connection upon topology changes.

SMBus to PCIe Transition

- If the NIC or the MC detects that the PCIe bus is available by detecting a function that moved to D0 state, it might request a transition using a Discovery Notify MCTP command on the PCIe bus. This command should be sent with a route to root-complex addressing as described in the *MCTP PCIe VDM Transport Binding Specification (DSP0238)* Section 6.8. The source EID should be the EID previously assigned on the SMBus.
- After receiving the Discovery Notify MCTP command on the PCIe bus, the bus owner sends a Set Endpoint ID MCTP command on the PCIe bus and updates the routing table. The bus owner might choose to wait for the Discovery Notify MCTP command of both the MC and the NIC to do the transition. The bus owner should try to keep the EID previously assigned on the SMBus as the EID on PCIe bus.
- After receiving the Set Endpoint ID MCTP command, the NIC waits for an NC-SI command from the MC indicating it is ready to transition the connection to PCIe. After receiving such a command, the NIC transitions its PT traffic to the PCIe bus using the newly received addresses.
- The MC on its side, needs to discover the PCIe address of the NIC. This can be done using the Resolve Endpoint ID command if only the physical address changed or using the Resolve Endpoint UUID command also if both EID and physical address changed. It can then send an NC-SI command to the NIC to initiate the transition. The MC should not send any PT packets from the moment it sent the first NC-SI command on the PCIe and the moment a response is received for this command.
- The transition of NC-SI traffic (PT or commands/responses) from SMBus to PCIe should be done on a packet boundary and should not interrupt a packet fragmentation or reassembly.

PCIe Target ID Change

The target ID of one of the endpoints might change due to a new enumeration of the PCIe bus. In this case the following flow should be used:

The target ID of one of the endpoints might change, either due to a new enumeration of the PCIe bus or due to the disabling of one of the functions in the device (move to a non D0 state). In this case, the following flow should be used:

- The endpoint should send a Discovery Notify MCTP command on the PCIe bus using the new Requester ID.
- After receiving the Discovery Notify MCTP command with the new requester ID, the bus owner sends a Set Endpoint ID MCTP command on the PCIe bus and updates the routing table. The bus owner should try to keep the EID previously assigned on the SMBus as the EID on the previous requester ID.
- The bus owner sends an Routing Information Update command to all supporting endpoints that might then update the parameters of their counterpart they use.

PCIe to SMBus Transition

- If the NIC or the MC detects that the PCIe bus is not available by detecting a transition of all functions to a non D0 state, it stops using the PCIe for PT traffic or NC-SI traffic.
- Upon detection of the unavailability of the PCIe bus, the MC transitions the NC-SI channel to the MCTP over SMBus as previously described.
- The transition of NC-SI traffic (PT or commands/responses) from PCIe to SMBus might done at any stage and might interrupt a packet fragmentation or reassembly,

as it is assumed that such a transition occurs only when the PCIe bus is not available anymore.

38.37.7.3 MCTP Over PCIe

38.37.7.3.1 Message Format

The message format used for NC-SI over MCTP over PCIe is as follows:

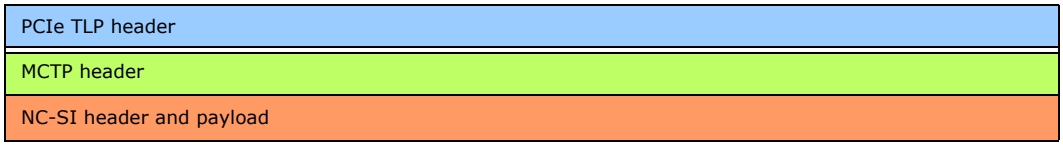


Table 38-596.NC-SI/Ethernet Over MCTP Over PCIe Message Format

+0								+1								+2								+3							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
FMT 011			Type 10r2r1r0[1]					R	TC 000			R	Attr	R	TH	TD	EP	Attr [1:0]		AT 00		Length 00_000x_xxxx											
PCI Requester ID																PCI Tag Field								Message Code Vendor Defined = 0111_1111b							
																R		Pad Len		MCTP VDM code - 0000b											
PCI Target ID (For Route by ID messages, otherwise = Reserved)																Vendor ID = 0x1AB4 (DMTF)															
MCTP Reserved				Header version = 1				Destination endpoint ID								Source endpoint ID								SOM	EOM	SEQ#			TO	Tag			
IC[2]	Message Type = 0x02/ 0x03							NC-SI Command/Pass Through data																							
.....																															
NC-SI Command/Pass Through data																															

Notes:

1. r2r1r0 =.
000b: Route to Root Complex.
010b: Route by ID.
011b: Broadcast from Root Complex.
2. TD = 0, EP = 0, IC = 0,TH = 0, Attr[2:0] = 0 for sent packets and is ignored for received packets.

38.37.7.3.2 PCIe Discovery Process

The 10 GbE controller follows the discovery process described in Section 5.9 of the *MCTP PCIe VDM Transport Binding Specification (DSP0238)*.

After receiving an endpoint discovery message (while in undiscovered stage), the 10 GbE controller exposes the endpoint on the selected function as previously described.

If the selected function moves after the endpoint was discovered, or if the bus number of the 10 GbE controller changes due to a re-enumeration of the bus, the 10 GbE controller sends a discovery notify message to indicate to the MC that it should do a re-enumeration of the device to discover the new endpoint.

38.37.7.3.3 MCTP Over PCIe Special Features

The 10 GbE controller supports the following optional features of MCTP when running over PCIe: rate limiting.

MCTP Uplink Rate Limiting

As the PCIe link can carry a traffic bandwidth much higher than what the MC can sustain, in order to avoid drop of packets, the 10 GbE controller allows rate limiting of the MCTP PT traffic. The 10 GbE controller supports rate limiting between 1 Mb/s and 1 Gb/s. The following parameters define the behavior of the rate limiter:

- Max rate limit (fixed from NVM via the MCTP rate in the MCTP Rate Limiter Config 1 word).
- The max burst size (fixed from NVM via the MCTP *Max Credits* field in the in the MCTP Rate Limiter Config 2 word). To limit the max burst to one VDM, set this parameter to five.
- Decision point (fixed from NVM via the *Decision Point* field in the in the MCTP Rate Limiter Config 2 word).

38.37.7.4 MCTP Over SMBus

The message format used for NC-SI over MCTP over SMBus is as follows:

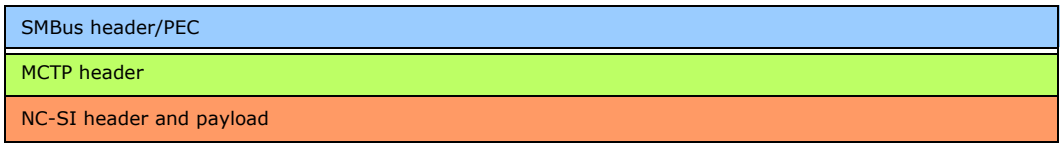


Table 38-597.NC-SI/Ethernet Over MCTP Over SMBus Message Format

+0								+1								+2								+3							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
Destination Slave Address							0	Command Code = MCTP = 0Fh								Byte count								Source Slave Address							1
MCTP Reserved				Header version = 1				Destination endpoint ID								Source endpoint ID								SOM	EOM	SEQ#			TO	Tag	
IC[1]	Message Type = 0x02/ 0x03							NC-SI Command/Pass Through data																							
.....																															
NC-SI Command/Pass Through data																															
PEC																															

Notes:

1. IC = 0.

38.37.7.4.1 SMBus Discovery Process

The 10 GbE controller follows the discovery process described in Section 6.5 of the *MCTP SMBus/I²C Transport Binding Specification (DSP0237)*. It indicates support for ASF in the SMBus getUID command (see Section 38.37.5.5.4). It responds to any SMBus command using the MCTP command code. This ensures that the bus owner knows the 10 GbE controller supports MCTP.

Note: MCTP commands over SMBus are received from any master address and are answered to the sender. There is no capturing of the bus owner address from any specific command.

38.37.7.4.2 MCTP Over SMBus Special Features

The 10 GbE controller supports the following optional feature of MCTP when running over SMBus:

1. Simplified MCTP mode.
2. Fairness arbitration.

Simplified MCTP Mode

For some point-to-point implementations of MCTP, the assembly process is simplified. In this mode, the destination EID, source EID, packet sequence number, Tag Owner (TO) bit and message tag are ignored and the assembly is based only on the *SOM* and *EOM* bits. This bit is set according to the *Simplified MCTP* bit in the MCTP configuration word in the NVM.

Note: This mode is not compliant with the MCTP specification.

In this mode, a Set EID command is not needed to start operation.

This mode is relevant only for MCTP over SMBus traffic and when the redirection sideband interface is set to 10b (MCTP over SMBus only - no pass through).

Fairness Arbitration

When sending MCTP messages over SMBus and when fairness arbitration is enabled, the 10 GbE controller should adhere to the fairness arbitration as defined in Section 5.13 of *DSP0237* when sending MCTP messages.

38.37.7.5 NC-SI Over MCTP

The 10 GbE controller support for NC-SI over MCTP is similar to the support for NC-SI over RBT with the following exceptions:

1. A set of new NC-SI OEM commands used to expose the NC-SI over MCTP capabilities.
2. The format of the packets is modified to account for the new transport layer as described in the sections that follow.

38.37.7.5.1 NC-SI Packets Format

NC-SI over MCTP defines two different message type for pass through and for control packets.

Packets with a message type equal to the *Control packets message type* field (default = 0x02) in the NVM are NC-SI control packets (commands, responses and AENs) and packets with a message type equal to the *Pass through packets message type* field (default = 0x03) in the NVM are NC-SI pass-through packets.

Control Packets

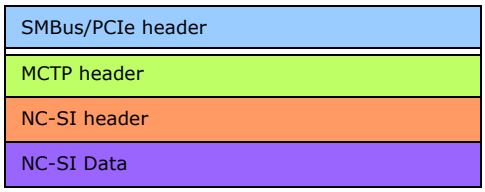
The format used for control packets (commands, responses and AENs) is as follows:

Table 38-598.NC-SI Over MCTP Over PCIe/SMBus Message Format

+0								+1								+2								+3							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
SMBus or PCIe header																															
MCTP Reserved				Header version = 1				Destination endpoint ID								Source endpoint ID								SOM		EOM		SEQ#			TO = 1	Tag	
IC = 0	Message Type = Control Packets Message type (0x02)							MC ID = 0x00								Header revision								Reserved							
IID								Command								Channel ID[1]								Reserved							
Payload Length[7:0]								Reserved																Payload Length[11:8]							
Reserved																															
Reserved								Command Data																							
....																															
Command Data																Checksum															
Checksum																															

Notes:

1. The channel ID is defined as described in Section 38.37.2.2.2.



Note that the MAC header and MAC FCS present when working over NC-SI are not part of the packet in MCTP mode.

PT Packets

The format used for PT packets are as follows. This format is the same for either packets received from the network or packets received from the host.

The CRC is never included in the packet. In receive, the CRC is checked and removed by the 10 GbE controller in transmit, the CRC is added by the 10 GbE controller.

Table 38-599.Ethernet Over MCTP Over PCIe/SMBus Message Format

+0								+1								+2								+3							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
SMBus or PCIe header																															
MCTP Reserved				Header version = 1				Destination endpoint ID								Source endpoint ID								SOM	EOM	SEQ#			TO = 1	Tag	
IC = 0	Message Type = Pass Through Packets Control Type							DA																							
DA																								SA							
SA																															
SA								Ether type																Ethernet Packet							
Ethernet packet																															
....																															
....																															

38.37.7.6 MCTP Programming

The MCTP programming model is based on:

1. A set of MCTP commands used for the discovery process and for the link management. The list of supported commands is described in Section 38.37.7.6.1.
2. A subset of the NC-SI commands used in the regular NC-SI interface, including all the OEM commands as described in Section 38.37.6.2 (NC-SI programming I/F). The specific commands supported are listed in Table 38-546 and Table 38-550.

Note: For all MCTP commands (both native MCTP commands and NCSI over MCTP), the response uses the Msg tag received in the request with *TO* bit cleared.

38.37.7.6.1 MCTP Commands Support

Table 38-600 lists the MCTP commands supported by the 10 GbE controller.

Table 38-600.MCTP Commands Support (Sheet 1 of 2)

Command Code	Command Name	General Description	10 GbE Controller Support As Initiator	10 GbE Controller Support As Responder
0x00	Reserved	Reserved	–	–
0x01	Set Endpoint ID	Assigns an EID to the endpoint at the given physical address.	N/A	Yes
0x02	Get Endpoint ID	Returns the EID presently assigned to an endpoint. Also returns information about what type the endpoint is and its level of use of static EIDs.	No	Yes
0x03	Get Endpoint UUID	Retrieves a per-device unique UUID associated with the endpoint.	No	Yes
0x04	Get MCTP Version Support	Lists which versions of the MCTP control protocol are supported on an endpoint.	No	Yes
0x05	Get Message Type Support	Lists the message types that an endpoint supports.	No	Yes
0x06	Get Vendor Defined Message Support	Used to discover an MCTP endpoint's vendor specific MCTP extensions and capabilities.	No	Yes

Table 38-600.MCTP Commands Support (Sheet 2 of 2)

Command Code	Command Name	General Description	10 GbE Controller Support As Initiator	10 GbE Controller Support As Responder
0x07	Resolve Endpoint ID	Used to get the physical address associated with a given EID.	No	N/A
0x08	Allocate Endpoint IDs	Used by the bus owner to allocate a pool of EIDs to an MCTP bridge.	N/A	N/A
0x09	Routing Information Update	Used by the bus owner to extend or update the routing information that is maintained by an MCTP bridge.	N/A	N/A
0x0A	Get Routing Table Entries	Used to request an MCTP bridge to return data corresponding to its present routing table entries.	No	N/A
0x0B	Prepare for Endpoint Discovery	Used to direct endpoints to clear their discovered flags to enable them to respond to the Endpoint Discovery command.	N/A	Yes[1]
0x0C	Endpoint Discovery	Used to discover MCTP-capable devices on a bus, provided that another discovery mechanism is not defined for the particular physical medium.	No	Yes
0x0D	Discovery Notify	Used to notify the bus owner that an MCTP device has become available on the bus.	Yes	N/A
0x0E	Get Network ID	Used to get the MCTP network ID.	No	No
0x0F	Query Hop	Used to discover what bridges, if any, are in the path to a given target endpoint and what transmission unit sizes the bridges will pass for a given message type when routing to the target endpoint.	No	No

Notes:

1. These commands are supported only for MCTP over PCIe.

Get Endpoint ID

The get endpoint ID response of the 10 GbE controller is listed in the following table.

Byte	Description	Value
1	Completion Code	
2	Endpoint ID	0x00 = EID not yet assigned. Otherwise = Returns EID assigned using Set Endpoint ID command.
3	Endpoint Type	0x00 (Dynamic EID, simple endpoint).
4	Medium Specific	SMBus = 0x01 - Fairness arbitration protocol supported. PCIe = 0x00.

Get Endpoint UUID

The UUID returned is calculated according to the following function:

- Time Low = Read from NVM words at offset 0x9 and 0xA of the sideband configuration structure.
- Time mid = Read from NVM word at offset 0xB of the sideband configuration structure.
- Time High and version = Read from NVM word at offset 0xC of the sideband configuration structure.
- Clock Sec and Reserved = Read from NVM word at offset 0xD of the sideband configuration structure.
- Node = MAC address as taken from the GLPCI_SERL and GLPCI_SERH registers.

Get MCTP Version Support

The following table lists the returned value according to the requested message type.

Byte	Description	Message Type					
		0xFF(Base)	0x00 (Control Protocol Message)	0x02 (NC-SI Over MCTP)	0x03 (Ethernet)	0x7E (PCIe Based VDM Messages)	All Other Or Unsupported Messages
1	Completion Code	0x0					0x80
2	Version Number Entry Count	3	3	1	1	2	0
6:3	Version Number Entry	0xF1F0FF00 (1.0)	0xF1F0FF00 (1.0)	0xF1F0FF00 (1.0)	0xF1F0FF00 (1.0)	0xF1F0FF00 (1.0)	0
9:7	Version Number Entry 2	0xF1F1F000 (1.1.0)	0xF1F1F000 (1.1.0)			0xF1F1F000 (1.1.0)	
13:10	Version Number Entry 3	0xF1F2F000 (1.2.0)	0xF1F2F000 (1.2.0)				

Get Message Type Support Command

The get message type support response of the 10 GbE controller is listed in the following table.

Byte	Description	Value
1	Completion Code	0x00.
2	MCTP Message Type Count	0x01/0x02/0x03 - The 10 GbE controller supports up to three additional message types, depending on the mode of operation and the bus used.
3:5	List of Message Type Numbers	0x02 (NC-SI over MCTP).
		0x03 (Ethernet).
		0x7E (PCIe based VDM messages) - over PCIe only.

Get Vendor Defined Message Support Command

The get vendor defined message type support response of Product Name is listed in the following table if vendor ID set selector equals 0x00:

Byte	Description	Value
1	Completion Code	0x00.
2	Vendor ID Set Selector	0xFF = No more capability sets.
2:4	Vendor ID	0x008086 (PCI ID indicator + Intel vendor ID).
5:6	Version	0x0100 (version 1.0).

Set Endpoint ID Command

The 10 GbE controller supports the set EID and force EID operations defined in the Set Endpoint ID command. When operating over PCIe, the set discovered flag operation is also supported. As endpoints in the 10 GbE controller can be set only through their own interface, set EID and force EID are equivalent. The Reset EID operation is not supported by the 10 GbE controller.

The Set Endpoint ID response of the 10 GbE controller is described in the following table.

Byte	Description	Value
1	Completion Code	0x00.
2	Completion Status	[7:6] = 00b - Reserved.
		[5:4] = 00b - EID assignment accepted.
		[3:2] = 00b - Reserved.
		[1:0] = 00b - Device does not use an EID pool.
3	EID Setting	If the EID setting was accepted, this value matches the EID passed in the request. Otherwise, this value returns the present EID setting.
4	EID Pool Size	Always return a zero.

38.37.7.7 Host Isolate Support

If a MC decides that malicious software prevents its usage of the LAN, it might decide to isolate the NIC from its driver. This is done using the TCO reset command (Section 38.37.6.4.12).

If TCO isolate is enabled in the NVM, The TCO Isolate command disables PCIe write operations to the LAN port. As the software device driver needs to access the CSR space in order to provide descriptors to the NIC, this operation also stops the network traffic including OS2BMC and MC-to-OS traffic as soon as the existing transmit and receive descriptor queues are exhausted.

MCTP over PCIe VDM are still available in this mode.

38.38 Programming Interface

This section details the programmer visible state inside the 10 GbE controller. In some cases, it describes hardware structures invisible to software in order to clarify a concept.

The 10 GbE controller address space is mapped into four regions with PCI Base Address registers. These regions are listed in the following table.

Table 38-601.Address Space Regions

Addressable Content	How Mapped	Size of Region
Memory BAR (Internal registers, memories and Flash)	Direct memory-mapped	4 MB - 16 MB
I/O BAR (optional Internal registers)	I/O Window mapped	32 bytes[1]
MSI-X BAR (optional)	Direct memory-mapped	32 KB
Expansion ROM BAR (optional)	Direct memory-mapped	64 KB - 8 MB

Notes:

1. The internal registers can be accessed though I/O space indirectly as explained in the sections that follow.

Rules for unsupported accesses:

- Accesses to non-implemented or disabled regions within a BAR are dropped for write accesses or responded with arbitrary data for read accesses. A PCIe error event is not generated and completions return with successful status.
- Refer to the section about supported PCIe access sizes to each of the BARs and its components.

38.38.1 Access Mechanisms

38.38.1.1 Memory-Mapped Access to Internal Registers and Memories

The internal registers and memories might be accessed as direct memory-mapped offsets from the Base Address register (BAR0 or BAR 0/1). Refer to the section about the appropriate offset for each specific internal register.

In IOV mode, this area is partially duplicated per VF. All replications contain only the subset of the register set that is available for VF programming. Refer to the section about the appropriate offset for each specific internal register for VFs.

38.38.1.2 Memory-Mapped Access to Flash

The external Flash can be accessed using direct memory-mapped offsets from the memory base address register (BAR0 in 32-bit addressing or BAR0/BAR1 in 64-bit addressing). See Table 38-604 for the location of Flash memory within the memory BAR. Access to Flash memory is restricted to the first 64 KB when GLPCI_LBARCTRL.FLASH_EXPOSE is set to 0b.

38.38.1.3 Memory-Mapped Access to MSI-X Tables

The MSI-X tables can be accessed as direct memory-mapped offsets from the base address register (BAR3 or BAR3/4). Refer to the section about the appropriate offset for each specific internal MSI-X register.

In IOV mode, this area is duplicated per VF. It requires a memory space of the maximum between 16 KB and the page size.

38.38.1.4 Memory-Mapped Access to Expansion ROM

The external Flash can also be accessed as a memory-mapped expansion ROM. Accesses to offsets starting from the expansion ROM base address reference the Flash provided that access is enabled from NVM, and if the expansion ROM base address register contains a valid (non-zero) base memory address.

38.38.1.5 I/O-Mapped Access to internal registers

To support pre-boot operation, all internal registers can be accessed using I/O operations. I/O accesses are supported only if an I/O base address is allocated and mapped (BAR2), and I/O address decoding is enabled in the PCIe configuration.

When an I/O BAR is mapped, the I/O address range allocated opens a 32-byte window in the system I/O address map. Within this window, two I/O addressable registers are implemented: IOADDR and IODATA. The IOADDR register is used to specify a reference to an internal Register and then the IODATA register is used as a window to the register address specified by IOADDR as listed in Table 38-602.

Table 38-602.IOADDR and IODATA in I/O Address Space

Offset	Abbreviation	Name	RW	Size
0x00	IOADDR	Internal Register Address. Covers the (4 MB - 64 KB) CSR space 0x00000-0x3EFFFF – Internal Registers. 0x3F0000-0xFFFFFFFF – Undefined.	RW	4 bytes
0x04	IODATA	Data field for reads or writes to the Internal Register location as identified by the current value in IOADDR. All 32 bits of this register can be read from and written to.	RW	4 bytes
0x08 – 0x1F	Reserved	Reserved.	RO	4 bytes

38.38.1.5.1 IOADDR (I/O Offset 0x00)

The IOADDR register must always be written as a Dword access. Refer to the section that describes how other access sizes are handled.

For IA programmers, the IN and OUT instructions must be used to cause I/O cycles to be used on the PCIe bus. Because writes must be to a 32-bit quantity, the source register of the OUT instruction must be EAX (the only 32-bit register supported by the OUT command). For reads, the IN instruction can have any size target register, but it is recommended that the 32-bit EAX register be used.

At hardware reset (RSMRST#) or PCI reset, this register value resets to 0x00000000. Once written, the value is retained until the next write or reset.

38.38.1.5.2 IODATA (I/O Offset 0x04)

The IODATA register must always be written as a Dword access (assuming the IOADDR register contains a value for the internal register space). Reads to IODATA returns a Dword of data. Refer to the section that describes how other access sizes are handled.

For software programmers, the IN and OUT instructions must be used to cause I/O cycles to be used on the PCIe bus. Where 32-bit quantities are required on writes, the source register of the OUT instruction must be EAX (the only 32-bit register supported by the OUT command).

Note: There are no special software timing requirements on accesses to IOADDR or IODATA. All accesses are immediate, except when data is not readily available or acceptable. In this case, the 10 GbE controller delays the results through normal bus methods (for example, split transaction or transaction retry).

Note: Because a register read or write takes two I/O cycles to complete, software must provide a guarantee that the two I/O cycles occur as an atomic operation. Otherwise, results can be non-deterministic from the software viewpoint.

38.38.1.5.3 Undefined I/O Offsets

I/O offsets 0x08 through 0x1F are considered to be reserved offsets with the I/O window.

38.38.1.6 Configuration Access to Internal Registers

To support legacy pre-boot 16-bit operating environments without requiring I/O address space, the 10 GbE controller enables accessing CSRs via configuration address space by mapping the *IOADDR* and *IODATA* registers into configuration address space. If the GLPCI_CAPSUP.CSR_CONF_EN bit is set to 1b, access to CSRs via configuration address space is enabled. The register mappings in this case are listed in Table 38-603.

Table 38-603.IOADDR and IODATA in Configuration Address Space

Configuration Address	Abbreviation	Name	RW	Size
0x98	IOADDR	Internal Register Address. Covers the (4 MB - 64 KB) CSR space. 0x00000-0x3EFFFF – Internal registers. 0x3F0000-0x7FFFFF – Undefined.	RW	4 bytes
0x9C	IODATA	Data field for reads or writes to the internal register location as identified by the current value in IOADDR. All 32 bits of this register can be read from and written to.	RW	4 bytes

Software writes data to an internal CSR via configuration space in the following manner:

1. CSR address is written to IOADDR where:
   a. Bit 31 (*IOADDR.Configuration IO Access Enable*) of IOADDR should be set to 1b.
   b. Bits 30:0 of IOADDR should hold the actual address of the internal register being written to.
2. Data to be written is written into IODATA.
   — IODATA is used as a window to the register address specified by IOADDR. As a result, the data written to IODATA is written into the CSR pointed to by bits 30:0 of IOADDR.
3. IOADDR is cleared (all bits [31:0]), to avoid un-intentional CSR read operations (that might cause clear by read) by other applications scanning the configuration space.

Software reads data from an internal CSR via configuration space in the following manner:

1. The CSR address is written to IOADDR where:
   a. Bit 31 (*IOADDR.Configuration IO Access Enable*) of IOADDR should be set to 1b.
   b. Bits 30:0 of IOADDR should hold the actual address of the internal register being read.
2. The CSR value is read from IODATA.
   a. IODATA is used as a window to the register address specified by IOADDR. As a result, the data read from IODATA is the data of the CSR pointed to by bits 30:0 of IOADDR.
3. IOADDR is cleared (all bits [31:0]), to avoid un-intentional CSR read operations (that might cause clear by read) by other applications scanning the configuration space.

Note: In the event that the GLPCI_CAPSUP.CSR_CONF_EN bit is cleared, accesses to IOADDR and IODATA via the configuration address space are ignored and have no effect on the register and the CSRs referenced by IOADDR.

Note: When a function is in D3 state, software should not attempt to access CSRs via IOADDR and IODATA.

Note: To enable CSR access via configuration space, software should set bit 31 (*IOADDR.Configuration IO Access Enable*) of IOADDR to 1b. Software should clear bit 31 of IOADDR after completing CSR access to avoid an unintentional clear by read operation, by another application scanning the configuration address space. Software should also clear bits 30:0 of IOADDR to remove any trace of previous accesses to the configuration space (see previous flows.

Note: Bit 31 of IOADDR (*IOADDR.Configuration IO Access Enable*) has no effect when initiating access via IO address space.

38.38.2 Memory BAR

38.38.2.1 PF BAR Structure

The memory BAR provides access to internal CSRs, to the protocol engine doorbells, and to the external Flash (NVM). This section describes where each of these is located within the BAR space.

The following configuration parameters define the structure of the PF memory BAR 0:

- *Flash_Expose* bit from the NVM (or the GLPCI_LBARCTRL.FLASH_EXPOSE CSR bit)
  - 0b = Flash memory is not mapped in the memory BAR
  - 1b = Flash memory is mapped in the memory BAR. Hardware default; Flash memory is exposed when during initialization the Flash is found to be blank or in error.
- Flash *Size* field from the NVM (or the GLPCI_LBARCTRL.FL_SIZE CSR field) - size is calculated as 64 KB * (2 ** Flash Size). Default value is 8 MB
- PE_DB_Size field from the NVM (or the GLPCI_LBARCTRL,PE_DB_SIZE CSR field) - determines the size of the memory space allocated to protocol engine doorbells
  - 00b = Memory space is not allocated for PE doorbells
  - 01b = A 64 KB area is allocated
  - 10b = A (4MB + 64KB)area is allocated
  - 11b = Reserved

Table 38-604 lists all supported partitions of the memory BAR as a function of the previously described parameters. Other combinations of the parameters (not covered in the table) are not supported and considered reserved for future expansion. The following rules apply:

- CSR space is located from the beginning of the BAR until address (4 MB-64 KB-1) (such as the first 4 MB - 64 KB)
- PE space (if exists) is located from address (4MB-64KB) and extends up to address 4MB or up to address 8MB (not including)
- The Flash space (if exposed) is always aligned to multiples of its size.

The default configuration of the memory BAR (like when the Flash is empty) is defined by hardware default values of the read-only GLPCI_LBARCTRL CSR. GLPCI_LBARCTRL is loaded from the NVM during normal operation (such as when Flash contents are valid).

Table 38-605.Structure of the PF Memory BAR

Flash Expose	Flash Size	PE_DB Size	PE Space		Flash Space		BAR Size
			Min Addr	Max Addr	Min Addr	Max Addr	
0	x	00	x	x	x	x	4 MB
"	"	01	4 MB-64 KB	4 MB	x	x	4 MB
"	"	10	4 MB-64 KB	8 MB	x	x	8MB
1	2MB	00	x	x	4 MB	6 MB	8 MB
"	"	01	4 MB-64 KB	4 MB	4 MB	6 MB	8 MB
"	"	10	4 MB-64 KB	8 MB	8 MB	10 MB	16 MB
1	4MB	00	x	x	4 MB	8 MB	8 MB
"	"	01	4 MB-64 KB	4 MB	4 MB	8 MB	8 MB
"	"	10	4 MB-64 KB	8 MB	8 MB	12 MB	16 MB
1	8MB	00	x	x	8 MB	16 MB	16 MB
"	"	01	4 MB-64 KB	4 MB	8 MB	16 MB	16 MB
"	"	10	4 MB-64 KB	8 MB	8 MB	1 6MB	1 6MB

38.38.2.2 VF BAR Structure

The VF memory BAR provides access to on-die CSRs and (optionally) to PE doorbells for VFs.

The GLPCI_LBARCTRL.VF_PE_DB_SIZE CSR field determines whether the PE doorbells are provided with a separate region within the BAR:

- 0b = Memory space is not allocated for PE doorbells
- 1b = A 64 KB area is allocated

Note: Although only 32 VFs supports RDMA, all the VFs requires the same memory space.

Table 38-606 lists all supported partitions of the memory BAR as function of the previously described parameters.

Table 38-606.Structure of the VF Memory BAR

VF_PE_DB Size	BAR Size
0	Max (64 KB, Page Size)
1	Max (128 KB, Page Size)

Table 38-607.Structure of the VF memory BAR

VF_PE_DB Size	Doorbells Space		BAR Size
	Min Addr	Max Addr	
0	x	x	Max (64 KB, Page Size)
1	64 KB	128 KB	Max (128 KB, Page Size)

38.38.3 MSI-X BAR

Refer to the section about the structure of the MSI-X BAR for a PF and VF.

38.38.4 CSR Organization and Mapping

This section describes how CSRs are mapped into the PF and VF memory BAR. This section does not apply to the following address space:

- The MSI-X BAR (defined per the PCI specifications)
- The memory space allocated to protocol engine doorbells

38.38.4.1 Mapping by Scope

Registers are associated with a scope. A scope is a set of attributes for a register that define which functions can access the register and how many instances exist for the register. The following table lists the different scopes.

Table 38-608.Scope Mapping

Scope	PF/VF	Quantity	Exposure	Comments
GL	PF	1	To all PFs.	
GLVF	PF, VF	1	To all PFs and VFs.	Registers are RO.
PRT	PF	4	Each PF has access to the registers of the port it is associated with.	Registers are shared by all PFs on a port.
PRTVF	PF, VF	4	Each PF or VF has access to the registers of the port it is associated with (for a VF, the port is the port the PF is associated with).	Registers are RO. Registers are shared by all PFs and VFs on a port.
PF	PF	16	Each PF has access to its copy only.	
VF[1]	PF, VF	128	Each PF has access to the registers of its VFs only. Each VF has access to its copy only.	
VP	PF	128	Each PF has access to the registers of its VFs only.	These registers control VF functionality.
Q	PF, VF	1536	Each PF has access to the registers allocated to it (including. its VFs). Each VF has access to the registers allocated to it.	
VSI	PF	384	All PFs.	Registers are shared by all PFs.
INTPF	PF	512	Each PF has access to the registers allocated to it[2].	
INTVF	PF, VF	512	Each VF has access to the registers allocated to it. Each PF has access to the registers of its VFs only.	
INTVP	PF	512	Each PF has access to the registers of its VFs only.	These registers handle VF interrupts.

Notes:

1. PE VF registers are a special case of VF registers. PE resources are provided only to a subset of the VFs (total of 32 VFs). However, all VFs have PE registers mapped in their address space. When a VF accesses a PE register, it is mapped into the internal address space as described here, followed by another translation into one of the 32 physical PE instances. This later translation is not described in this section.
2. The register descriptions in section Device Registers - PF and section Device Registers - VF list 512 instances per register.

38.38.5 Register Conventions

All registers in the 10 GbE controller are defined to be 32 bits, should be accessed as 32-bit Dwords, There are some exceptions to this rule:

- Register pairs where two 32-bit registers make up a larger 64-bit logical unit
- Accesses to Flash memory (via expansion ROM space, secondary BAR space, or the I/O space) might be byte, word or double word accesses. I/O accesses are limited to Dword accesses (see Section 38.38.1.5).
- Accesses to BAR0 of a VF might be byte, word or Dword accesses. 64-bit (Qword) accesses to this BAR are completed with an Completer Abort (CA) error.

- Access to the MSI-X BAR of the PFs and the VFs might be Dword or Qword accesses.

Reserved bit positions — Some registers contain certain bits that are marked as reserved. Writes to a reserved field must set the field to its initial value unless specified differently in the field description. Reads from registers containing reserved bits might return indeterminate values in the reserved bit-positions unless read values are explicitly stated. When read, these reserved bits should be ignored by software.

Reserved and/or undefined addresses — Any register address not explicitly declared in this AN should be considered to be reserved, and should not be written to. Writing to reserved or undefined register addresses might cause indeterminate behavior. Reads from reserved or undefined configuration register addresses might return indeterminate values unless read values are explicitly stated for specific addresses.

Initial values: most registers define the initial hardware values prior to being programmed. In some cases, hardware initial values are undefined and is listed as such via the text undefined, unknown, or X. Such configuration values might need to be set via NVM configuration or via software in order for proper operation to occur; this need is dependent on the function of the bit. Other registers might cite a hardware default which is overridden by a higher-precedence operation. Operations that might supersede hardware defaults might include a valid NVM load, completion of a hardware operation (such as hardware auto-negotiation), or writing of a different register whose value is then reflected in another bit.

For registers that should be accessed as 32-bit Dwords, partial writes (less than a 32-bit Dword) does not take effect (the write is ignored). Partial reads returns all 32 bits of data regardless of the byte enables.

Note: Partial reads to clear-on-read registers (ICR) can have unexpected results since all 32 bits are actually read regardless of the byte enables. Partial reads should not be done.

Note: All statistics registers are implemented as 32-bit registers. Though some logical statistics registers represent counters in excess of 32 bits in width, registers must be accessed using 32-bit operations (for example, independent access to each 32-bit field). When reading 64-bit statistics registers the least significant 32-bit register should be read first.

See special notes for VLAN filter table, multicast table arrays and packet buffer memory that appear in the specific register definitions.

38.38.6 Register Terminologies

The following table lists the access type of registers' bit fields. The access rights of the PFs and the VFs to the entire registers are defined per PF and VF registers. In some cases, the PFs and VFs might have Read Only (RO) access rights to registers that can be programmed by the internal logic (either auto-load from the NVM or programmed by the firmware). Registers defined as RO access, override any access type defined for its fields.

Abbreviation	Description
RO	**Read Only** - A register bit field with this attribute can be read. Writes have no effect on the bit field value.
RSV	**Reserved** - A register bit field with this attribute can be read and returns an indeterministic value. Writes must set the bit field to its initial value unless specified differently in the field description.
RW	**Read/Write** - A register with this attribute can be read and written. Read return the default value, the last value written, or updated status from a previous operation. The field description specifies the field's actual behavior.

Abbreviation	Description
RCW	**Read Clear / Write** - A register bit field with this attribute can be written or read. The value returned on the read might be different than the value written (typically a counter) and the value is cleared after the read.
RW1C	**Read/Write 1 to Clear** - A register bit field with this attribute can be read and written. Writing an individual bit within the field to a 1b clears (sets to 0b) the corresponding bit and a write of a 0b has no effect. The value read might return the last value written or the status of a previous operation. The field description specifies the field's actual read behavior.
RW1S	**Read/Write 1 to Set** - A register bit field with this attribute can be read and written. Writing an individual bit within the field to a 1b sets (sets to 1b) the corresponding bit and a write of a 0b has no effect. The value read might return the last value written or the status of a previous operation. The field description specifies the field's actual read behavior.

38.39 Device Registers - PF

38.39.1 BAR0 Registers Summary

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
PF - General Registers			
0x00074400 + 0x4*VF, VF=0...127	VFGEN_RSTAT[VF]	VF Reset Status	3458
0x00083048	GL_FWSTS	Firmware Status Register	3458
0x00088000	PFGEN_STATE	PF State	3459
0x00088100 + 0x4*n, n=0...29	GLGEN_GPIO_CTL[n]	Global GPIO Control	3459
0x00088178	GLGEN_LED_CTL	Global LED Control	3461
0x0008817C	GLGEN_GPIO_STAT	Global GPIO Status	3461
0x00088180	GLGEN_GPIO_TRANSIT	Global GPIO Transition Status	3462
0x00088184	GLGEN_GPIO_SET	Global GPIO Set	3462
0x0008818C + 0x4*n, n=0...3	GLGEN_MSCA[n]	MDI Single Command and Address	3462
0x0008819C + 0x4*n, n=0...3	GLGEN_MSRWD[n]	MDI Single Read and Write Data	3463
0x000881AC + 0x4*n, n=0...3	GLGEN_I2CPARAMS[n]	I^2C Parameters	3463
0x000881BC	GLVFGEN_TIMER	Global Device Timer	3464
0x000881C0 + 0x4*n, n=0...3	GLGEN_MDIO_I2C_SEL[n]	Global MDIO or I^2C Select	3464
0x000881D0 + 0x4*n, n=0...3	GLGEN_MDIO_CTRL[n]	MDIO Control	3465
0x000881E0 + 0x4*n, n=0...3	GLGEN_I2CCMD[n]	I^2C Command	3466
0x00090000 + 0x4*VSI, VSI=0...383	VSIGEN_RTRIG[VSI]	VM Reset Trigger	3466
0x00090800 + 0x4*VSI, VSI=0...383	VSIGEN_RSTAT[VSI]	VM Reset Status	3467
0x00091800 + 0x4*VF, VF=0...127	VPGEN_VFRTRIG[VF]	VF Reset Trigger	3467
0x00091C00 + 0x4*VF, VF=0...127	VPGEN_VFRSTAT[VF]	VF Reset Status	3467
0x00092400	PFGEN_CTRL	PF Control	3467
0x00092500	PFGEN_DRUN	PF Driver Unload	3467
0x00092600 + 0x4*n, n=0...3	GLGEN_VFLRSTAT[n]	Global VF Level Reset Status	3468
0x000B612C	GLGEN_STAT	Global Status	3468
0x000B8100	PRTGEN_STATUS	General Port Status	3468
0x000B8120	PRTGEN_CNF	General Port Configuration	3469
0x000B8160	PRTGEN_CNF2	General Port Configuration2	3469
0x000B8180	GLGEN_RSTCTL	Global Reset Delay	3469

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x000B8184	GLGEN_CLKSTAT	Global Clock Status	3470
0x000B8188	GLGEN_RSTAT	Global Reset Status	3471
0x000B8190	GLGEN_RTRIG	Global Reset Trigger	3471
0x001C0480	PFGEN_PORTNUM	LAN Port Number	3471
0x001C0AB4	GLGEN_PCIFCNCNT	PCI Function Count	3472
PF - PCIe Registers			
0x0009C000	PF_FUNC_RID	Function Requester ID Information Register	3472
0x0009C080	PF_PCI_CIAA	PCIe Configuration Indirect Access Address	3472
0x0009C100	PF_PCI_CIAD	PCIe Configuration Indirect Access Data	3472
0x0009C180	PFPCI_FACTPS	Function Active and Power State	3473
0x0009C300	PFPCI_VMINDEX	PCIe VM Pending Index	3473
0x0009C380	PFPCI_VMPEND	PCIe VM Pending Status	3473
0x0009C480	GLPCI_DREVID	PCIe Default Revision ID	3473
0x0009C48C	GLPCI_GSCL_1	PCIe Statistic Control Register #1	3474
0x0009C490	GLPCI_GSCL_2	PCIe Statistic Control Register #2	3474
0x0009C494 + 0x4*n, n=0...3	GLPCI_GSCL_5_8[n]	PCIe Statistic Control Registers #5...#8	3475
0x0009C4A4 + 0x4*n, n=0...3	GLPCI_GSCN_0_3[n]	PCIe Statistic Counter Registers #0...#3	3475
0x0009C4EC	GLPCI_PQ_MAX_USED_SPC	PCIe PQs Max Used Space	3475
0x0009C4F0	GLPCI_PM_MUX_PFB	PCIe Mux Selector for PFB	3475
0x0009C4F4	GLPCI_PM_MUX_NPQ	PCIe Mux Selector for NPQs	3476
0x0009C4F8	GLPCI_SPARE_BITS_0	PCIe Regs Spare Bits 0	3476
0x0009C4FC	GLPCI_SPARE_BITS_1	PCIe Regs Spare Bits 1	3476
0x0009C500	GLPCI_CUR_RLAN_ALWD	PCIe NPQ Current Status of Allowed Resources for RLAN	3476
0x0009C504	GLPCI_CUR_TLAN_ALWD	PCIe NPQ Current Status of Allowed Resources for TLAN	3476
0x0009C508	GLPCI_CUR_RXPE_ALWD	PCIe NPQ Current Status of Allowed Resources for RXPE	3476
0x0009CA8C	GLPCI_GSCL_0	PCIe* Statistic Control Register #0	3477
0x0009C50C	GLPCI_CUR_TXPE_ALWD	PCIe NPQ Current Status of Allowed Resources for TXPE	3477
0x0009C510	GLPCI_CUR_PMAT_ALWD	PCIe NPQ Current Status of Allowed Resources for PMAT	3477
0x0009C514	GLPCI_CUR_MNG_ALWD	PCIe NPQ Current Status of Allowed Resources for MNG	3477
0x0009C518	GLPCI_CUR_TDPU_ALWD	PCIe NPQ Current Status of Allowed Resources for TDPU	3478
0x0009C580	GLPCI_CUR_RLAN_RSVD	PCIe NPQ Current Status of Reserved Resources for RLAN	3478
0x0009C584	GLPCI_CUR_TLAN_RSVD	PCIe NPQ Current Status of Reserved Resources for TLAN	3478
0x0009C588	GLPCI_CUR_RXPE_RSVD	PCIe NPQ Current Status of Reserved Resources for RXPE	3478

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x0009C58C	GLPCI_CUR_TXPE_RSVD	PCIe NPQ Current Status of Reserved Resources for TX PE	3479
0x0009C590	GLPCI_CUR_PMAT_RSVD	PCIe NPQ Current Status of Reserved Resources for PMAT	3479
0x0009C594	GLPCI_CUR_MNG_RSVD	PCIe NPQ Current Status of Reserved Resources for MNG	3479
0x0009C598	GLPCI_CUR_TDPU_RSVD	PCIe NPQ Current Status of Reserved Resources for TDPU	3479
0x0009C600 + 0x4*VF, VF=0...127	PFPCI_VF_FLUSH_DONE[VF]	PCIe VF Flush Done	3479
0x0009C800	PFPCI_PF_FLUSH_DONE	PCIe PF Flush Done	3480
0x0009C880	PFPCI_VM_FLUSH_DONE	PCIe VM Flush Done	3480
0x0009CA00	GLPCI_NPQ_CFG	PCIe NPQ Config	3480
0x0009CA18	GLPCI_CUR_CLNT_COMMON	PCIe NPQ Current Status of Common Bucket	3480
0x0009CA28	GLPCI_CUR_WATMK_CLNT_COMMON	PCIe NPQ Watermark of Common Bucket	3481
0x0009CA30	GLPCI_WATMK_CLNT_PIPEMON	PCIe NPQ Watermark of Pipe Monitor	3481
0x0009CA8C	GLPCI_GSCL_0	PCIe Statistic Control Register #0	3480
0x0009CB00	GLPCI_WATMK_RLAN_ALWD	PCIe NPQ Watermark of Allowed Resources for RLAN	3481
0x0009CB04	GLPCI_WATMK_TLAN_ALWD	PCIe NPQ Watermark of Allowed Resources for TLAN	3481
0x0009CB08	GLPCI_WATMK_RXPE_ALWD	PCIe NPQ Watermark of Allowed Resources for RXPE	3482
0x0009CB0C	GLPCI_WATMK_TXPE_ALWD	PCIe NPQ Watermark of Allowed Resources for TXPE	3482
0x0009CB10	GLPCI_WATMK_PMAT_ALWD	PCIe NPQ Watermark of Allowed Resources for PMAT	3482
0x0009CB14	GLPCI_WATMK_MNG_ALWD	PCIe NPQ Watermark of Allowed Resources for MNG	3482
0x0009CB18	GLPCI_WATMK_TPDU_ALWD	PCIe NPQ Watermark of Allowed Resources for TDPU	3483
0x000BE000	PFPCI_CNF	PCIe PF Configuration	3483
0x000BE080	PFPCI_DEVID	PCIe PF Device ID	3483
0x000BE180	PFPCI_FUNC2	PCIe Functions Configuration 2	3483
0x000BE200	PFPCI_FUNC	PCIe Functions Configuration	3484
0x000BE280	PFPCI_STATUS1	PCIe Function Status 1	3484
0x000BE300	PFPCI_PM	PCIe PM	3484
0x000BE400	PFPCI_CLASS	PCIe Storage Class	3485
0x000BE484	GLPCI_LBARCTRL	PCI BAR Control	3485
0x000BE48C	GLPCI_SUBVENID	PCIe Subsystem ID	3486
0x000BE490	GLPCI_PWRDATA	PCIe Power Data Register	3486
0x000BE494	GLPCI_CNF2	PCIe Global Config 2	3486
0x000BE498	GLPCI_SERL	PCIe Serial Number MAC Address Low	3487
0x000BE49C	GLPCI_SERH	PCIe Serial Number MAC Address High	3487
0x000BE4A4	GLPCI_CAPCTRL	PCIe Capabilities Control	3487

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x000BE4A8	GLPCI_CAPSUP	PCIe Capabilities Support	3487
0x000BE4AC	GLPCI_LINKCAP	PCIe Link Capabilities	3488
0x000BE4B0	GLPCI_PMSUP	PCIe PM Support	3489
0x000BE4B4	GLPCI_REVID	PCIe Revision ID	3489
0x000BE4B8	GLPCI_VFSUP	PCIe VF Capabilities Support	3489
0x000BE4C0	GLPCI_CNF	PCIe Global Config	3489
0x000BE4F8	GLPCI_UPADD	PCIe Upper Address	3490
0x000BE518	GLPCI_VENDORID	PCIe Vendor ID	3490
PF - MAC Registers			
0x001E2120	PRTGL_SAL	Port MAC Address Low	3490
0x001E2140	PRTGL_SAH	Port MAC Address High	3490
0x001E2440	PRTMAC_LINK_DOWN_COUNTER	Link Down Counter	3491
PF - Power Management Registers			
0x000B8140	PRTPM_GC	General Control	3491
0x001E4320	Reserved	Reserved	3491
0x001E4360	Reserved	Reserved	
0x001E4380	Reserved	Reserved	
0x001E43A0	Reserved	Reserved	3491
0x001E43C0	Reserved	Reserved	
0x001E43E0	Reserved	Reserved	
0x001E4400	Reserved	Reserved	
PF - Wake Up and Proxying Registers			
0x0006B200	PFPM_WUC	Wake Up Control Register	3491
0x0006B400	PFPM_WUFC	Wake Up Filter Control Register	3492
0x0006B600	PFPM_WUS	Wake Up Status Register	3492
0x0006C800	GLPM_WUMC	WU on MNG Control	3493
0x000B8080	PFPM_APM	APM Control Register	3493
0x001E4440 + 0x20*n, n=0...3	PRTPM_SAL[n]	Station Address Low	3493
0x001E44C0 + 0x20*n, n=0...3	PRTPM_SAH[n]	Station Address High	3493
PF - NVM Registers			
0x000B6008	GLNVM_ULD	Unit Load Status	3494
0x000B6010 + 0x4*n, n=0...59	GLNVM_PROTCSR[n]	Protected CSR List	3494
0x000B6100	GLNVM_GENS	Global NVM General Status Register	3495
0x000B6104	GLNVM_FLASHID	Flash ID Register	3495
0x000B6108	GLNVM_FLA	Flash Access Register	3495
0x000B6140	GLNVM_ALTIMERS	Auto-Load Timers	3496
0x000B6154	GLNVM_ULT	Unit Load Timeout	3496
0x000B615C	MEM_INIT_DONE_STAT	MEM_INIT_DONE_STAT	3496
0x000B6164	GLNVM_AL_REQ	GLNVM_AL_REQ	3497
PF - Analyzer Registers			

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x001C0B20	PRT_L2TAGSEN	L2 Tag - Enable	3497
PF - Switch Registers			
0x0026CFB8 + 0x4*n, n=0...1	GL_SWR_DEF_ACT_EN[n]	Switching Table Default Action Enable Bitmap	3498
0x00270200 + 0x4*n, n=0...35	GL_SWR_DEF_ACT[n]	Switching Table Default Action	3498
PF - Interrupt Registers			
0x00020000 + 0x800*n + 0x4*INTVF, n=0...2, INTVF=0...511	VFINT_ITRN[n,INTVF]	VF Interrupt Throttling for Interrupt N	3498
0x00024800 + 0x4*INTVF, INTVF=0...511	VFINT_DYN_CTLN[INTVF]	VF Interrupt N Dynamic Control	3499
0x00025000 + 0x4*INTVF, INTVF=0...511	VPINT_LNKLSTN[INTVF]	Protected VF Interrupt N Linked List	3500
0x00025800 + 0x4*INTVF, INTVF=0...511	VPINT_RATEN[INTVF]	Protected VF Interrupt N Rate Limit	3500
0x00026800 + 0x4*INTVF, INTVF=0...511	VPINT_CEQCTL[INTVF]	VF PE Completion Event Queue Interrupt Cause Control	3500
0x00028000 + 0x400*n + 0x4*VF, n=0...2, VF=0...127	VFINT_ITR0[n,VF]	VF Interrupt Throttling for Interrupt Zero	3501
0x0002A000 + 0x4*VF, VF=0...127	VFINT_STAT_CTL0[VF]	VF Interrupt Zero Static Control	3501
0x0002A400 + 0x4*VF, VF=0...127	VFINT_DYN_CTL0[VF]	VF Interrupt Zero Dynamic Control	3502
0x0002A800 + 0x4*VF, VF=0...127	VPINT_LNKLST0[VF]	Protected VF Interrupt Zero Linked List	3503
0x0002AC00 + 0x4*VF, VF=0...127	VPINT_RATE0[VF]	Protected VF Interrupt Zero Rate Limit	3503
0x0002B800 + 0x4*VF, VF=0...127	VPINT_AEQCTL[VF]	VF PE Asynchronous Event Queue Interrupt Cause Control	3503
0x0002BC00 + 0x4*VF, VF=0...127	VFINT_ICR0[VF]	VF Interrupt Zero Cause	3504
0x0002C000 + 0x4*VF, VF=0...127	VFINT_ICR0_ENA[VF]	VF Interrupt Zero Cause Enablement	3504
0x00030000 + 0x800*n + 0x4*INTPF, n=0...2, INTPF=0...511	PFINT_ITRN[n,INTPF]	PF Interrupt Throttling for Interrupt N	3504
0x00034800 + 0x4*INTPF, INTPF=0...511	PFINT_DYN_CTLN[INTPF]	PF Interrupt N Dynamic Control	3505
0x00035000 + 0x4*INTPF, INTPF=0...511	PFINT_LNKLSTN[INTPF]	PF Interrupt N Linked List	3506
0x00035800 + 0x4*INTPF, INTPF=0...511	PFINT_RATEN[INTPF]	PF Interrupt N Rate Limit	3506
0x00036800 + 0x4*INTPF, INTPF=0...511	PFINT_CEQCTL[INTPF]	PF PE Completion Event Queue Interrupt Cause Control	3506
0x00038000 + 0x80*n, n=0...2	PFINT_ITR0[n]	PF Interrupt Throttling for Interrupt Zero	3507
0x00038400	PFINT_STAT_CTL0	PF Interrupt Zero Static Control	3507
0x00038480	PFINT_DYN_CTL0	PF Interrupt Zero Dynamic Control	3508
0x00038500	PFINT_LNKLST0	PF Interrupt Zero Linked List	3509
0x00038580	PFINT_RATE0	PF Interrupt Zero Rate Limit	3509
0x00038700	PFINT_AEQCTL	PF PE Asynchronous Event Queue Interrupt Cause Control	3509
0x00038780	PFINT_ICR0	PF Interrupt Zero Cause	3510
0x00038800	PFINT_ICR0_ENA	PF Interrupt Zero Cause Enablement	3511

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x0003A000 + 0x4*Q, Q=0...1535	QINT_RQCTL[Q]	Receive Queue Interrupt Cause Control	3512
0x0003C000 + 0x4*Q, Q=0...1535	QINT_TQCTL[Q]	Transmit Queue Interrupt Cause Control	3512
0x0003F100	PFGEN_PORTMDIO_NUM	LAN Port MDIO Number	3513
0x00088080	PFINT_GPIO_ENA	PF General Purpose IO Interrupt Enablement	3514
0x00088188	EMPINT_GPIO_ENA	EMP General Purpose IO Interrupt Enablement	3514
PF - Virtualization PF Registers			
0x000442F4	GL_MDCK_TDAT	Malicious Driver TX Data Checks Enable	3515
0x000E6000 + 0x4*VF, VF=0...127	VP_MDET_TX[VF]	Malicious Driver Detected on TX	3516
0x0012A000 + 0x4*VF, VF=0...127	VP_MDET_RX[VF]	Malicious Driver Detected on RX	3518
0x0012A400 + 0x4*PF, PF=0...15	PF_MDET_RX[PF]	Malicious Driver Detected on RX	3518
0x000E6400 + 0x4*PF, PF=0...15	PF_MDET_TX[PF]	Malicious Driver Detected on TX	3516
0x000E6480	GL_MDET_TX	Malicious Driver TX Event Details	3517
0x0012A50C	GL_MDCK_RX	Malicious Driver RX Checks Enabled	3516
0x0012A510	GL_MDET_RX	Malicious Driver RX Event Details	3518
0x000E648C	GL_MDCK_TCMD	Malicious Driver TX Command Checks Enable	3516
0x001C0500 + 0x4*PF, PF=0...15	PF_VT_PFALLOC[PF]	PF Resources Allocation	3518
PF - DCB Registers			
0x00083000	PRTDCB_GENC	Port DCB General Control	3519
0x00083020	PRTDCB_GENS	Port DCB General Status	3519
0x00083044	GLDCB_GENC	Global DCB General Control	3519
0x00098060	PRTDCB_TETSC_TPB	DCB Transmit ETS Control for TPB	3520
0x000A0040 + 0x20*n, n=0...7	PRTDCB_TFMSTC[n]	DCB Transmit Frame Monitoring Status per TC	3520
0x000A0180	PRTDCB_TDPMC	DCB Transmit Data Pipe Monitor Control	3520
0x000A2040 + 0x20*n, n=0...7	PRTDCB_TCMSTC[n]	DCB Transmit Command Waiting Status per TC	3520
0x000A21A0	PRTDCB_TCPMC	DCB Transmit Command Pipe Monitor Control	3521
0x000AE060	PRTDCB_TETSC_TCB	DCB Transmit ETS Control for TCB	3521
0x00122180 + 0x20*n, n=0...7	PRTDCB_RETSTCC[n]	DCB Receive ETS per TC Control	3521
0x001223A0	PRTDCB_RPPMC	DCB Receive per Port Pipe Monitor Control	3522
0x001223E0	PRTDCB_RETSC	DCB Receive ETS Control	3522
0x00122618	GLDCB_RUPTI	DCB Receive per UP PFC Timer Indication	3522
0x001C0980	PRTDCB_TC2PFC	DCB TC to PFC Mapping	3522
0x001C09A0	PRTDCB_RUP2TC	DCB Receive UP to TC Mapping for RCB	3523
0x001C0B00	PRTDCB_RUP	DCB Receive UP in PPRS	3523
0x001E2400	PRTDCB_MFLCN	MAC Flow Control Register	3523

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x001E4560	PRTDCB_TFCS	Transmit Flow Control Status	3524
0x001E4580 + 0x20*n, n=0...3	PRTDCB_FCTTVN[n]	Flow Control Transmit Timer Value n	3524
0x001E4600	PRTDCB_FCRTV	Flow Control Refresh Threshold Value	3525
0x001E4640	PRTDCB_FCCFG	Flow Control Configuration	3525
0x001E4660 + 0x20*n, n=0...7	PRTDCB_TPFCTS[n]	DCB Transmit PFC Timer Status	3525
PF - Receive Packet Buffer Registers			
0x000AC100 + 0x20*n, n=0...7	PRTRPB_DHW[n]	RPB Dedicated Pool High Watermark	3525
0x000AC220 + 0x20*n, n=0...7	PRTRPB_DLW[n]	RPB Dedicated Pool Low Watermark	3526
0x000AC320 + 0x20*n, n=0...7	PRTRPB_DPS[n]	RPB Dedicated Pool Size	3526
0x000AC480 + 0x20*n, n=0...7	PRTRPB_SHT[n]	RPB Shared Pool High Threshold	3526
0x000AC580	PRTRPB_SHW	RPB Shared Pool High Watermark	3526
0x000AC5A0 + 0x20*n, n=0...7	PRTRPB_SLT[n]	RPB Shared Pool Low Threshold	3526
0x000AC6A0	PRTRPB_SLW	RPB Shared Pool Low Watermark	3527
0x000AC7C0	PRTRPB_SPS	RPB Shared Pool Size	3527
0x000AC828	GLRPB_DPSS	RPB Dedicated Pool Size for Single Shared Buffer State	3527
0x000AC830	GLRPB_GHW	RPB Global High Watermark	3527
0x000AC834	GLRPB_GLW	RPB Global Low Watermark	3527
0x000AC844	GLRPB_PHW	RPB Packet High Watermark	3527
0x000AC848	GLRPB_PLW	RPB Packet Low Watermark	3527
PF - Host Memory Cache Registers			
0x000A60E8	GLPEOC_CACHESIZE	PEOC Cache Attributes	3528
0x000A80BC	GLPBLOC_CACHESIZE	PBLOC Cache Attributes	3528
0x000AA0DC	GLFOC_CACHESIZE	FOC Cache Attributes	3528
0x000C0000	PFHMC_SDCMD	Private Memory Space Segment Descriptor Command	3528
0x000C0100	PFHMC_SDDATALOW	Private Memory Space Segment Descriptor Data Low	3529
0x000C0200	PFHMC_SDDATAHIGH	Private Memory Space Segment Descriptor Data High	3530
0x000C0300	PFHMC_PDINV	Private Memory Space Page Descriptor Invalidate	3530
0x000C0400	PFHMC_ERRORINFO	Host Memory Cache Error Information Register	3531
0x000C0500	PFHMC_ERRORDATA	Host Memory Cache Error Data Register	3532
0x000C0800 + 0x4*n, n=0...15	GLHMC_SDPART[n]	Private Memory Segment Table Partitioning Registers	3533
0x000C0880 + 0x4*n, n=0...15	GLHMC_PFPESDPART[n]	Private Memory Segment Table Partitioning Registers	3533
0x000C2004	GLHMC_LANTXOBJSZ	Private Memory LAN TX Object Size	3533
0x000C2008	GLHMC_LANQMAX	Private Memory LAN Queue Maximum	3533
0x000C200C	GLHMC_LANRXOBJSZ	Private Memory LAN RX Object Size	3534
0x000C201C	GLHMC_PEQPOBJSZ	Private Memory PE QP Object Size	3534

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x000C2020	GLHMC_PECQOBJSZ	Private Memory PE CQ Object Size	3534
0x000C2024	GLHMC_PESRQOBJSZ	Private Memory PE S-RQ Object Size	3534
0x000C2028	GLHMC_PESRQMAX	Private Memory Protocol Engine SRQ Max	3534
0x000C202C	GLHMC_PEHTEOBJSZ	Private Memory PE Hash Table Entry Object Size	3535
0x000C2030	GLHMC_PEHTMAX	Private Memory Protocol Engine Hash Entry Max	3535
0x000C2034	GLHMC_PEARPOBJSZ	Private Memory PE ARP Table Entry Object Size	3535
0x000C2038	GLHMC_PEARPMAX	Private Memory Protocol Engine ARP Table Entry Max	3535
0x000C203C	GLHMC_PEMROBJSZ	Private Memory PE Memory Region Table Entry Object Size	3535
0x000C2040	GLHMC_PEMRMAX	Private Memory Protocol Engine Memory Registration Max	3536
0x000C2044	GLHMC_PEXFOBJSZ	Private Memory PE Transmit FIFO Object Size	3536
0x000C2048	GLHMC_PEXFMAX	Private Memory Protocol Engine Transmit FIFO Entry Max	3536
0x000C204C	GLHMC_PEXFFLMAX	Private Memory Protocol Engine Transmit FIFO Free List Max	3536
0x000C2050	GLHMC_PEQ1OBJSZ	Private Memory PE IRRQ Object Size	3536
0x000C2054	GLHMC_PEQ1MAX	Private Memory Protocol Engine Q1 Max	3537
0x000C2058	GLHMC_PEQ1FLMAX	Private Memory Protocol Engine Q1 Free List Max	3537
0x000C205C	GLHMC_FSIMCOBJSZ	Private Memory FSI Multicast Group Object Size	3537
0x000C2060	GLHMC_FSIMCMAX	Private Memory FSI Multicast Group Max	3537
0x000C2064	GLHMC_FSIAVOBJSZ	Private Memory FSI Address Vector Object Size	3537
0x000C2068	GLHMC_FSIAVMAX	Private Memory FSI Address Vector Max	3538
0x000C206C	GLHMC_PEPBLMAX	Private Memory Protocol Engine Physical Buffer List Max	3538
0x000C2080	GLHMC_PETIMEROBJSZ	Private Memory PE Timer Object Size	3538
0x000C2084	GLHMC_PETIMERMAX	Private Memory PE Timer Object Max	3538
0x000C20E4	GLHMC_PEPFFIRSTSD	Protocol Engine PF First SD	3538
0x000C20EC	GLHMC_DBQPMAX	Private Memory Protocol Engine Queue Pair Max	3539
0x000C20F0	GLHMC_DBCQMAX	Private Memory Protocol Engine Completion Queue Max	3539
0x000C4000 + 0x4*n, n=0...15	GLHMC_PEQPBASE[n]	FPM PE QP Base	3539
0x000C4100 + 0x4*n, n=0...15	GLHMC_PEQPCNT[n]	FPM PE QP Object Count	3539
0x000C4200 + 0x4*n, n=0...15	GLHMC_PECQBASE[n]	FPM PE CQ Base	3540
0x000C4300 + 0x4*n, n=0...15	GLHMC_PECQCNT[n]	FPM PE CQ Object Count	3540
0x000C4400 + 0x4*n, n=0...15	GLHMC_PESRQBASE[n]	FPM PE Shared RQ Base	3540

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x000C4500 + 0x4*n, n=0...15	GLHMC_PESRQCNT[n]	FPM PE Shared RQ Size	3540
0x000C4600 + 0x4*n, n=0...15	GLHMC_PEHTEBASE[n]	FPM PE Hash Table Entry Base	3541
0x000C4700 + 0x4*n, n=0...15	GLHMC_PEHTCNT[n]	FPM PE Hash Table Object Count	3541
0x000C4800 + 0x4*n, n=0...15	GLHMC_PEARPBASE[n]	FPM PE ARP Table Base	3541
0x000C4900 + 0x4*n, n=0...15	GLHMC_PEARPCNT[n]	FPM PE ARP Table Object Count	3541
0x000C4A00 + 0x4*n, n=0...15	GLHMC_APBVTINUSEBASE[n]	FPM PE APBVT In-Use Base	3542
0x000C4C00 + 0x4*n, n=0...15	GLHMC_PEMRBASE[n]	FPM PE MRT Base	3542
0x000C4D00 + 0x4*n, n=0...15	GLHMC_PEMRCNT[n]	FPM PE Memory Region Table Object Count	3542
0x000C4E00 + 0x4*n, n=0...15	GLHMC_PEXFBASE[n]	FPM PE Transmit FIFO Base	3542
0x000C4F00 + 0x4*n, n=0...15	GLHMC_PEXFCNT[n]	FPM PE Transmit FIFO Object Count	3543
0x000C5000 + 0x4*n, n=0...15	GLHMC_PEXFFLBASE[n]	FPM PE Transmit FIFO Free List Base	3543
0x000C5200 + 0x4*n, n=0...15	GLHMC_PEQ1BASE[n]	FPM PE IRRQ Base	3543
0x000C5300 + 0x4*n, n=0...15	GLHMC_PEQ1CNT[n]	FPM PE IRRQ Object Count	3543
0x000C5400 + 0x4*n, n=0...15	GLHMC_PEQ1FLBASE[n]	FPM PE IRRQ Free List Base	3544
0x000C5600 + 0x4*n, n=0...15	GLHMC_FSIAVBASE[n]	FPM FSI Address Vector Base	3544
0x000C5700 + 0x4*n, n=0...15	GLHMC_FSIAVCNT[n]	FPM FSI Address Vector Object Count	3544
0x000C5800 + 0x4*n, n=0...15	GLHMC_PEPBLBASE[n]	FPM PE Physical Buffer List Base	3544
0x000C5900 + 0x4*n, n=0...15	GLHMC_PEPBLCNT[n]	FPM PE PBL Object Count	3545
0x000C5A00 + 0x4*n, n=0...15	GLHMC_PETIMERBASE[n]	FPM PE Timer Base	3545
0x000C5B00 + 0x4*n, n=0...15	GLHMC_PETIMERCNT[n]	FPM PE Timer Object Count	3545
0x000C6000 + 0x4*n, n=0...15	GLHMC_FSIMCBASE[n]	FPM FSI Multicast Group Base	3546
0x000C6100 + 0x4*n, n=0...15	GLHMC_FSIMCCNT[n]	FPM FSI Multicast Group Object Count	3546
0x000C6200 + 0x4*n, n=0...15	GLHMC_LANTXBASE[n]	FPM LAN TX Queue Base	3546
0x000C6300 + 0x4*n, n=0...15	GLHMC_LANTXCNT[n]	FPM LAN TX Queue Object Count	3546
0x000C6400 + 0x4*n, n=0...15	GLHMC_LANRXBASE[n]	FPM LAN RX Queue Base	3547
0x000C6500 + 0x4*n, n=0...15	GLHMC_LANRXCNT[n]	FPM LAN RX Queue Object Count	3547
0x000C8300 + 0x4*n, n=0...31	GLHMC_VFPDINV[n]	Private Memory Space Page Descriptor Invalidate	3547
0x000C8800 + 0x4*n, n=0...31	GLHMC_VFSDPART[n]	Private Memory Segment Table Partitioning Registers	3548
0x000CC000 + 0x4*n, n=0...31	GLHMC_VFPEQPBASE[n]	FPM PE QP Base	3548
0x000CC100 + 0x4*n, n=0...31	GLHMC_VFPEQPCNT[n]	FPM PE QP Object Count	3548
0x000CC200 + 0x4*n, n=0...31	GLHMC_VFPECQBASE[n]	FPM PE CQ Base	3549
0x000CC300 + 0x4*n, n=0...31	GLHMC_VFPECQCNT[n]	FPM PE CQ Object Count	3549
0x000CC400 + 0x4*n, n=0...31	GLHMC_VFPESRQBASE[n]	FPM PE Shared RQ Base	3549
0x000CC500 + 0x4*n, n=0...31	GLHMC_VFPESRQCNT[n]	FPM PE Shared RQ Size	3550
0x000CC600 + 0x4*n, n=0...31	GLHMC_VFPEHTEBASE[n]	FPM PE Hash Table Entry Base	3550
0x000CC700 + 0x4*n, n=0...31	GLHMC_VFPEHTCNT[n]	FPM PE Hash Table Object Count	3550
0x000CC800 + 0x4*n, n=0...31	GLHMC_VFPEARPBASE[n]	FPM PE ARP Table Base	3551
0x000CC900 + 0x4*n, n=0...31	GLHMC_VFPEARPCNT[n]	FPM PE ARP Table Object Count	3551
0x000CCA00 + 0x4*n, n=0...31	GLHMC_VFAPBVTINUSEBASE[n]	FPM PE APBVT In-Use Base	3551

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x000CCC00 + 0x4*n, n=0...31	GLHMC_VFPEMRBASE[n]	FPM PE MRT Base	3552
0x000CCD00 + 0x4*n, n=0...31	GLHMC_VFPEMRCNT[n]	FPM PE Memory Region Table Object Count	3552
0x000CCE00 + 0x4*n, n=0...31	GLHMC_VFPEXFBASE[n]	FPM PE Transmit FIFO Base	3552
0x000CCF00 + 0x4*n, n=0...31	GLHMC_VFPEXFCNT[n]	FPM PE Transmit FIFO Object Count	3553
0x000CD000 + 0x4*n, n=0...31	GLHMC_VFPEXFFLBASE[n]	FPM PE Transmit FIFO Free List Base	3553
0x000CD200 + 0x4*n, n=0...31	GLHMC_VFPEQ1BASE[n]	FPM PE IRRQ Base	3553
0x000CD300 + 0x4*n, n=0...31	GLHMC_VFPEQ1CNT[n]	FPM PE IRRQ Object Count	3554
0x000CD400 + 0x4*n, n=0...31	GLHMC_VFPEQ1FLBASE[n]	FPM PE IRRQ Free List Base	3554
0x000CD600 + 0x4*n, n=0...31	GLHMC_VFFSIAVBASE[n]	FPM FSI Address Vector Base	3554
0x000CD700 + 0x4*n, n=0...31	GLHMC_VFFSIAVCNT[n]	FPM FSI Address Vector Object Count	3555
0x000CD800 + 0x4*n, n=0...31	GLHMC_VFPEPBLBASE[n]	FPM PE Physical Buffer List Base	3555
0x000CD900 + 0x4*n, n=0...31	GLHMC_VFPEPBLCNT[n]	FPM PE PBL Object Count	3555
0x000CDA00 + 0x4*n, n=0...31	GLHMC_VFPETIMERBASE[n]	FPM VF PE Timer Base	3556
0x000CDB00 + 0x4*n, n=0...31	GLHMC_VFPETIMERCNT[n]	FPM VF PE Timer Object Count	3556
0x000D0088	GLPDOC_CACHESIZE	PDOC Cache Attributes	3556
0x00131240 + 0x4*n, n=0...15	GLHMC_DBCQPART[n]	Private Memory CQ Doorbell Partition Registers	3557
0x001312C0 + 0x4*n, n=0...15	GLHMC_CEQPART[n]	Private Memory CEQ Partitioning Registers	3557
0x00132140 + 0x4*n, n=0...31	GLHMC_VFDBCQPART[n]	Private Memory CQ Doorbell Partition Registers	3557
0x00132240 + 0x4*n, n=0...31	GLHMC_VFCEQPART[n]	Private Memory CEQ Partitioning Registers	3558
0x00138D80 + 0x4*n, n=0...15	GLHMC_DBQPPART[n]	Private Memory QP Doorbell Partition Registers	3558
0x00138E00 + 0x4*n, n=0...31	GLHMC_VFDBQPPART[n]	Private Memory VF QP Doorbell Partition Registers	3558
PF - Context Manager Registers			
0x0010C100 + 0x80*n, n=0...3	PFCM_LANCTXDATA[n]	CMLAN Context Data Registers	3559
0x0010C300	PFCM_LANCTXCTL	CMLAN Context Control Register	3559
0x0010C380	PFCM_LANCTXSTAT	CMLAN Context Status Register	3559
0x0010C4D8	GLCM_LAN_CACHESIZE	CMLAN Cache Attributes	3560
0x00138FE4	GLCM_PE_CACHESIZE	CMPE Cache Attributes	3560
PF - Admin Queue			
0x00080000	PF_ATQBAL	PF Admin Transmit Queue Base Address Low	3560
0x00080040	GL_ATQBAL	Global Admin Transmit Queue Base Address Low	3560
0x00080080	PF_ARQBAL	PF Admin Receive Queue Base Address Low	3560
0x000800C0	GL_ARQBAL	Global Admin Receive Queue Base Address Low	3561
0x00080100	PF_ATQBAH	PF Admin Transmit Queue Base Address High	3561

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x00080140	GL_ATQBAH	Global Admin Transmit Queue Base Address High	3561
0x00080180	PF_ARQBAH	PF Admin Receive Queue Base Address High	3561
0x000801C0	GL_ARQBAH	Global Admin Receive Queue Base Address High	3561
0x00080200	PF_ATQLEN	PF Admin Transmit Queue Length	3561
0x00080240	GL_ATQLEN	Global Admin Transmit Queue Length	3562
0x00080280	PF_ARQLEN	PF Admin Receive Queue Length	3562
0x00080300	PF_ATQH	PF Admin Transmit Head	3562
0x00080340	GL_ATQH	Global Admin Transmit Head	3563
0x00080380	PF_ARQH	PF Admin Receive Queue Head	3563
0x000803C0	GL_ARQH	Global Admin Receive Queue Head	3563
0x00080400	PF_ATQT	PF Admin Transmit Tail	3563
0x00080440	GL_ATQT	Global Admin Transmit Tail	3563
0x00080480	PF_ARQT	PF Admin Receive Queue Tail	3563
0x000804C0	GL_ARQT	Global Admin Receive Queue Tail	3564
0x00080800 + 0x4*VF, VF=0...127	VF_ATQBAL[VF]	VF Admin Transmit Queue Base Address Low	3564
0x00080C00 + 0x4*VF, VF=0...127	VF_ARQBAL[VF]	VF Admin Receive Queue Base Address Low	3564
0x00081000 + 0x4*VF, VF=0...127	VF_ATQBAH[VF]	VF Admin Transmit Queue Base Address High	3564
0x00081400 + 0x4*VF, VF=0...127	VF_ARQBAH[VF]	VF Admin Receive Queue Base Address High	3564
0x00081800 + 0x4*VF, VF=0...127	VF_ATQLEN[VF]	VF Admin Transmit Queue Length	3564
0x00081C00 + 0x4*VF, VF=0...127	VF_ARQLEN[VF]	VF Admin Receive Queue Length	3565
0x00082000 + 0x4*VF, VF=0...127	VF_ATQH[VF]	VF Admin Transmit Head	3565
0x00082400 + 0x4*VF, VF=0...127	VF_ARQH[VF]	VF Admin Receive Queue Head	3565
0x00082800 + 0x4*VF, VF=0...127	VF_ATQT[VF]	VF Admin Transmit Tail	3566
0x00082C00 + 0x4*VF, VF=0...127	VF_ARQT[VF]	VF Admin Receive Queue Tail	3566
PF - Statistics Registers			
0x00300000 + 0x8*n, n=0...3	GLPRT_GORCL[n]	Port Good Octets Received Count Low	3566
0x00300004 + 0x8*n, n=0...3	GLPRT_GORCH[n]	Port Good Octets Received Count High	3566
0x00300020 + 0x8*n, n=0...3	GLPRT_MLFC[n]	Port MAC Local Fault Count	3566
0x00300040 + 0x8*n, n=0...3	GLPRT_MRFC[n]	Port MAC Remote Fault Count	3567
0x00300080 + 0x8*n, n=0...3	GLPRT_CRCERRS[n]	Port CRC Error Count	3567
0x003000A0 + 0x8*n, n=0...3	GLPRT_RLEC[n]	Receive Length Error Count	3567
0x003000E0 + 0x8*n, n=0...3	GLPRT_ILLERRC[n]	Port Illegal Byte Error Count	3567
0x00300100 + 0x8*n, n=0...3	GLPRT_RUC[n]	Receive Undersize Count	3567
0x00300120 + 0x8*n, n=0...3	GLPRT_ROC[n]	Receive Oversize Count	3567
0x00300140 + 0x8*n, n=0...3	GLPRT_LXONRXC[n]	Port Link XON Received Count	3568
0x00300160 + 0x8*n, n=0...3	GLPRT_LXOFFRXC[n]	Port Link XOFF Received Count	3568

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x00300180 + 0x8*n + 0x20*m, n=0...3, m=0...7	GLPRT_PXONRXC[n,m]	Priority XON Received Count	3568
0x00300280 + 0x8*n + 0x20*m, n=0...3, m=0...7	GLPRT_PXOFFRXC[n,m]	Priority XOFF Received Count	3568
0x00300380 + 0x8*n + 0x20*m, n=0...3, m=0...7	GLPRT_RXON2OFFCNT[n,m]	Priority XON to XOFF Count	3568
0x00300480 + 0x8*n, n=0...3	GLPRT_PRC64L[n]	Packets Received [64 Bytes] Count Low	3568
0x00300484 + 0x8*n, n=0...3	GLPRT_PRC64H[n]	Packets Received [64 Bytes] Count High	3569
0x003004A0 + 0x8*n, n=0...3	GLPRT_PRC127L[n]	Packets Received [65-127 Bytes] Count Low	3569
0x003004A4 + 0x8*n, n=0...3	GLPRT_PRC127H[n]	Packets Received [65-127 Bytes] Count High	3569
0x003004C0 + 0x8*n, n=0...3	GLPRT_PRC255L[n]	Packets Received [128-255 Bytes] Count Low	3569
0x003004C4 + 0x8*n, n=0...3	GLPRT_PRC255H[n]	Packets Received [128-255 Bytes] Count High	3570
0x003004E0 + 0x8*n, n=0...3	GLPRT_PRC511L[n]	Packets Received [256-511 Bytes] Count Low	3570
0x003004E4 + 0x8*n, n=0...3	GLPRT_PRC511H[n]	Packets Received [256-511 Bytes] Count High	3570
0x00300500 + 0x8*n, n=0...3	GLPRT_PRC1023L[n]	Packets Received [512-1023 Bytes] Count Low	3570
0x00300504 + 0x8*n, n=0...3	GLPRT_PRC1023H[n]	Packets Received [512-1023 Bytes] Count High	3571
0x00300520 + 0x8*n, n=0...3	GLPRT_PRC1522L[n]	Packets Received [1024-1522] Count Low	3571
0x00300524 + 0x8*n, n=0...3	GLPRT_PRC1522H[n]	Packets Received [1024-1522] Count High	3571
0x00300540 + 0x8*n, n=0...3	GLPRT_PRC9522L[n]	Packets Received [1523-9522 Bytes] Count Low	3571
0x00300544 + 0x8*n, n=0...3	GLPRT_PRC9522H[n]	Packets Received [1523-9522 Bytes] Count High	3572
0x00300560 + 0x8*n, n=0...3	GLPRT_RFC[n]	Receive Fragment Count	3572
0x00300580 + 0x8*n, n=0...3	GLPRT_RJC[n]	Receive Jabber Count	3572
0x003005A0 + 0x8*n, n=0...3	GLPRT_UPRCL[n]	Port Unicast Packets Received Count Low	3572
0x003005A4 + 0x8*n, n=0...3	GLPRT_UPRCH[n]	Port Unicast Packets Received Count High	3572
0x003005C0 + 0x8*n, n=0...3	GLPRT_MPRCL[n]	Port Multicast Packets Received Count Low	3573
0x003005C4 + 0x8*n, n=0...3	GLPRT_MPRCH[n]	Port Multicast Packets Received Count High	3573
0x003005E0 + 0x8*n, n=0...3	GLPRT_BPRCL[n]	Port Broadcast Packets Received Count Low	3573
0x003005E4 + 0x8*n, n=0...3	GLPRT_BPRCH[n]	Port Broadcast Packets Received Count High	3573
0x00300600 + 0x8*n, n=0...3	GLPRT_RDPC[n]	Port Receive Packets Discarded Count	3573
0x00300620 + 0x8*n, n=0...3	GLPRT_LDPC[n]	VM-VM Loopback Packets Discarded Count	3573

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x00300660 + 0x8*n, n=0...3	GLPRT_RUPP[n]	Port Received with No Destination	3574
0x00300680 + 0x8*n, n=0...3	GLPRT_GOTCL[n]	Port Good Octets Transmit Count Low	3574
0x00300684 + 0x8*n, n=0...3	GLPRT_GOTCH[n]	Port Good Octets Transmit Count High	3574
0x003006A0 + 0x8*n, n=0...3	GLPRT_PTC64L[n]	Packets Transmitted [64 Bytes] Count Low	3574
0x003006A4 + 0x8*n, n=0...3	GLPRT_PTC64H[n]	Packets Transmitted [64 Bytes] Count High	3574
0x003006C0 + 0x8*n, n=0...3	GLPRT_PTC127L[n]	Packets Transmitted [65-127 Bytes] Count Low	3575
0x003006C4 + 0x8*n, n=0...3	GLPRT_PTC127H[n]	Packets Transmitted [65-127 Bytes] Count High	3575
0x003006E0 + 0x8*n, n=0...3	GLPRT_PTC255L[n]	Packets Transmitted [128-255 Bytes] Count Low	3575
0x003006E4 + 0x8*n, n=0...3	GLPRT_PTC255H[n]	Packets Transmitted [128-255 Bytes] Count High	3575
0x00300700 + 0x8*n, n=0...3	GLPRT_PTC511L[n]	Packets Transmitted [256-511 Bytes] Count Low	3576
0x00300704 + 0x8*n, n=0...3	GLPRT_PTC511H[n]	Packets Transmitted [256-511 Bytes] Count High	3576
0x00300720 + 0x8*n, n=0...3	GLPRT_PTC1023L[n]	Packets Transmitted [512-1023 Bytes] Count Low	3576
0x00300724 + 0x8*n, n=0...3	GLPRT_PTC1023H[n]	Packets Transmitted [512-1023 Bytes] Count High	3576
0x00300740 + 0x8*n, n=0...3	GLPRT_PTC1522L[n]	Packets Transmitted [1024-1522 Bytes] Count Low	3577
0x00300744 + 0x8*n, n=0...3	GLPRT_PTC1522H[n]	Packets Transmitted [1024-1522 Bytes] Count High	3577
0x00300760 + 0x8*n, n=0...3	GLPRT_PTC9522L[n]	Packets Transmitted [1523-9522 Bytes] Count Low	3577
0x00300764 + 0x8*n, n=0...3	GLPRT_PTC9522H[n]	Packets Transmitted [1523-9522 Bytes] Count High	3577
0x00300780 + 0x8*n + 0x20*m, n=0...3, m=0...7	GLPRT_PXONTXC[n,m]	Priority XON Transmitted Count	3578
0x00300880 + 0x8*n + 0x20*m, n=0...3, m=0...7	GLPRT_PXOFFTXC[n,m]	Priority XOFF Transmitted Count	3578
0x00300980 + 0x8*n, n=0...3	GLPRT_LXONTXC[n]	Port Link XON Transmitted Count	3578
0x003009A0 + 0x8*n, n=0...3	GLPRT_LXOFFTXC[n]	Port Link XOFF Transmitted Count	3578
0x003009C0 + 0x8*n, n=0...3	GLPRT_UPTCL[n]	Port Unicast Packets Transmit Count Low	3578
0x003009C4 + 0x8*n, n=0...3	GLPRT_UPTCH[n]	Port Unicast Packets Transmit Count High	3578
0x003009E0 + 0x8*n, n=0...3	GLPRT_MPTCL[n]	Port Multicast Packets Transmit Count Low	3579
0x003009E4 + 0x8*n, n=0...3	GLPRT_MPTCH[n]	Port Multicast Packets Transmit Count High	3579
0x00300A00 + 0x8*n, n=0...3	GLPRT_BPTCL[n]	Port Broadcast Packets Transmit Count Low	3579
0x00300A04 + 0x8*n, n=0...3	GLPRT_BPTCH[n]	Port Broadcast Packets Transmit Count High	3579
0x00300A20 + 0x8*n, n=0...3	GLPRT_TDOLD[n]	Transmit Discard on Link Down	3579

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x00310000 + 0x8*n, n=0...383	GLV_RDPC[n]	VSI Received Discard Packet Count	3580
0x00318000 + 0x8*n, n=0...143	GL_RXERR1_L[n]	Receive Error Counter 1 Low	3580
0x00328000 + 0x8*n, n=0...383	GLV_GOTCL[n]	VSI Good Octets Transmit Count Low	3580
0x00328004 + 0x8*n, n=0...383	GLV_GOTCH[n]	VSI Good Octets Transmit Count High	3580
0x0032C000 + 0x8*n, n=0...15	GLSW_GOTCL[n]	Switch Good Octets Transmit Count Low	3580
0x0032C004 + 0x8*n, n=0...15	GLSW_GOTCH[n]	Switch Good Octets Transmit Count High	3580
0x00330000 + 0x8*n, n=0...127	GLVEBVL_GOTCL[n]	VEB VLAN Transmit Byte Count Low	3581
0x00330004 + 0x8*n, n=0...127	GLVEBVL_GOTCH[n]	VEB VLAN Transmit Byte Count High	3581
0x00334000 + 0x8*n + 0x40*m, n=0...7, m=0...15	GLVEBTC_TBCL[n,m]	VEB TC Transmit byte count Low	3581
0x00334004 + 0x8*n + 0x40*m, n=0...7, m=0...15	GLVEBTC_TBCH[n,m]	VEB TC Transmit byte count High	3581
0x00338000 + 0x8*n + 0x40*m, n=0...7, m=0...15	GLVEBTC_TPCL[n,m]	VEB TC Transmit Packet Count Low	3581
0x00338004 + 0x8*n + 0x40*m, n=0...7, m=0...15	GLVEBTC_TPCH[n,m]	VEB TC Transmit Packet Count High	3582
0x0033C000 + 0x8*n, n=0...383	GLV_UPTCL[n]	VSI Unicast Packets Transmit Count Low	3582
0x0033C004 + 0x8*n, n=0...383	GLV_UPTCH[n]	VSI Unicast Packets Transmit Count High	3582
0x0033CC00 + 0x8*n, n=0...383	GLV_MPTCL[n]	VSI Multicast Packets Transmit Count Low	3582
0x0033CC04 + 0x8*n, n=0...383	GLV_MPTCH[n]	VSI Multicast Packets Transmit Count High	3582
0x0033D800 + 0x8*n, n=0...383	GLV_BPTCL[n]	VSI Broadcast Packets Transmit Count Low	3583
0x0033D804 + 0x8*n, n=0...383	GLV_BPTCH[n]	VSI Broadcast Packets Transmit Count High	3583
0x00340000 + 0x8*n, n=0...15	GLSW_UPTCL[n]	Switch Unicast Packets Transmit Count Low	3583
0x00340004 + 0x8*n, n=0...15	GLSW_UPTCH[n]	Switch Unicast Packets Transmit Count High	3583
0x00340080 + 0x8*n, n=0...15	GLSW_MPTCL[n]	Switch Multicast Packets Transmit Count Low	3583
0x00340084 + 0x8*n, n=0...15	GLSW_MPTCH[n]	Switch Multicast Packets Transmit Count High	3584
0x00340100 + 0x8*n, n=0...15	GLSW_BPTCL[n]	Switch Broadcast Packets Transmit Count Low	3584
0x00340104 + 0x8*n, n=0...15	GLSW_BPTCH[n]	Switch Broadcast Packets Transmit Count High	3584
0x00344000 + 0x4*VSI, VSI=0...383	GLV_TEPC[VSI]	VSI Transmit Error Packet Count	3584
0x00348000 + 0x8*n, n=0...15	GLSW_TDPC[n]	Switch Transmit packets Discarded Count	3584
0x00358000 + 0x8*n, n=0...383	GLV_GORCL[n]	VSI Good Octets Received Count Low	3584
0x00358004 + 0x8*n, n=0...383	GLV_GORCH[n]	VSI Good Octets Received Count High	3585
0X0035C000 + 0x8*n, n=0...15	GLSW_GORCL[n]	Switch Good Octets Received Count Low	3585

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x0035C004 + 0x8*n, n=0...15	GLSW_GORCH[n]	Switch Good Octets Received Count High	3585
0x00360000 + 0x8*n, n=0...127	GLVEBVL_GORCL[n]	VEB VLAN Receive Byte Count Low	3585
0x00360004 + 0x8*n, n=0...127	GLVEBVL_GORCH[n]	VEB VLAN Receive Byte Count High	3585
0x00364000 + 0x8*n + 0x40*m, n=0...7, m=0...15	GLVEBTC_RBCL[n,m]	VEB TC Receive Byte Count Low	3586
0x00364004 + 0x8*n + 0x40*m, n=0...7, m=0...15	GLVEBTC_RBCH[n,m]	VEB TC Receive Byte Count High	3586
0x00368000 + 0x8*n + 0x40*m, n=0...7, m=0...15	GLVEBTC_RPCL[n,m]	VEB TC Receive Packet Count Low	3586
0x00368004 + 0x8*n + 0x40*m, n=0...7, m=0...15	GLVEBTC_RPCH[n,m]	VEB TC Receive Packet Count High	3586
0x0036C000 + 0x8*n, n=0...383	GLV_UPRCL[n]	VSI Unicast Packets Received Count Low	3586
0x0036C004 + 0x8*n, n=0...383	GLV_UPRCH[n]	VSI Unicast Packets Received Count High	3587
0x0036CC00 + 0x8*n, n=0...383	GLV_MPRCL[n]	VSI Multicast Packets Received Count Low	3587
0x0036CC04 + 0x8*n, n=0...383	GLV_MPRCH[n]	VSI Multicast Packets Received Count High	3587
0x0036D800 + 0x8*n, n=0...383	GLV_BPRCL[n]	VSI Broadcast Packets Received Count Low	3587
0x0036D804 + 0x8*n, n=0...383	GLV_BPRCH[n]	VSI Broadcast Packets Received Count High	3587
0x0036E400 + 0x8*n, n=0...383	GLV_RUPP[n]	VSI Received Unknown Packet Protocol Count	3588
0x00370000 + 0x8*n, n=0...15	GLSW_UPRCL[n]	Switch Unicast Packets Received Count Low	3588
0x00370004 + 0x8*n, n=0...15	GLSW_UPRCH[n]	Switch Unicast Packets Received Count High	3588
0x00370080 + 0x8*n, n=0...15	GLSW_MPRCL[n]	Switch Multicast Packets Received Count Low	3588
0x00370084 + 0x8*n, n=0...15	GLSW_MPRCH[n]	Switch Multicast Packets Received Count High	3588
0x00370100 + 0x8*n, n=0...15	GLSW_BPRCL[n]	Switch Broadcast Packets Received Count Low	3589
0x00370104 + 0x8*n, n=0...15	GLSW_BPRCH[n]	Switch Broadcast Packets Received Count High	3589
0x00370180 + 0x8*n, n=0...15	GLSW_RUPP[n]	Switch Received Unknown Packet Protocol Count	3589
0x00374000 + 0x8*n, n=0...127	GLVEBVL_UPCL[n]	VEB VLAN Unicast Packet Count Low	3589
0x00374004 + 0x8*n, n=0...127	GLVEBVL_UPCH[n]	VEB VLAN Unicast Packet Count High	3589
0x00374400 + 0x8*n, n=0...127	GLVEBVL_MPCL[n]	VEB VLAN Multicast Packet Count Low	3590
0x00374404 + 0x8*n, n=0...127	GLVEBVL_MPCH[n]	VEB VLAN Multicast Packet Count High	3590
0x00374800 + 0x8*n, n=0...127	GLVEBVL_BPCL[n]	VEB VLAN Broadcast Packet Count Low	3590
0x00374804 + 0x8*n, n=0...127	GLVEBVL_BPCH[n]	VEB VLAN Broadcast Packet Count High	3590
PF - Protocol Engine Statistics Registers			

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x00010000 + 0x4*n, n=0...15	GLPES_PFRXVLANERR[n]	Protocol Engine Statistics Received VLAN_ID Errors	3590
0x00010200 + 0x8*n, n=0...15	GLPES_PFIP4RXOCTSLO[n]	Protocol Engine Statistics IPv4 Received Octets Low	3591
0x00010204 + 0x8*n, n=0...15	GLPES_PFIP4RXOCTSHI[n]	Protocol Engine Statistics IPv4 Received Octets High	3591
0x00010400 + 0x8*n, n=0...15	GLPES_PFIP4RXPKTSLO[n]	Protocol Engine Statistics IPv4 Received Packets Low	3591
0x00010404 + 0x8*n, n=0...15	GLPES_PFIP4RXPKTSHI[n]	Protocol Engine Statistics IPv4 Received Packets High	3591
0x00010600 + 0x4*n, n=0...15	GLPES_PFIP4RXDISCARD[n]	Protocol Engine Statistics IPv4 Discards	3592
0x00010700 + 0x4*n, n=0...15	GLPES_PFIP4RXTRUNC[n]	Protocol Engine Statistics IPv4 Truncated Packets	3592
0x00010800 + 0x8*n, n=0...15	GLPES_PFIP4RXFRAGSLO[n]	Protocol Engine Statistics IPv4 Received Fragments Low	3592
0x00010804 + 0x8*n, n=0...15	GLPES_PFIP4RXFRAGSHI[n]	Protocol Engine Statistics IPv4 Received Fragments High	3592
0x00010A00 + 0x8*n, n=0...15	GLPES_PFIP4RXMCOCTSLO[n]	Protocol Engine Statistics IPv4 Received Multicast Octets Low	3592
0x00010A04 + 0x8*n, n=0...15	GLPES_PFIP4RXMCOCTSHI[n]	Protocol Engine Statistics IPv4 Received Multicast Octets High	3593
0x00010C00 + 0x8*n, n=0...15	GLPES_PFIP4RXMCPKTSLO[n]	Protocol Engine Statistics IPv4 Received Multicast Packets Low	3593
0x00010C04 + 0x8*n, n=0...15	GLPES_PFIP4RXMCPKTSHI[n]	Protocol Engine Statistics IPv4 Received Multicast Packets High	3593
0x00010E00 + 0x8*n, n=0...15	GLPES_PFIP6RXOCTSLO[n]	Protocol Engine Statistics IPv6 Received Octets Low	3593
0x00010E04 + 0x8*n, n=0...15	GLPES_PFIP6RXOCTSHI[n]	Protocol Engine Statistics IPv6 Received Octets High	3594
0x00011000 + 0x8*n, n=0...15	GLPES_PFIP6RXPKTSLO[n]	Protocol Engine Statistics IPv6 Received Packets Low	3594
0x00011004 + 0x8*n, n=0...15	GLPES_PFIP6RXPKTSHI[n]	Protocol Engine Statistics IPv6 Received Packets High	3594
0x00011200 + 0x4*n, n=0...15	GLPES_PFIP6RXDISCARD[n]	Protocol Engine Statistics IPv6 Discards	3594
0x00011300 + 0x4*n, n=0...15	GLPES_PFIP6RXTRUNC[n]	Protocol Engine Statistics IPv6 Truncated Packets	3595
0x00011400 + 0x8*n, n=0...15	GLPES_PFIP6RXFRAGSLO[n]	Protocol Engine Statistics IPv6 Received Fragments Low	3595
0x00011404 + 0x8*n, n=0...15	GLPES_PFIP6RXFRAGSHI[n]	Protocol Engine Statistics IPv6 Received Fragments High	3595
0x00011600 + 0x8*n, n=0...15	GLPES_PFIP6RXMCOCTSLO[n]	Protocol Engine Statistics IPv6 Received Multicast Octets Low	3595
0x00011604 + 0x8*n, n=0...15	GLPES_PFIP6RXMCOCTSHI[n]	Protocol Engine Statistics IPv6 Received Multicast Octets High	3596
0x00011800 + 0x8*n, n=0...15	GLPES_PFIP6RXMCPKTSLO[n]	Protocol Engine Statistics IPv6 Received Multicast Packets Low	3596
0x00011804 + 0x8*n, n=0...15	GLPES_PFIP6RXMCPKTSHI[n]	Protocol Engine Statistics IPv6 Received Multicast Packets High	3596
0x00011A00 + 0x8*n, n=0...15	GLPES_PFIP4TXOCTSLO[n]	Protocol Engine Statistics IPv4 Transmitted Octets Low	3596

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x00011A04 + 0x8*n, n=0...15	GLPES_PFIP4TXOCTSHI[n]	Protocol Engine Statistics IPv4 Transmitted Octets High	3597
0x00011C00 + 0x8*n, n=0...15	GLPES_PFIP4TXPKTSLO[n]	Protocol Engine Statistics IPv4 Transmitted Packets Low	3597
0x00011C04 + 0x8*n, n=0...15	GLPES_PFIP4TXPKTSHI[n]	Protocol Engine Statistics IPv4 Transmitted Packets High	3597
0x00011E00 + 0x8*n, n=0...15	GLPES_PFIP4TXFRAGSLO[n]	Protocol Engine Statistics IPv4 Transmitted Fragments Low	3597
0x00011E04 + 0x8*n, n=0...15	GLPES_PFIP4TXFRAGSHI[n]	Protocol Engine Statistics IPv4 Transmitted Fragments High	3598
0x00012000 + 0x8*n, n=0...15	GLPES_PFIP4TXMCOCTSLO[n]	Protocol Engine Statistics IPv4 Transmitted Multicast Octets Low	3598
0x00012004 + 0x8*n, n=0...15	GLPES_PFIP4TXMCOCTSHI[n]	Protocol Engine Statistics IPv4 Transmitted Multicast Octets High	3598
0x00012200 + 0x8*n, n=0...15	GLPES_PFIP4TXMCPKTSLO[n]	Protocol Engine Statistics IPv4 Transmitted Multicast Packets Low	3598
0x00012204 + 0x8*n, n=0...15	GLPES_PFIP4TXMCPKTSHI[n]	Protocol Engine Statistics IPv4 Transmitted Multicast Packets High	3599
0x00012400 + 0x8*n, n=0...15	GLPES_PFIP6TXOCTSLO[n]	Protocol Engine Statistics IPv6 Transmitted Octets Low	3599
0x00012404 + 0x8*n, n=0...15	GLPES_PFIP6TXOCTSHI[n]	Protocol Engine Statistics IPv6 Transmitted Octets High	3599
0x00012600 + 0x8*n, n=0...15	GLPES_PFIP6TXPKTSLO[n]	Protocol Engine Statistics IPv6 Transmitted Packets Low	3599
0x00012604 + 0x8*n, n=0...15	GLPES_PFIP6TXPKTSHI[n]	Protocol Engine Statistics IPv6 Transmitted Packets High	3600
0x00012800 + 0x8*n, n=0...15	GLPES_PFIP6TXFRAGSLO[n]	Protocol Engine Statistics IPv6 Transmitted Fragments Low	3600
0x00012804 + 0x8*n, n=0...15	GLPES_PFIP6TXFRAGSHI[n]	Protocol Engine Statistics IPv6 Transmitted Fragments High	3600
0x00012A00 + 0x8*n, n=0...15	GLPES_PFIP6TXMCOCTSLO[n]	Protocol Engine Statistics IPv6 Transmitted Multicast Octets Low	3600
0x00012A04 + 0x8*n, n=0...15	GLPES_PFIP6TXMCOCTSHI[n]	Protocol Engine Statistics IPv6 Transmitted Multicast Octets High	3601
0x00012C00 + 0x8*n, n=0...15	GLPES_PFIP6TXMCPKTSLO[n]	Protocol Engine Statistics IPv6 Transmitted Multicast Packets Low	3601
0x00012C04 + 0x8*n, n=0...15	GLPES_PFIP6TXMCPKTSHI[n]	Protocol Engine Statistics IPv6 Transmitted Multicast Packets High	3601
0x00012E00 + 0x4*n, n=0...15	GLPES_PFIP4TXNOROUTE[n]	Protocol Engine Statistics IPv4 Discarded No Route Packets	3601
0x00012F00 + 0x4*n, n=0...15	GLPES_PFIP6TXNOROUTE[n]	Protocol Engine Statistics IPv6 Discarded No Route Packets	3601
0x00013000 + 0x8*n, n=0...15	GLPES_PFTCPRXSEGSLO[n]	Protocol Engine Statistics TCP Received Segments Low	3602
0x00013004 + 0x8*n, n=0...15	GLPES_PFTCPRXSEGSHI[n]	Protocol Engine Statistics TCP Received Segments High	3602
0x00013200 + 0x4*n, n=0...15	GLPES_PFTCPRXOPTERR[n]	Protocol Engine Statistics TCP Received Segments with Unsupported Options	3602
0x00013300 + 0x4*n, n=0...15	GLPES_PFTCPRXPROTOERR[n]	Protocol Engine Statistics TCP Dropped Segments Due to Protocol Errors	3602
0x00013400 + 0x8*n, n=0...15	GLPES_PFTCPTXSEGLO[n]	Protocol Engine Statistics TCP Transmitted Segments Low	3603

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x00013404 + 0x8*n, n=0...15	GLPES_PFTCPTXSEGHI[n]	Protocol Engine Statistics TCP Transmitted Segments High	3603
0x00013600 + 0x4*n, n=0...15	GLPES_PFTCPRTXSEG[n]	Protocol Engine Statistics TCP Retransmitted Segments	3603
0x00013800 + 0x8*n, n=0...15	GLPES_PFUDPRXPKTSLO[n]	Protocol Engine Statistics UDP Received Packets Low	3603
0x00013804 + 0x8*n, n=0...15	GLPES_PFUDPRXPKTSHI[n]	Protocol Engine Statistics UDP Received Packets High	3603
0x00013A00 + 0x8*n, n=0...15	GLPES_PFUDPTXPKTSLO[n]	Protocol Engine Statistics UDP Transmitted Packets Low	3604
0x00013A04 + 0x8*n, n=0...15	GLPES_PFUDPTXPKTSHI[n]	Protocol Engine Statistics UDP Transmitted Packets High	3604
0x00013C00 + 0x8*n, n=0...15	GLPES_PFRDMARXWRSLO[n]	Protocol Engine Statistics RDMA Received Write Messages Low	3604
0x00013C04 + 0x8*n, n=0...15	GLPES_PFRDMARXWRSHI[n]	Protocol Engine Statistics RDMA Received Write Messages High	3604
0x00013E00 + 0x8*n, n=0...15	GLPES_PFRDMARXRDSLO[n]	Protocol Engine Statistics RDMA Received Read Request Messages Low	3605
0x00013E04 + 0x8*n, n=0...15	GLPES_PFRDMARXRDSHI[n]	Protocol Engine Statistics RDMA Received Read Request Messages High	3605
0x00014000 + 0x8*n, n=0...15	GLPES_PFRDMARXSNDSLO[n]	Protocol Engine Statistics RDMA Received Send Messages Low	3605
0x00014004 + 0x8*n, n=0...15	GLPES_PFRDMARXSNDSHI[n]	Protocol Engine Statistics RDMA Received Send Messages High	3605
0x00014200 + 0x8*n, n=0...15	GLPES_PFRDMATXWRSLO[n]	Protocol Engine Statistics RDMA Transmitted Write Messages Low	3606
0x00014204 + 0x8*n, n=0...15	GLPES_PFRDMATXWRSHI[n]	Protocol Engine Statistics RDMA Transmitted Write Messages High	3606
0x00014400 + 0x8*n, n=0...15	GLPES_PFRDMATXRDSLO[n]	Protocol Engine Statistics RDMA Transmitted Read Request Messages Low	3606
0x00014404 + 0x8*n, n=0...15	GLPES_PFRDMATXRDSHI[n]	Protocol Engine Statistics RDMA Transmitted Read Request Messages High	3606
0x00014600 + 0x8*n, n=0...15	GLPES_PFRDMATXSNDSLO[n]	Protocol Engine Statistics RDMA Transmitted Send Messages Low	3607
0x00014604 + 0x8*n, n=0...15	GLPES_PFRDMATXSNDSHI[n]	Protocol Engine Statistics RDMA Transmitted Send Messages High	3607
0x00014800 + 0x8*n, n=0...15	GLPES_PFRDMAVBNDLO[n]	Protocol Engine Statistics RDMA Verbs Bind Operations Low	3607
0x00014804 + 0x8*n, n=0...15	GLPES_PFRDMAVBNDHI[n]	Protocol Engine Statistics RDMA Verbs Bind Operations High	3607
0x00014A00 + 0x8*n, n=0...15	GLPES_PFRDMAVINVLO[n]	Protocol Engine Statistics RDMA Verbs Invalidate Operations Low	3608
0x00014A04 + 0x8*n, n=0...15	GLPES_PFRDMAVINVHI[n]	Protocol Engine Statistics RDMA Verbs Invalidate Operations High	3608
0x00018000 + 0x4*n, n=0...31	GLPES_VFRXVLANERR[n]	Protocol Engine Statistics Received VLAN_ID Errors	3608
0x00018200 + 0x8*n, n=0...31	GLPES_VFIP4RXOCTSLO[n]	Protocol Engine Statistics IPv4 Received Octets Low	3608
0x00018204 + 0x8*n, n=0...31	GLPES_VFIP4RXOCTSHI[n]	Protocol Engine Statistics IPv4 Received Octets High	3609

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x00018400 + 0x8*n, n=0...31	GLPES_VFIP4RXPKTSLO[n]	Protocol Engine Statistics IPv4 Received Packets Low	3609
0x00018404 + 0x8*n, n=0...31	GLPES_VFIP4RXPKTSHI[n]	Protocol Engine Statistics IPv4 Received Packets High	3609
0x00018600 + 0x4*n, n=0...31	GLPES_VFIP4RXDISCARD[n]	Protocol Engine Statistics IPv4 Discards	3609
0x00018700 + 0x4*n, n=0...31	GLPES_VFIP4RXTRUNC[n]	Protocol Engine Statistics IPv4 Truncated Packets	3610
0x00018800 + 0x8*n, n=0...31	GLPES_VFIP4RXFRAGSLO[n]	Protocol Engine Statistics IPv4 Received Fragments Low	3610
0x00018804 + 0x8*n, n=0...31	GLPES_VFIP4RXFRAGSHI[n]	Protocol Engine Statistics IPv4 Received Fragments High	3610
0x00018A00 + 0x8*n, n=0...31	GLPES_VFIP4RXMCOCTSLO[n]	Protocol Engine Statistics IPv4 Received Multicast Octets Low	3610
0x00018A04 + 0x8*n, n=0...31	GLPES_VFIP4RXMCOCTSHI[n]	Protocol Engine Statistics IPv4 Received Multicast Octets High	3611
0x00018C00 + 0x8*n, n=0...31	GLPES_VFIP4RXMCPKTSLO[n]	Protocol Engine Statistics IPv4 Received Multicast Packets Low	3611
0x00018C04 + 0x8*n, n=0...31	GLPES_VFIP4RXMCPKTSHI[n]	Protocol Engine Statistics IPv4 Received Multicast Packets High	3611
0x00018E00 + 0x8*n, n=0...31	GLPES_VFIP6RXOCTSLO[n]	Protocol Engine Statistics IPv6 Received Octets Low	3611
0x00018E04 + 0x8*n, n=0...31	GLPES_VFIP6RXOCTSHI[n]	Protocol Engine Statistics IPv6 Received Octets High	3612
0x00019000 + 0x8*n, n=0...31	GLPES_VFIP6RXPKTSLO[n]	Protocol Engine Statistics IPv6 Received Packets Low	3612
0x00019004 + 0x8*n, n=0...31	GLPES_VFIP6RXPKTSHI[n]	Protocol Engine Statistics IPv6 Received Packets High	3612
0x00019200 + 0x4*n, n=0...31	GLPES_VFIP6RXDISCARD[n]	Protocol Engine Statistics IPv6 Discards	3612
0x00019300 + 0x4*n, n=0...31	GLPES_VFIP6RXTRUNC[n]	Protocol Engine Statistics IPv6 Truncated Packets	3613
0x00019400 + 0x8*n, n=0...31	GLPES_VFIP6RXFRAGSLO[n]	Protocol Engine Statistics IPv6 Received Fragments Low	3613
0x00019404 + 0x8*n, n=0...31	GLPES_VFIP6RXFRAGSHI[n]	Protocol Engine Statistics IPv6 Received Fragments High	3613
0x00019600 + 0x8*n, n=0...31	GLPES_VFIP6RXMCOCTSLO[n]	Protocol Engine Statistics IPv6 Received Multicast Octets Low	3613
0x00019604 + 0x8*n, n=0...31	GLPES_VFIP6RXMCOCTSHI[n]	Protocol Engine Statistics IPv6 Received Multicast Octets High	3614
0x00019800 + 0x8*n, n=0...31	GLPES_VFIP6RXMCPKTSLO[n]	Protocol Engine Statistics IPv6 Received Multicast Packets Low	3614
0x00019804 + 0x8*n, n=0...31	GLPES_VFIP6RXMCPKTSHI[n]	Protocol Engine Statistics IPv6 Received Multicast Packets High	3614
0x00019A00 + 0x8*n, n=0...31	GLPES_VFIP4TXOCTSLO[n]	Protocol Engine Statistics IPv4 Transmitted Octets Low	3614
0x00019A04 + 0x8*n, n=0...31	GLPES_VFIP4TXOCTSHI[n]	Protocol Engine Statistics IPv4 Transmitted Octets High	3615
0x00019C00 + 0x8*n, n=0...31	GLPES_VFIP4TXPKTSLO[n]	Protocol Engine Statistics IPv4 Transmitted Packets Low	3615
0x00019C04 + 0x8*n, n=0...31	GLPES_VFIP4TXPKTSHI[n]	Protocol Engine Statistics IPv4 Transmitted Packets High	3615

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x00019E00 + 0x8*n, n=0...31	GLPES_VFIP4TXFRAGSLO[n]	Protocol Engine Statistics IPv4 Transmitted Fragments Low	3615
0x00019E04 + 0x8*n, n=0...31	GLPES_VFIP4TXFRAGSHI[n]	Protocol Engine Statistics IPv4 Transmitted Fragments High	3616
0x0001A000 + 0x8*n, n=0...31	GLPES_VFIP4TXMCOCTSLO[n]	Protocol Engine Statistics IPv4 Transmitted Multicast Octets Low	3616
0x0001A004 + 0x8*n, n=0...31	GLPES_VFIP4TXMCOCTSHI[n]	Protocol Engine Statistics IPv4 Transmitted Multicast Octets High	3616
0x0001A200 + 0x8*n, n=0...31	GLPES_VFIP4TXMCPKTSLO[n]	Protocol Engine Statistics IPv4 Transmitted Multicast Packets Low	3616
0x0001A204 + 0x8*n, n=0...31	GLPES_VFIP4TXMCPKTSHI[n]	Protocol Engine Statistics IPv4 Transmitted Multicast Packets High	3617
0x0001A400 + 0x8*n, n=0...31	GLPES_VFIP6TXOCTSLO[n]	Protocol Engine Statistics IPv6 Transmitted Octets Low	3617
0x0001A404 + 0x8*n, n=0...31	GLPES_VFIP6TXOCTSHI[n]	Protocol Engine Statistics IPv6 Transmitted Octets High	3617
0x0001A600 + 0x8*n, n=0...31	GLPES_VFIP6TXPKTSLO[n]	Protocol Engine Statistics IPv6 Transmitted Packets Low	3617
0x0001A604 + 0x8*n, n=0...31	GLPES_VFIP6TXPKTSHI[n]	Protocol Engine Statistics IPv6 Transmitted Packets High	3618
0x0001A800 + 0x8*n, n=0...31	GLPES_VFIP6TXFRAGSLO[n]	Protocol Engine Statistics IPv6 Transmitted Fragments Low	3618
0x0001A804 + 0x8*n, n=0...31	GLPES_VFIP6TXFRAGSHI[n]	Protocol Engine Statistics IPv6 Transmitted Fragments High	3618
0x0001AA00 + 0x8*n, n=0...31	GLPES_VFIP6TXMCOCTSLO[n]	Protocol Engine Statistics IPv6 Transmitted Multicast Octets Low	3618
0x0001AA04 + 0x8*n, n=0...31	GLPES_VFIP6TXMCOCTSHI[n]	Protocol Engine Statistics IPv6 Transmitted Multicast Octets High	3619
0x0001AC00 + 0x8*n, n=0...31	GLPES_VFIP6TXMCPKTSLO[n]	Protocol Engine Statistics IPv6 Transmitted Multicast Packets Low	3619
0x0001AC04 + 0x8*n, n=0...31	GLPES_VFIP6TXMCPKTSHI[n]	Protocol Engine Statistics IPv6 Transmitted Multicast Packets High	3619
0x0001AE00 + 0x4*n, n=0...31	GLPES_VFIP4TXNOROUTE[n]	Protocol Engine Statistics IPv4 Discarded No Route Packets	3619
0x0001AF00 + 0x4*n, n=0...31	GLPES_VFIP6TXNOROUTE[n]	Protocol Engine Statistics IPv6 Discarded No Route Packets	3620
0x0001B000 + 0x8*n, n=0...31	GLPES_VFTCPRXSEGSLO[n]	Protocol Engine Statistics TCP Received Segments Low	3620
0x0001B004 + 0x8*n, n=0...31	GLPES_VFTCPRXSEGSHI[n]	Protocol Engine Statistics TCP Received Segments High	3620
0x0001B200 + 0x4*n, n=0...31	GLPES_VFTCPRXOPTERR[n]	Protocol Engine Statistics TCP Received Segments with Unsupported Options	3620
0x0001B300 + 0x4*n, n=0...31	GLPES_VFTCPRXPROTOERR[n]	Protocol Engine Statistics TCP Dropped Segments Due to Protocol Errors	3621
0x0001B400 + 0x8*n, n=0...31	GLPES_VFTCPTXSEGLO[n]	Protocol Engine Statistics TCP Transmitted Segments Low	3621
0x0001B404 + 0x8*n, n=0...31	GLPES_VFTCPTXSEGHI[n]	Protocol Engine Statistics TCP Transmitted Segments High	3621
0x0001B600 + 0x4*n, n=0...31	GLPES_VFTCPRTXSEG[n]	Protocol Engine Statistics TCP Retransmitted Segments	3621
0x0001B800 + 0x8*n, n=0...31	GLPES_VFUDPRXPKTSLO[n]	Protocol Engine Statistics UDP Received Packets Low	3622

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x0001B804 + 0x8*n, n=0...31	GLPES_VFUDPRXPKTSHI[n]	Protocol Engine Statistics UDP Received Packets High	3622
0x0001BA00 + 0x8*n, n=0...31	GLPES_VFUDPTXPKTSLO[n]	Protocol Engine Statistics UDP Transmitted Packets Low	3622
0x0001BA04 + 0x8*n, n=0...31	GLPES_VFUDPTXPKTSHI[n]	Protocol Engine Statistics UDP Transmitted Packets High	3622
0x0001BC00 + 0x8*n, n=0...31	GLPES_VFRDMARXWRSLO[n]	Protocol Engine Statistics RDMA Received Write Messages Low	3623
0x0001BC04 + 0x8*n, n=0...31	GLPES_VFRDMARXWRSHI[n]	Protocol Engine Statistics RDMA Received Write Messages High	3623
0x0001BE00 + 0x8*n, n=0...31	GLPES_VFRDMARXRDSLO[n]	Protocol Engine Statistics RDMA Received Read Request Messages Low	3623
0x0001BE04 + 0x8*n, n=0...31	GLPES_VFRDMARXRDSHI[n]	Protocol Engine Statistics RDMA Received Read Request Messages High	3623
0x0001C000 + 0x8*n, n=0...31	GLPES_VFRDMARXSNDSLO[n]	Protocol Engine Statistics RDMA Received Send Messages Low	3624
0x0001C004 + 0x8*n, n=0...31	GLPES_VFRDMARXSNDSHI[n]	Protocol Engine Statistics RDMA Received Send Messages High	3624
0x0001C200 + 0x8*n, n=0...31	GLPES_VFRDMATXWRSLO[n]	Protocol Engine Statistics RDMA Transmitted Write Messages Low	3624
0x0001C204 + 0x8*n, n=0...31	GLPES_VFRDMATXWRSHI[n]	Protocol Engine Statistics RDMA Transmitted Write Messages High	3624
0x0001C400 + 0x8*n, n=0...31	GLPES_VFRDMATXRDSLO[n]	Protocol Engine Statistics RDMA Transmitted Read Request Messages Low	3625
0x0001C404 + 0x8*n, n=0...31	GLPES_VFRDMATXRDSHI[n]	Protocol Engine Statistics RDMA Transmitted Read Request Messages High	3625
0x0001C600 + 0x8*n, n=0...31	GLPES_VFRDMATXSNDSLO[n]	Protocol Engine Statistics RDMA Transmitted Send Messages Low	3625
0x0001C604 + 0x8*n, n=0...31	GLPES_VFRDMATXSNDSHI[n]	Protocol Engine Statistics RDMA Transmitted Send Messages High	3625
0x0001C800 + 0x8*n, n=0...31	GLPES_VFRDMAVBNDLO[n]	Protocol Engine Statistics RDMA Verbs Bind Operations Low	3626
0x0001C804 + 0x8*n, n=0...31	GLPES_VFRDMAVBNDHI[n]	Protocol Engine Statistics RDMA Verbs Bind Operations High	3626
0x0001CA00 + 0x8*n, n=0...31	GLPES_VFRDMAVINVLO[n]	Protocol Engine Statistics RDMA Verbs Invalidate Operations Low	3626
0x0001CA04 + 0x8*n, n=0...31	GLPES_VFRDMAVINVHI[n]	Protocol Engine Statistics RDMA Verbs Invalidate Operations High	3626
0x0001E000	GLPES_RDMARXUNALIGN	Protocol Engine Statistics RDMA Received Unaligned FPDUs	3627
0x0001E004	GLPES_RDMARXOOONOMARK	Protocol Engine Statistics RDMA Received Out of Order No Markers FPDUs	3627
0x0001E010	GLPES_RDMARXMULTFPDUSLO	Protocol Engine Statistics RDMA Received Multiple FPDUs Low	3627
0x0001E014	GLPES_RDMARXMULTFPDUSHI	Protocol Engine Statistics RDMA Received Multiple FPDUs High	3627
0x0001E018	GLPES_RDMARXOOODDPLO	Protocol Engine Statistics RDMA Out of Order Placed DDP Segments Low	3627
0x0001E01C	GLPES_RDMARXOOODDPHI	Protocol Engine Statistics RDMA Out of Order Placed DDP Segments High	3627

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x0001E020	GLPES_TCPRXPUREACKSLO	Protocol Engine Statistics TCP Received Pure Acks Low	3628
0x0001E024	GLPES_TCPRXPUREACKHI	Protocol Engine Statistics TCP Received Pure Acks High	3628
0x0001E028	GLPES_TCPRXONEHOLELO	Protocol Engine Statistics TCP Receive First Hole Low	3628
0x0001E02C	GLPES_TCPRXONEHOLEHI	Protocol Engine Statistics TCP Received First Hole High	3628
0x0001E030	GLPES_TCPRXTWOHOLELO	Protocol Engine Statistics TCP Receive Second Hole Low	3628
0x0001E034	GLPES_TCPRXTWOHOLEHI	Protocol Engine Statistics TCP Received Second Hole High	3628
0x0001E038	GLPES_TCPRXTHREEHOLELO	Protocol Engine Statistics TCP Receive Third Hole Low	3629
0x0001E03C	GLPES_TCPRXTHREEHOLEHI	Protocol Engine Statistics TCP Received Third Hole High	3629
0x0001E040	GLPES_TCPRXFOURHOLELO	Protocol Engine Statistics TCP Receive Fourth Hole Low	3629
0x0001E044	GLPES_TCPRXFOURHOLEHI	Protocol Engine Statistics TCP Receive Fourth Hole High	3629
0x0001E048	GLPES_TCPTXRETRANSFASTLO	Protocol Engine Statistics TCP Fast Retransmissions Low	3629
0x0001E04C	GLPES_TCPTXRETRANSFASTHI	Protocol Engine Statistics TCP Fast Retransmissions High	3630
0x0001E050	GLPES_TCPTXTOUTSFASTLO	Protocol Engine Statistics TCP Fast Retransmissions Timeouts Low	3630
0x0001E054	GLPES_TCPTXTOUTSFASTHI	Protocol Engine Statistics TCP Fast Retransmissions Timeouts High	3630
0x0001E058	GLPES_TCPTXTOUTSLO	Protocol Engine Statistics TCP Retransmissions Timeouts Low	3630
0x0001E05C	GLPES_TCPTXTOUTSHI	Protocol Engine Statistics TCP Retransmissions Timeouts High	3630
PF - LAN Transmit Receive Registers			
0x000442D8	GLLAN_TSOMSK_F	Global TSO TCP Mask First	3631
0x000442DC	GLLAN_TSOMSK_M	Global TSO TCP Mask Middle	3631
0x000442E0	GLLAN_TSOMSK_L	Global TSO TCP Mask Last	3631
0x00051060	GL_RDPU_CNTRL	Receive Processing Block Control	3631
0x00070000 + 0x400*n + 0x4*VF, n=0...15, VF=0...127	VPLAN_QTABLE[n,VF]	VF PF Queue Mapping Table	3631
0x00074000 + 0x4*VF, VF=0...127	VPLAN_MAPENA[VF]	VF LAN Enablement	3632
0x00074800 + 0x4*VF, VF=0...127	VPLAN_QBASE[VF]	VF PF Queue Range	3632
0x000E4000 + 0x4*Q, Q=0...1535	QTX_HEAD[Q]	Global Transmit Queue Head	3632
0x000E6500 + 0x4*n, n=0...11	GLLAN_TXPRE_QDIS[n]	Global Transmit Pre Queue Disable	3632
0x00100000 + 0x4*Q, Q=0...1535	QTX_ENA[Q]	Global Transmit Queue Enable	3633
0x00104000 + 0x4*Q, Q=0...1535	QTX_CTL[Q]	Global Transmit Queue Control	3633
0x00108000 + 0x4*Q, Q=0...1535	QTX_TAIL[Q]	Global Transmit Queue Tail	3633
0x00120000 + 0x4*Q, Q=0...1535	QRX_ENA[Q]	Global Receive Queue Enable	3634
0x00128000 + 0x4*Q, Q=0...1535	QRX_TAIL[Q]	Global Receive Queue Tail	3634
0x0012A500	GLLAN_RCTL_0	Global RLAN Control 0	3634

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x001C0400	PFLAN_QALLOC	PF Queue Allocation	3634
0x00200000 + 0x800*n + 0x4*VSI, n=0...7, VSI=0...383	VSILAN_QTABLE[n,VSI]	VSI Receive Queue Mapping Table	3635
0x0020C800 + 0x4*VSI, VSI=0...383	VSILAN_QBASE[VSI]	VSI Queue Control	3635
PF - Rx Filters Registers			
0x001C0000 + 0x4*VF, VF=0...127	VPQF_CTL[VF]	VF Queue Filter Control	3635
0x001C0AC0	PFQF_CTL_0	PF Queue Filter Control 0	3636
0x00206000 + 0x800*n + 0x4*VSI, n=0...3, VSI=0...383	VSIQF_TCREGION[n,VSI]	VSI Receive Traffic Class Queues	3636
0x0020D800 + 0x4*VSI, VSI=0...383	VSIQF_CTL[VSI]	VSI Queue Filter Control	3637
0x00220000 + 0x800*n + 0x4*VSI, n=0...15, VSI=0...383	VSIQF_HLUT[n,VSI]	VF Queue Filter Hash LUT	3637
0x0022E000 + 0x400*n + 0x4*VF, n=0...7, VF=0...127	VFQF_HREGION[n,VF]	VF Queue Filter Hash Region of Queues	3638
0x00230800 + 0x400*n + 0x4*VF, n=0...1, VF=0...127	VFQF_HENA[n,VF]	VF Queue Filter Hash Enabled Packet Type	3638
0x00240000 + 0x80*n, n=0...127	PFQF_HLUT[n]	PF Queue Filter Hash LUT	3639
0x00245400 + 0x80*n, n=0...7	PFQF_HREGION[n]	PF Queue Filter Hash Region of Queues	3639
0x00245900 + 0x80*n, n=0...1	PFQF_HENA[n]	PF Queue Filter Hash Enabled Packet Type	3640
0x00245D80	PFQF_CTL_1	PF Queue Filter Control 1	3640
0x00246280	PFQF_FDALLOC	PF Queue Filter Flow Director Allocation	3640
0x00246380	PFQF_FDSTAT	PF Queue Filter Flow Director Allocation Status	3641
0x00253800 + 0x20*n, n=0...63	PRTQF_FD_FLXINSET[n]	Port Queue Filter Flow Director Input Set	3641
0x00255200 + 0x20*n, n=0...8	PRTQF_FLX_PIT[n]	Port Queue Filter - Flexible Parser Information Table	3641
0x00256E60	PRTQF_CTL_0	Port Queue Filter Control 0	3642
0x00260000 + 0x4*n, n=0...2047	GLQF_APBVT[n]	Global Queue Filter Accelerated Port Bit Vector	3642
0x00266800 + 0x4*n, n=0...511	GLQF_PCNT[n]	Global Queue Filter Packet Counter	3642
0x00267E00 + 0x4*n + 0x8*m, n=0...1, m=0...63	GLQF_SWAP[n,m]	Global Queue Filter SWAP Fields	3642
0x00268000 + 0x4*n, n=0...63	GLQF_FD_PCTYPES[n]	Flow Director PCTYPE Translation Table	3643
0x00269BA4	GLQF_CTL	Global Queue Filter Control	3643
0x00269BAC	GLQF_FDCNT_0	Global Queue Filter Flow Director Status 0	3644
0x00269D00 + 0x4*n, n=0...63	GLQF_HSYM[n]	Global Queue Filter Symmetric Hash Enablement	3644
0x00270140 + 0x4*n, n=0...12	GLQF_HKEY[n]	Global Queue Filter Hash Key	3644
0x00270300	PFQF_CTL_2	PF Queue Filter Control 2	3645
0x00270384 + 0x4*n, n=0...1	GLQF_FDEVICTENA[n]	Global Flow Director PCTYPE Evict Enable	3645
0x00270280	GLQF_FDEVICTFLAG	Global Flow Director Flag Evict Enable	3645

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x002A0000 + 0x800*n + 0x4*VSI, n=0...12, VSI=0...383	VSIQF_HKEY[n,VSI]	VSI Queue Filter Hash Key	3646
PF - Timesync (IEEE 1588) Registers			
0x00085020	PRTTSYN_CTL1	Port Time Sync Control 1	3646
0x00085040 + 0x20*n, n=0...3	PRTTSYN_RXTIME_H[n]	port Time Sync Receive PTP Packet Time High	3647
0x000850C0 + 0x20*n, n=0...3	PRTTSYN_RXTIME_L[n]	Port Time Sync Receive PTP Packet Time Low	3647
0x00085140	PRTTSYN_STAT_1	Port Time Sync Status 1	3647
0x001E4040	PRTTSYN_INC_L	Port Time Sync Increment Value Low	3648
0x001E4060	PRTTSYN_INC_H	Port Time Sync Increment Value High	3648
0x001E4080 + 0x20*n, n=0...1	PRTTSYN_EVNT_L[n]	Port Time Sync Event Time Low	3648
0x001E40C0 + 0x20*n, n=0...1	PRTTSYN_EVNT_H[n]	Port Time Sync Event Time High	3648
0x001E4100	PRTTSYN_TIME_L	Port Time Sync Time Low	3648
0x001E4120	PRTTSYN_TIME_H	Port Time Sync Time High	3648
0x001E4140 + 0x20*n, n=0...1	PRTTSYN_TGT_L[n]	Port Time Sync Target Time Low	3648
0x001E4180 + 0x20*n, n=0...1	PRTTSYN_TGT_H[n]	Port Time Sync Target Time High	3649
0x001E41C0	PRTTSYN_TXTIME_L	Port Time Sync Transmit Packet Time Low	3649
0x001E41E0	PRTTSYN_TXTIME_H	Port Time Sync Transmit Packet Time High	3649
0x001E4200	PRTTSYN_CTL0	Port Time Sync Control 0	3649
0x001E4220	PRTTSYN_STAT_0	Port Time Sync Status 0	3650
0x001E4240 + 0x20*n, n=0...1	PRTTSYN_CLKO[n]	Port Time Sync Clock Out Duration	3650
0x001E4280	PRTTSYN_ADJ	Port Time Sync Adjustment	3650
0x001E42A0 + 0x20*n, n=0...1	PRTTSYN_AUX_0[n]	Port Time Sync AUX Control 0	3650
0x001E42E0 + 0x20*n, n=0...1	PRTTSYN_AUX_1[n]	Port Time Sync AUX Control 1	3651
PF - Protocol Engine Registers			
0x00000000 + 0x4*VF, VF=0...127	VFPE_CQPDB[VF]	Protocol Engine VF CQP Doorbell	3651
0x00000400 + 0x4*VF, VF=0...127	VFPE_CQPTAIL[VF]	Protocol Engine VF CQP Tail	3652
0x00000800 + 0x4*VF, VF=0...127	VFPE_CCQPSTATUS[VF]	Protocol Engine VF Create CQP Status	3652
0x00000C00 + 0x4*VF, VF=0...127	VFPE_CCQPLOW[VF]	Protocol Engine VF Create CQP Low	3653
0x00001000 + 0x4*VF, VF=0...127	VFPE_CCQPHIGH[VF]	Protocol Engine VF Create CQP High	3653
0x00001400 + 0x4*VF, VF=0...127	VFPE_IPCONFIG0[VF]	Protocol Engine VF IP Config 0	3653
0x00001800 + 0x4*VF, VF=0...127	VFPE_CQPERRCODES[VF]	Protocol Engine CQP Error Codes	3653
0x00002C00 + 0x4*VF, VF=0...127	VFPE_TCPNOWTIMER[VF]	Protocol Engine VF TCP Now Timer	3654
0x00003000 + 0x4*VF, VF=0...127	VFPE_MRTEIDXMASK[VF]	Protocol Engine VF MRTE Index Mask	3654
0x00003400 + 0x4*VF, VF=0...127	VFPE_RCVUNEXPECTEDERROR[VF]	Protocol Engine VF Unexpected Error	3654
0x00008000	PFPE_CQPDB	Protocol Engine CQP Doorbell	3654
0x00008080	PFPE_CQPTAIL	Protocol Engine CQP Tail	3655
0x00008100	PFPE_CCQPSTATUS	Protocol Engine Create CQP Status	3655
0x00008180	PFPE_CCQPLOW	Protocol Engine Create CQP Low	3656

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x00008200	PFPE_CCQPHIGH	Protocol Engine Create CQP High	3656
0x00008280	PFPE_IPCONFIG0	Protocol Engine IP Config 0	3656
0x00008580	PFPE_TCPNOWTIMER	Protocol Engine TCP Now Timer	3656
0x00008600	PFPE_MRTEIDXMASK	Protocol Engine MRTE Index Mask	3656
0x00008680	PFPE_RCVUNEXPECTEDERROR	Protocol Engine Unexpected Error	3657
0x00008700	PFPE_UDACTRL	Protocol Engine UDA Control Register	3657
0x00008780	PFPE_UDAUCFBQPN	Protocol Engine UDA Unicast Fallback QP Number	3658
0x00008880	PFPE_CQPERRCODES	Protocol Engine CQP Error Codes	3658
0x00008900	PFPE_FLMXMITALLOCERR	Protocol Engine FLM Transmit Allocate Error	3658
0x00008980	PFPE_FLMQ1ALLOCERR	Protocol Engine FLM Q1 Allocate Error	3658
0x0000C000 + 0x4*n, n=0...31	GLPE_VFUDACTRL[n]	Protocol Engine UDA Control Register	3659
0x0000C100 + 0x4*n, n=0...31	GLPE_VFUDAUCFBQPN[n]	Protocol Engine UDA Unicast Fallback QP Number	3660
0x0000C600 + 0x4*n, n=0...31	GLPE_VFFLMXMITALLOCERR[n]	Protocol Engine VF FLM Transmit Allocate Error	3660
0x0000C700 + 0x4*n, n=0...31	GLPE_VFFLMQ1ALLOCERR[n]	Protocol Engine VF FLM Q1 Allocate Error	3660
0x0000D040	GLPE_CPUSTATUS0	Protocol Engine CPU Status 0	3660
0x0000D044	GLPE_CPUSTATUS1	Protocol Engine CPU Status 1	3661
0x0000D048	GLPE_CPUSTATUS2	Protocol Engine CPU Status 2	3661
0x0000D060	GLPE_CPUTRIG0	Protocol Engine CPU Trigger 0	3661
0x0000D400 + 0x4*n, n=0...31	GLPE_VFFLMOBJCTRL[n]	Protocol Engine VF FLM Object Control	3661
0x0000DA00	GLPE_RUPM_GCTL	Protocol Engine RUPM General Control Register	3662
0x0000DA04	GLPE_DUAL40_RUPM	Protocol Engine RUPM DUAL40 Config Register	3662
0x0000DA08	GLPE_RUPM_TXHOST_EN	Protocol Engine RUPM Enable Register	3662
0x0000DA20	PRTPE_RUPM_THRES	Protocol Engine RUPM Spad Thresholds	3662
0x0000DA40	PRTPE_RUPM_CTL	Protocol Engine RUPM Control Register	3663
0x0000DA60	PRTPE_RUPM_PFCCTL	Protocol Engine RUPM PFC Control Register	3663
0x0000DA80	PRTPE_RUPM_PFCPC	Protocol Engine RUPM PFC Port Control Register	3663
0x0000DAA0	PRTPE_RUPM_PFCTCC	Protocol Engine RUPM PFC TC Control Register	3664
0x0000DAC0	GLPE_RUPM_PUSHPOOL	Protocol Engine RUPM Push Pool	3664
0x0000DAC4	GLPE_RUPM_FLRPOOL	Protocol Engine RUPM FLR Pool	3664
0x0000DAC8	GLPE_RUPM_PTXPOOL	Protocol Engine RUPM PTX Pool	3664
0x0000DACC	GLPE_RUPM_CQPPOOL	Protocol Engine RUPM CQP Pool	3664

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x0000DAE0	PRTE_RUPM_TCCNTR03	Protocol Engine RUPM TC Counters 0-3	3665
0x0000DB00	PRTPE_RUPM_TCCNTR47	Protocol Engine RUPM TC Counters 4-7	3665
0x0000DB20	PRTPE_RUPM_CNTR	Protocol Engine RUPM Counters	3665
0x0000DB40	PRTPE_RUPM_PTXTCCNTR03	Protocol Engine RUPM PTX TC Counters 0-3	3665
0x0000DB60	PRTPE_RUPM_PTCTCCNTR47	Protocol Engine RUPM PTX TC Counters 4-7	3666
0x0001F040 + 0x20*n, n=0...7	PRTDCB_TCMSTC_RLPM[n]	DCB Transmit Command Monitoring Status per TC	3666
0x0001F140	PRTDCB_RLPMC	DCB TC to PFC Mapping	3666
0x0001F1A0	PRTDCB_TCPMC_RLPM	DCB Transmit Command Pipe Monitor Control	3666
0x00130400 + 0x4*VF, VF=0...127	VFPE_CQARM[VF]	Protocol Engine VF CQ Arm	3667
0x00130800 + 0x4*VF, VF=0...127	VFPE_CQACK[VF]	Protocol Engine VF CQ Ack	3667
0x00130C00 + 0x4*VF, VF=0...127	VFPE_AEQALLOC[VF]	Protocol Engine VF AEQ Allocate	3667
0x00131080	PFPE_CQARM	Protocol Engine CQ Arm	3668
0x00131100	PFPE_CQACK	Protocol Engine CQ Ack	3668
0x00131180	PFPE_AEQALLOC	Protocol Engine AEQ Allocate	3668
0x00131340 + 0x4*n, n=0...15	GLPE_PFCQEDROPCNT[n]	Protocol Engine CQE Drop Count	3668
0x001313C0 + 0x4*n, n=0...15	GLPE_PFCEQEDROPCNT[n]	Protocol Engine CEQE Drop Count	3669
0x00131440 + 0x4*n, n=0...15	GLPE_PFAEQEDROPCNT[n]	Protocol Engine AEQE Drop Count	3669
0x00132340 + 0x4*n, n=0...31	GLPE_VFCQEDROPCNT[n]	Protocol Engine CQE Drop Count	3669
0x00132440 + 0x4*n, n=0...31	GLPE_VFCEQEDROPCNT[n]	Protocol Engine CEQE Drop Count	3669
0x00132540 + 0x4*n, n=0...31	GLPE_VFAEQEDROPCNT[n]	Protocol Engine AEQE Drop Count	3670
0x00138000 + 0x4*VF, VF=0...127	VFPE_WQEALLOC[VF]	Protocol Engine VF WQE Allocate Register	3670
0x00138C00	PFPE_WQEALLOC	Protocol Engine WQE Allocate Register	3670
PF - Manageability Registers			
0x00083100	GL_FWRESETCNT	Firmware Reset Count	3671
0x00085160 + 0x20*n, n=0...7	PRT_MNG_FTFT_MASK[n]	Flexible TCO Filter Table Registers - Mask	3671
0x00085260	PRT_MNG_FTFT_LENGTH	Flexible TCO Filter Table Registers - Length	3671
0x000852A0 + 0x20*n, n=0...31	PRT_MNG_FTFT_DATA[n]	Flexible TCO Filter Table Registers - Data	3671
0x000B6130	GL_MNG_HWARB_CTRL	Hardware Arbitration Control	3672
0x000B6134	GL_MNG_FWSM	Firmware Semaphore	3672
0x00254200 + 0x20*n, n=0...15	PRT_MNG_MIPAF6[n]	Manageability IPv6 Address Filter	3674
0x00254E00 + 0x20*n, n=0...15	PRT_MNG_MFUTP[n]	Management Flex UDP/TCP Ports	3674
0x00255900 + 0x20*n, n=0...7	PRT_MNG_MAVTV[n]	Management VLAN TAG Value	3674
0x00255D00 + 0x20*n, n=0...7	PRT_MNG_MDEF[n]	Manageability Decision Filters1	3675
0x00255F00 + 0x20*n, n=0...7	PRT_MNG_MDEF_EXT[n]	Manageability Decision Filters	3676
0x00256280 + 0x20*n, n=0...3	PRT_MNG_MIPAF4[n]	Manageability IPv4 Address Filter	3677

Table 38-609.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x00256380 + 0x20*n, n=0...3	PRT_MNG_MMAH[n]	Manageability MAC Address High	3677
0x00256480 + 0x20*n, n=0...3	PRT_MNG_MMAL[n]	Manageability MAC Address Low	3677
0x00256580 + 0x20*n, n=0...3	PRT_MNG_MDEFVSI[n]	Management Decision Filters Buffers	3678
0x00256780 + 0x20*n, n=0...3	PRT_MNG_METF[n]	Management Ethernet Type Filters	3678
0x00256A20	PRT_MNG_MANC	Management Control Register	3678
0x00256A60	PRT_MNG_MNGONLY	Management Only Traffic Register	3679
0x00256AA0	PRT_MNG_MSFM	Manageability Special Filters Modifiers	3680

38.39.2 Detailed Register Description - BAR0

38.39.2.1 PF - General Registers

This category contains registers for general device control and status.

38.39.2.1.1 VF Reset Status - VFGEN_RSTAT[VF] (0x00074400 + 0x4*VF, VF=0...127; RW)

Field	Bit(s)	Init.	Type	Description
VFR_STATE	1:0	00b	RW	**VFR State** Defines the VFR reset progress as follows: 00b = VFR in progress 01b = VFR completed 10b = Reserved 11b = Reserved This field is used to communicate the reset progress to the VF with no impact on hardware functionality.
RESERVED	31:2	0x0	RSV	Reserved.

38.39.2.1.2 Firmware Status Register - GL_FWSTS (0x00083048; RO)

This register reports the status of the EMP.

Field	Bit(s)	Init.	Type	Description
FWS0B	7:0	0x0	RO	**Firmware Status 0 Byte** This bit is RO through the host interface.
FWROWD	8	0b	RW1C	**Firmware Reset on WD indication** Set when a firmware reset is asserted due to watch dog expiration. Cleared when the host writes a 1b to it. Writing a 0b to this bit does not change its value. This bit is also set after LAN_PWR_GOOD, and is RO through the AUX interface.
FWRI	9	1b	RW1C	**Firmware Reset Indication** Set when a firmware reset is asserted. Cleared when the host writes a 1b to it. Writing a 0b to this bit does not change its value. This bit is also set after LAN_PWR_GOOD, and is RO through the AUX interface.
RESERVED	15:10	0x0	RSV	Reserved.

Field	Bit(s)	Init.	Type	Description
FWS1B	23:16	0x0	RO	**Firmware Status 1 Byte** This bit is RO through the host interface. 0x30 = STATUS_RECOVERY_MODE_CORER. 0x31 = STATUS_RECOVERY_MODE_GLOBR. 0x32 = STATUS_RECOVERY_MODE_TRANSITION. 0x33 = STATUS_RECOVERY_MODE_NVM. Others = Reserved.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.1.3 PF State - PFGEN_STATE (0x00088000; RO)

This register defines the main characteristics of the PF to software. It does not have any direct impact on device functionality.

Field	Bit(s)	Init.	Type	Description
RESERVED	0	0b	RSV	Reserved.
RESERVED	1	0b	RW	**Reserved.**
PFLINKEN	2	0b	RW	**PF Link Enable** 0b = The PF driver is disabled. The software device driver must at minimum report it does not have link. 1b = The PF driver is enabled.
PFSCEN	3	0b	RW	**PF iSCSI Enable** 0b = iSCSI should not be used for this function. 1b = iSCSI can be used for this function.
RESERVED	31:4	0x0	RSV	Reserved.

38.39.2.1.4 Global GPIO Control - GLGEN_GPIO_CTL[n] (0x00088100 + 0x4*n, n=0...29; RW)

General GPIO control registers. These registers control the mode of operation of general purpose I/O (GPIOn) pins. There are 30 GPIO pins that can be configured to different modes and assigned to different ports. There is one register per GPIOn pin. Register GLGEN_GPIO_CTL [n] controls GPIOn pin. This register is initialized only at LAN Power Good, preserving the GPIO states across software and PCIe resets.

Field	Bit(s)	Init.	Type	Description
PRT_NUM	1:0	00b	RW	**Port Number** Controls the port number associated with this GPIOn pin. This field is valid when the un-assigned port number bit (*PRT_NUM_NA*) is set to 0b. If a port number is not assigned, this GPIOn pin is a global resource not associated with any specific port.
RESERVED	2	0b	RSV	Reserved.
PRT_NUM_NA	3	0b	RW	**Port Number Not Assigned** Set to 1b if this GPIOn pin is not assigned to any port. The *PRT_NUM* field is ignored when this bit is set 1b (used by software/firmware only).
PIN_DIR	4	0b	RW	**Pin Direction** Controls whether this GPIOn pin is configured as an input or output: 0b = Input. 1b = Output. This bit is not affected by software or system reset, only by initial power-on or direct software writes. ***Note:*** When GPIO functionality is set to LED, the pin direction is set implicitly to output regardless of this field's value.

Field	Bit(s)	Init.	Type	Description
TRI_CTL	5	0b	RW	**Tristate Control** When this GPIOn pin is configured as an output, this field controls whether the pin is driven to tristate or driven high when *SDP_DATA* is set to 1b: 0b = Tristate. 1b = Driven high. The pin is driven to active low when the *SDP_DATA* is set to 0b. This bit is not affected by software or system reset, only by initial power-on or direct software writes. The *SDP_DATA* is set through the GLGEN_GPIO_SET register (Section 38.39.2.1.8). ***Note:*** This field is not used when GPIO is set to LED functionality.
OUT_CTL	6	0b	RW	**Output Control** Controls the output state during reset or D3 power state when no WoL or no manageability. 0b = The output transitions to tri-state during reset or D3 power state. 1b = The output is driven high during reset or D3 power state. This bit is ignored when this GPIO pin is configured as an input. This bit is not affected by software or system reset, only by initial power-on or direct software writes. ***Note:*** This field is not used when GPIO functionality is set to LED.
PIN_FUNC	9:7	000b	RW	**Pin Functionality** Controls the operation mode (or functionality) for this GPIOn pin. 000b = SDP (software definable input or output) 001b = LED (drives external LEDs) 010b = RoL (Reset on LAN) 011b = Timesync 0 100b = Timesync 1 All other values are reserved. When this pin is configure in SDP mode, the output value is set through GLGEN_GPIO_SET register (Section 38.39.2.1.8), and the value of the GPIO pin is read through GLGEN_GPIO_STAT register (Section 38.39.2.1.6).
LED_INVRT	10	0b	RW	**LED Invert** Specifies the polarity/inversion of the LED source prior to output or blink control. By default, the output drives the cathode of the LED so when the LED output is 0b the LED is on. 0b = LED output is active low. 1b = LED output is active high. The *LED_INVRT* setting is meaningful only if the *PIN_FUNC* is set to LED (001b).
LED_BLINK	11	0b	RW	**LED Blink** Specifies whether to apply blink logic to the (inverted) LED control source prior to the LED output. 0b = Do not blink LED output. 1b = Blink LED output. The *LED_BLINK* setting is meaningful only if the *PIN_FUNC* is set to LED (001b).
LED_MODE	15:12	0x0	RW	**LED Mode** Specifies the control source for the LED output. LED modes are described in detail in the LED section. The *LED_MODE* setting is meaningful only if the *PIN_FUNC* is set to LED (001b).
RESERVED	16	0b	RSV	Reserved.
INT_MODE	18:17	00b	RW	**Interrupt Mode** Selects the interrupt mode for this GPIOn pin. 00b = No interrupt. 01b = Interrupt on rising edge. 10b = Interrupt on falling edge. 11b = Interrupt on any transition. Set this field to generate interrupts only when *PIN_FUNC* is set to SDP (000b).

Field	Bit(s)	Init.	Type	Description
OUT_DEFAULT	19	0b	RW	**Output Default** Default value driven on this GPIOn pin when configured as a SDP output. 0b = Output driven low. 1b = Output driven high. The value is ignored when GPIO pin is configured as a SDP input or when *PIN_FUNC* is not SDP (000b).
PHY_PIN_NAME	25:20	0x0	RW	**PHY Pin Name** Used to indicate to firmware the name and functionality for which this GPIO pin is being used when connecting to an external PHY/module control/status pin. 0x3F = SDP is not used for PHY/module connectivity. 0x0 to 0x7 = Control/status pin names when connecting to an SFP+ module. 0x0 = Rx_LOS. 0x1 = Mod_ABS. 0x2 = RS0/RS1. 0x3 = TxDisable. 0x4 = TxFault. All other values are reserved. 0x8 to 0xF = Control status pin names when connecting to a QSFP+ module. 0x8 = IntL. 0x9 = ResetL. 0xA = ModPresL. 0xB = LPMode. 0xC = ModSelL. All other values are reserved. 0x10 to 0x17 = Control status pin names when connecting to 10GBASE-T PHYs. 0x10 = PhyInt. 0x11 = PhyRst. 0x12 = TxDisable. All other values are reserved.
RESERVED	31:26	0x0	RSV	Reserved.

38.39.2.1.5 Global LED Control - GLGEN_LED_CTL (0x00088178; RW)

Field	Bit(s)	Init.	Type	Description
GLOBAL_BLINK__MODE	0	0b	RW	**Global Blink Mode** Specifies the blink mode of all LEDs. 0b = Blink at 200 ms on and 200 ms off. 1b = Blink at 83 ms on and 83 ms off.
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.1.6 Global GPIO Status - GLGEN_GPIO_STAT (0x0008817C; RO)

Field	Bit(s)	Init.	Type	Description
GPIO_VALUE	29:0	0x0	RO	**GPIO Value** Each bit 'n' in this field reflects the value of GPIOn pin (either input or output). For example, Bit[0] reflects the value on GPIO0 pin (1b is high and 0b is low), Bit[1] indicates the value of GPIO1 pin, and so on.
RESERVED	31:30	00b	RSV	Reserved.

38.39.2.1.7 Global GPIO Transition Status - GLGEN_GPIO_TRANSIT (0x00088180; RW1C)

Field	Bit(s)	Init.	Type	Description
GPIO_TRANSITION	29:0	0x0	RW1C	**GPIO Transition** These register bits are used to latch any transition (low-to-high or high-to-low) on the GPIOn pins (either input or output) since the last time the register bits were cleared. Bit[0] reflects the transition status of GPIO0 pin, Bit[1] indicates the transition status of GPIO1 pin, and so on. The PF is expected to clear those flags that it cares about by writing 1b to the corresponding bit position. The PF is expected to clear the flags only for the GPIOn pins that it owns.
RESERVED	31:30	00b	RSV	Reserved.

38.39.2.1.8 Global GPIO Set - GLGEN_GPIO_SET (0x00088184; RW)

This register is used to set the value of the GPIOn output pins (0b = low, 1b = high). Attempting to write a value to GPIO input pins has no effect.

Field	Bit(s)	Init.	Type	Description
GPIO_INDX	4:0	0x0	RW	**GPIO Index** Defines the GPIO pin number that is driven to *SDP_DATA* value. The *GPIO_INDX* can be set to any value between 0 and 29 decimal (00000b to 11001b). Software is permitted to drive only GPIO pins that are defined as output SDPs by the GLGEN_GPIO_CTL registers (Section 38.39.2.1.4). Any PFs or EMP can access all GPIOs. It is the PF software responsibility to control only those GPIOs that the PF owns.
SDP_DATA	5	0b	RW	**SPD Data** The value in this field is driven to a GPIOn pin pointed to by the index *GPIO_INDX*. 0b = Driven low. 1b = Driven high.
DRIVE_SDP	6	0b	RW	**Drive SPD** Set this flag to 1b to drive the *SDP_DATA* to the GPIOn pin pointed by the index GPIO_INDX.
RESERVED	31:7	0x0	RSV	Reserved.

38.39.2.1.9 MDI Single Command and Address - GLGEN_MSCA[n] (0x0008818C + 0x4*n, n=0...3; RW)

This register is used for writing the command and address to initiate read/write access over the MDIO interface. It is used when the MDIOn_SDAn/MDCn_SCLn pins are configured for MDIO operation through the GLGEN_MDIO_I2C_SEL[n] register (Section 38.39.2.1.13).

Field	Bit(s)	Init.	Type	Description
MDIADD	15:0	0x0	RW	**MDI Address** Address used for new protocol MDI accesses (default = 0x0000).
DEVADD	20:16	0x0	RW	**Device Type/Reg Address** Five bits representing either device type if *STCODE* = 00b, or register address if *STCODE* = 01b.
PHYADD	25:21	0x0	RW	**PHY Address** The address of the external device.

Field	Bit(s)	Init.	Type	Description
OPCODE	27:26	00b	RW	**Op Code** Two bits identifying operation to be performed (default = 00b). 00b = Address cycle (new protocol only) 01b = Write operation 10b = Read, increment address (new protocol only) 11b = Read operation
STCODE	29:28	01b	RW	**ST Code** Two bits identifying start of frame and old or new protocol (default = 01b). 00b = New protocol 01b = Old protocol 1xb = Illegal
MDICMD	30	0b	RW1C	**MDI Command** Perform the MDI operation in this register. Cleared when done. 0b = MDI ready; operation complete (default). 1b = Perform operation; operation in progress.
MDIINPROGEN	31	0b	RW	**MDI in Progress Enable** Generate MDI in progress when operation completes. 0b = MDI ready disable (default). 1b = MDI in progress enable.

38.39.2.1.10 MDI Single Read and Write Data - GLGEN_MSRWD[n] (0x0008819C + 0x4*n, n=0...3; RW)

This register is used for reading and writing data to and from the MDIO interface. It is used when the MDIOn_SDAn/MDCn_SCLn pins are configured for MDIO operation through the GLGEN_MDIO_I2C_SEL[n] register (Section 38.39.2.1.13).

Field	Bit(s)	Init.	Type	Description
MDIWRDATA	15:0	0x0	RW	**MDI Write Data** For MDI writes to the external device.
MDIRDDATA	31:16	0x0	RO	**MDI Read Data** Read data from the external device.

38.39.2.1.11 I^2C Parameters - GLGEN_I2CPARAMS[n] (0x000881AC + 0x4*n, n=0...3; RW)

This register is used to set the parameters for I^2C access to the 2-wire management interface, and to enable bit banging access to the I^2C interface. It is used when the MDIOn_SDAn/MDCn_SCLn pins are configured for I^2C interface operation through the GLGEN_MDIO_I2C_SEL[n] register (Section 38.39.2.1.13).

Field	Bit(s)	Init.	Type	Description
WRITE_TIME	4:0	0x6	RW	**Write Time** Defines the delay between a write access and the next access. The value is in milliseconds. A value of zero is not valid.
READ_TIME	7:5	010b	RW	**Read Time** Defines the delay between a read access and the next access. The value is in milliseconds. A value of zero is not valid.
I2CBB_EN	8	0b	RW	**I^2C Bit Bang Enable** If set, the I2C_CLK and I2C_DATA lines are controlled via the *CLK*, *DATA* and *DATA_OE_N* fields of this register. Otherwise, they are controlled by the hardware machine activated via the GLGEN_I2CCMD[n] register (Section 38.39.2.1.15).

Field	Bit(s)	Init.	Type	Description
CLK	9	1b	RW	**I²C Clock out value** Used to drive the value of I2C_CLK (SCL output to PAD). While in bit bang mode, controls the value driven on the I²C clock pad MDCn_SCLn.
DATA_OUT	10	1b	RW	**I²C Data Out value** Used to drive the value of I²C data (SDA output to PAD). While in bit bang mode, and when the *DATA_OE_N* field is zero, controls the value driven on the I²C data pad MDIOn_SDAn.
DATA_OE_N	11	0b	RW	**I²C Data Output Enable** While in bit bang mode, controls the direction of the I2C_DATA pad MDIOn_SDAn. Pad is output. Pad is input.
DATA_IN	12	0b	RO	**I²C Data In value** Provides the value of I2C_DATA (SDA input from external PAD). This bit is RO. Reflects the value of the I2C_DATA pad MDIOn_SDAn. While in bit bang mode when the *DATA_OE_N* field is zero, this field reflects the value set in the *DATA_OUT* field.
CLK_OE_N	13	0b	RW	**I²C Clock Output Enable** While in bit bang mode, controls the direction of the I2C_CLK pad MDCn_SCLn. Pad is output. Pad is input.
CLK_IN	14	0b	RO	**I²C Clock In value** (SCL input from external PAD) This bit is RO. Reflects the value of the I²C CLK pad MDCn_SCLn. While in bit bang mode when the *CLK_OE_N* field is zero, this field reflects the value set in the *CLK_OUT* field.
CLK_STRETCH_DIS	15	0b	RW	**I²C Clock Stretch Disable** 0b = Enable slave clock stretching support in I²C access. 1b = Disable clock stretching support in I²C access.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.1.12 Global Device Timer - GLVFGEN_TIMER (0x000881BC; RW)

Field	Bit(s)	Init.	Type	Description
GTIME	31:0	0x0	RW	GTIME is a free running timer fed by a 1 us clock.

38.39.2.1.13 Global MDIO or I²C Select - GLGEN_MDIO_I2C_SEL[n] (0x000881C0 + 0x4*n, n=0...3; RW)

This register is used to select between the MDIO interface or I²C interface on the MDIOn_SDAn/MDCn_SCLn pins. Each pair of these pins can be independently configured for MDIO or I²C interface protocol by using GLGEN_MDIO_I2C_SEL[n] registers.

Field	Bit(s)	Init.	Type	Description
MDIO_I2C_SEL	0	0b	RW	**MDIO/I^{2}C Select** Used to select between the MDIO interface or I^{2}C interface over the MDIOn_SDAn/MDCn_SCLn pins, where n=0..3. 0b = MDIO interface is selected. 1b = I^{2}C interface is selected.
PHY_PORT_NUM	4:1	0x0	RW	**PHY Port Number** Each bit in this field indicates the port number of the external PHY/module accessed through this interface. For MDC/MDIO, one or more PHYs can be managed by the same pair of pins. Therefore, more than one bit can be set in this field. In I^{2}C mode, only a single bit can be set since only a single module might be connected. Bit[0] = Port 0 PHY/module. Bit[1] = Port 1 PHY/module. Bit[2] = Port 2 PHY/module. Bit[3] = Port 3 PHY/module.
PHY0_ADDRESS	9:5	0x0	RW	**PHY0 Address** If this interface is used for connecting to PHY0, this is the address of the PHY device to use. This field is used by firmware.
PHY1_ADDRESS	14:10	0x0	RW	**PHY1 Address** If this interface is used for connecting to PHY1, this is the address of the PHY device to use. This field is used by firmware.
PHY2_ADDRESS	19:15	0x0	RW	**PHY2 Address** If this interface is used for connecting to PHY2, this is the address of the PHY device to use. This field is used by firmware.
PHY3_ADDRESS	24:20	0x0	RW	**PHY3 Address** If this interface is used for connecting to PHY3, this is the address of the PHY device to use. This field is used by firmware.
MDIO_IF_MODE	28:25	0x0	RW	**MDIO IF Mode** When interface mode is set to MDC/MDIO, this field indicates to firmware the frame structure to use. This field contains a bit per port for scenarios where more than one PHY device is connected through a single interface. Each bit indicates the following options: 0b = Use IEEE Clause 45 Frame Structure. 1b = Use IEEE Clause 22 Frame Structure. where: Bit[0] = Port0 MDIO_IF_MODE. Bit[1] = Port1 MDIO_IF_MODE. Bit[2] = Port2 MDIO_IF_MODE. Bit[3] = Port3 MDIO_IF_MODE.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.1.14 MDIO Control - GLGEN_MDIO_CTRL[n] (0x000881D0 + 0x4*n, n=0...3; RW)

This register is used to control the MDC clock speed and mode when the MDIOn_SDAn/MDCn_SCLn pins are configured for use as MDIO interface. Each pair of these pins are independently configured as MDIO or I^{2}C interface through the GLGEN_MDIO_I2C_SEL[n] registers, where n=0..3 (Section 38.39.2.1.13).

Field	Bit(s)	Init.	Type	Description
LEGACY_RSVD2	16:0	0x2FFB	RW	Reserved. Do not modify this value.
CONTMDC	17	0b	RW	**Continuous MDC** Turn off MDC between MDIO packets. 0b = MDC off between packets (default). 1b = Continuous MDC.
LEGACY_RSVD1	31:18	0x400	RW	Reserved. Do not modify this value.

38.39.2.1.15 I²C command - GLGEN_I2CCMD[n] (0x000881E0 + 0x4*n, n=0...3; RW)

This register is used to read or write to the configuration registers over the I²C interface when the MDIOn_SDAn/MDCn_SCLn pins are configured for I²C operation. Each pair of these pins are independently configured as a MDIO or I²C interface through the GLGEN_MDIO_I2C_SEL[n] registers, where n=0..3 (Section 38.39.2.1.13).

Field	Bit(s)	Init.	Type	Description
DATA	15:0	0x0	RW	**I²C Data** For a write command, firmware/software places the data bits in this field and hardware shifts them out to the I²C bus. For a read command, hardware reads the bits serially from the I²C bus and places the data in this field so firmware/software can fetch the data from this location. This field is read in byte order and not in word order. Ordering can be changed by using GLGEN_I2CPARAMS.*I2C_DATA_ORDER* bit (Section 38.39.2.1.11).
REGADD	23:16	0x0	RW	**I²C Register Address** For example, register 0, 1, 2... 255.
PHYADD	26:24	000b	RW	**PHY Address** Device Address Bits[2:0]. The actual address used is 1010b (PHYADD[2:0]).
OP	27	0b	RW	**Op Code** 0b = I²C write. 1b = I²C read.
RESET	28	0b	RW1C	**Reset Sequence** If set, sends a reset sequence before the actual read or write. This bit is self clearing. A reset sequence is defined as nine consecutive stop conditions.
R	29	0b	RW	**Ready Bit** Indicates a read or write operation on the I²C interface has completed. Set to 1b by hardware at the end of the I²C transaction. Reset by a firmware/software write of a new command to this register.
RESERVED	30	0b	RSV	Reserved.
E	31	0b	RW	**Error** This bit set is to 1b by hardware when it fails to complete an I²C read. Reset by a firmware/software write of a new command. This bit is valid only when Ready bit *R* is set to 1b by hardware.

38.39.2.1.16 VM Reset Trigger - VSIGEN_RTRIG[VSI] (0x00090000 + 0x4*VSI, VSI=0...383; RW)

Field	Bit(s)	Init.	Type	Description
VMSWR	0	0b	RW	**VM Software Reset** VM software reset is initiated by setting the *VMSWR* bit. At completion of the reset flow, the PF software clears this bit.
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.1.17 VM Reset Status - VSIGEN_RSTAT[VSI] (0x00090800 + 0x4*VSI, VSI=0...383; RO)

Field	Bit(s)	Init.	Type	Description
VMRD	0	1b	RO	**VM Software Reset Done Indication** This flag is cleared when the VM reset is initiated (by setting the *VMSWR* flag in the matched VSIGEN_RTRIG register (Section 38.39.2.1.16), or any stronger reset that affects the VM). It is set back to 1b when hardware completes its VM reset sequence.
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.1.18 VF Reset Trigger - VPGEN_VFRTRIG[VF] (0x00091800 + 0x4*VF, VF=0...127; RW)

This register affects the VF, but is exposed only to the parent PF.

Field	Bit(s)	Init.	Type	Description
VFSWR	0	0b	RW	**VF Software Reset** VF software reset is done by the PF setting the *VFSWR* bit. At reset completion, the PF clears this bit.
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.1.19 VF Reset Status - VPGEN_VFRSTAT[VF] (0x00091C00 + 0x4*VF, VF=0...127; RO)

This register affects the VF, but is exposed only to the parent PF.

Field	Bit(s)	Init.	Type	Description
VFRD	0	1b	RO	**VF Software Reset Done Indication** This flag is cleared when the VF reset is initiated by setting the *VFSWR* flag in the matched VPGEN_VFRTRIG register (Section 38.39.2.1.18) or any stronger reset that affects the VF. It is set back to 1b when hardware completes its VF reset sequence.
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.1.20 PF Control - PFGEN_CTRL (0x00092400; RW)

Field	Bit(s)	Init.	Type	Description
PFSWR	0	0b	RW	**PF Software Reset** PF software reset is done by setting the *PFSWR* bit. At reset completion, hardware clears this bit.
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.1.21 PF Driver Unload - PFGEN_DRUN (0x00092500; RW)

Field	Bit(s)	Init.	Type	Description
DRVUNLD	0	0b	RW	**Driver Unload** The PF sets this bit to indicate its driver is unloading.
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.1.22 Global VF Level Reset Status - GLGEN_VFLRSTAT[n] (0x00092600 + 0x4*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
VFLRE	31:0	0x0	RW1C	**VFLR Status Indication** Bit 'm' in register 'n' reflects a VFLR event on VF index '32 x n + m', where 'n' is the register index and 'm' is the bit index.

38.39.2.1.23 Global Status - GLGEN_STAT (0x000B612C; RO)

Field	Bit(s)	Init.	Type	Description
RESERVED	1:0	00b	RSV	Reserved.
DCBEN	2	0b	RW	**Global DCB Enable** Defines if DCB is enabled. Loaded from NVM. Can be overridden by firmware. 0b = DCB disabled. 1b = DCB enabled.
VTEN	3	0b	RW	**Global VT Enable** Defines if the virtualization offloads are enabled. Loaded from NVM. Can be overridden by Firmware. 0b = VT disabled. 1b = VT enabled.
RESERVED	4	0b	RW	Reserved.
EVBEN	5	0b	RW	**Global EVB Enable** Defines if the PE structures (s-comp/M-comp) offloads are enabled. 0b = EVB disabled. 1b = EVB enabled.
RESERVED	31:6	0x0	RSV	Reserved.

38.39.2.1.24 General Port Status - PRTGEN_STATUS (0x000B8100; RO)

This register contains general port status information.

Field	Bit(s)	Init.	Type	Description
PORT_VALID	0	0b	RO	**Port Valid** Denotes if the respective Ethernet port is enabled. 0b = Port is disabled. 1b = Port is enabled.
PORT_ACTIVE	1	0b	RO	**Port Active** Denotes if the respective Ethernet port is active. 0b = Port is inactive and its respective PHY is in power down. 1b = Port is active and its respective PHY is powered up. *PORT_ACTIVE* is 1b when: *PORT_VALID* = 1b AND If (device is in D0 state) then port should be active: If the software device driver activated the link (*ACTIVATE_PORT_LINK* = 1b) or link is used for manageability. Else, when device is in Dr state, the port should be active+ If link is used for manageability or WoL.
RESERVED	31:2	0x0	RSV	Reserved.

38.39.2.1.25 General Port Configuration - PRTGEN_CNF (0x000B8120; RO)

This register contains configuration per Ethernet port loaded from the NVM.

Field	Bit(s)	Init.	Type	Description
PORT_DIS	0	1b	RW	**Port Disable** Defines if the Ethernet port is enabled from the NVM. Exception: Port 0 is always enabled and cannot be disabled from the NVM. 0b = Enabled. 1b = Disabled. Default: 0b = Port 0. 1b = Other ports.
ALLOW_PORT_DIS	1	0b	RW	**Allow Port Disable** 0b = Asserting LAN_DIS_N has no effect on this port. 1b = Asserting LAN_DIS_N disables this port.
EMP_PORT_DIS	2	0b	RW	**EMP Port Disable** Set by the EMP to disable the Ethernet port. The NVM value for this bit should always be 0b (enabled). NVM should use the *PORT_DIS* bit to disable a port.
RESERVED	31:3	0x0	RSV	Reserved.

38.39.2.1.26 General Port Configuration2 - PRTGEN_CNF2 (0x000B8160; RO)

This register contains configuration per Ethernet port loaded from the NVM.

Field	Bit(s)	Init.	Type	Description
ACTIVATE_PORT_LINK	0	1b	RW	**Activate Port Link** When this field is set to 0b, the port's link is powered down. This field can be used by an application to disable the link until the software device driver is loaded and enables the link. ***Notes:*** 1. The PCIe functions associated with the port are not affected by the link loss. 2. Deactivating the link using this configuration is ignored when the interface is used for manageability. To implement this, hardware masks this configuration when the relevant port's PRTPM_GC.*EMP_LINK_ON* is set to 1b (Section 38.39.2.4.1). 3. If APM or ACPI power management are used, this bit should be set.
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.1.27 Global Reset Delay - GLGEN_RSTCTL (0x000B8180; RO)

Field	Bit(s)	Init.	Type	Description
GRSTDEL	5:0	0x8	RW	**Global Reset Delay** Global/Core and EMP resets delay defined in 100 ms units. Setting *GRSTDEL* to zero bypasses the delay counter.
RESERVED	7:6	00b	RSV	Reserved.
ECC_RST_ENA	8	0b	RW	**ECC Reset Enable** Graceful GLOBR reset on ECC in any memory other than the EMP memories. 0b = A detected ECC error on these memories generates only an interrupt to the PFs. 1b = A detected ECC error generated a graceful GLOBR.
RESERVED	31:9	0x0	RSV	Reserved.

38.39.2.1.28 Global Clock Status - GLGEN_CLKSTAT (0x000B8184; RO)

Field	Bit(s)	Init.	Type	Description
CLKMODE	0	0b	RW	**Clock Mode** Controls the operating mode of internal clocks: 0b = Performance mode — Device operates at its highest internal frequencies, independent of the Ethernet link speed. 1b = Power mode — Device adjusts its internal frequencies to match the speed of the Ethernet link.
RESERVED	3:1	000b	RSV	Reserved.
U_CLK_SPEED	5:4	00b	RO	**Upper Clock Speed** Debug Feature. Represents the current speed of the upper clock (core_clk) as follows: 00b = 390.625 MHz. 01b = 195.3125 MHz. 10b = 97.65625 MHz. 11b = Reserved.
RESERVED	7:6	00b	RSV	Reserved.
P0_CLK_SPEED	10:8	000b	RO	**P0 Clock Speed** Debug Feature. Represents the current speed of the Rx clock for MAC 0. 3 bits per port (synchronized to the equivalent MAC clock). Speeds are represented as follows: 000b = Reserved. 001b = 1 GbE. 010b = 10 GbE. 011b = Reserved. All other values are reserved.
RESERVED	11	0b	RSV	Reserved.
P1_CLK_SPEED	14:12	000b	RO	**P1 Clock Speed** Debug Feature. Represents the current speed of the Rx clock for MAC 1. 3 bits per port (synchronized to the equivalent MAC clock). Speeds are represented as follows: 000b = Reserved. 001b = 1 GbE. 010b = 10 GbE. 011b = Reserved. All other values are reserved.
RESERVED	15	0b	RSV	Reserved.
P2_CLK_SPEED	18:16	000b	RO	**P2 Clock Speed** Debug Feature. Represents the current speed of the Rx clock for MAC 2. 3 bits per port (synchronized to the equivalent MAC clock). Speeds are represented as follows: 000b = Reserved. 001b = 1 GbE. 010b = 10 GbE. 011b = Reserved. All other values are reserved.
RESERVED	19	0b	RSV	Reserved.
P3_CLK_SPEED	22:20	000b	RO	**P3 Clock Speed** Debug Feature. Represents the current speed of the Rx clock for MAC 3. 3 bits per port (synchronized to the equivalent MAC clock). Speeds are represented as follows: 000b = Reserved. 001b = 1 GbE. 010b = 10 GbE. 011b = Reserved. All other values are reserved.
RESERVED	31:23	0x0	RSV	Reserved.

38.39.2.1.29 Global Reset Status - GLGEN_RSTAT (0x000B8188; RO)

Field	Bit(s)	Init.	Type	Description
DEVSTATE	1:0	00b	RO	**Device State** Device can be at one of the following states: 00b = Device active. 01b = Reset requested. 10b = Reset in progress. 11b = Reserved.
RESET_TYPE	3:2	00b	RO	**Reset Type** Reflects one of the following resets that are/were in progress: 00b = POR. 01b = CORER. 10b = GLOBR. 11b = EMPR.
CORERCNT	5:4	00b	RO	**Core Reset Count** Counts the number of initiated core resets since POR. The counter wraps around from 11b to 00b.
GLOBRCNT	7:6	00b	RO	**Global Reset Count** Counts the number of initiated global resets since POR. The counter wraps around from 11b to 00b.
EMPRCNT	9:8	00b	RO	**EMP Reset Count** Counts the number of initiated EMP resets since POR. The counter wraps around from 11b to 00b.
TIME_TO_RST	15:10	0x0	RO	**Time to Reset** The reset time is a down counter, loaded by GLRSTDEL following a GLOBR or CORER or EMPR. When GLOBRTIME reaches a zero value, the actual reset is initiated.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.1.30 Global Reset Trigger - GLGEN_RTRIG (0x000B8190; RW)

Field	Bit(s)	Init.	Type	Description
CORER	0	0b	RW	**Core Reset** Setting the CORER triggers a graceful core reset flow.
GLOBR	1	0b	RW	**Global Reset** Setting the GLOBR triggers a graceful global reset flow.
RESERVED	31:2	0x0	RSV	Reserved.

38.39.2.1.31 LAN Port Number - PFGEN_PORTNUM (0x001C0480; RO)

Field	Bit(s)	Init.	Type	Description
PORT_NUM	1:0	00b	RW	**Port Number** Indicates the LAN port connected to this function. 00b = Port 0. 01b = Port 1. 10b = Port 2. 11b = Port 3.
RESERVED	31:2	0x0	RSV	Reserved.

38.39.2.1.32 PCI Function Count - GLGEN_PCIFCNCNT (0x001C0AB4; RO)

Field	Bit(s)	Init.	Type	Description
PCIPFCNT	4:0	0x1	RW	**PCI PF Count** Reports the function number of the highest PCI physical function plus 1 as it is loaded from the NVM. For example, if the NVM enables two functions 0 and 7, the value is 8 (7+1). Used to partition the interrupt vectors among supported PFs.
RESERVED	15:5	0x0	RSV	Reserved.
PCIVFCNT	23:16	0x0	RW	**PCI VF Count** Reports the function number of highest PCI virtual function plus 1 for all PFs combined, as it is loaded from the NVM. Used to partition the interrupt vectors among supported VFs.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.2 PF - PCIe Registers

This category contains registers for PCIe configuration and control.

38.39.2.2.1 Function Requester ID Information Register - PF_FUNC_RID (0x0009C000; RO)

Field	Bit(s)	Init.	Type	Description
FUNCTION_NUMBER	2:0	000b	RO	**Function Number** Function number assigned to the function based on BIOS/OS enumeration.
DEVICE_NUMBER	7:3	0x0	RO	**Device Number** No-ARI mode: Device number assigned to the function based on BIOS/OS enumeration. ARI mode: Upper 5 bits of the 8-bit function number.
BUS_NUMBER	15:8	0x0	RO	**Bus Number** Bus number assigned to the function based on BIOS/OS enumeration.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.2.2 PCIe Configuration Indirect Access Address - PF_PCI_CIAA (0x0009C080; RW)

Field	Bit(s)	Init.	Type	Description
ADDRESS	11:0	0x0	RW	**Address** The configuration space address to access.
VF_NUM	18:12	0x0	RW	**VF Number** Defines the VF number to access. The VF number is the absolute VF number in the device.
RESERVED	31:19	0x0	RSV	Reserved. Ignore on read. Write 0b.

38.39.2.2.3 PCIe Configuration Indirect Access Data - PF_PCI_CIAD (0x0009C100; RW)

Field	Bit(s)	Init.	Type	Description
DATA	31:0	0x0	RW	**Data** Used to access the configuration registers of the VF. Operates together with the PF_PCI_CIAA register (Section 38.39.2.2.2) as follows: Reading this register returns the content of the register at offset = ADDRESS in the configuration space of VF index = VF_NUM (the ADDRESS and VF_NUM parameters are defined by the PF_PCI_CIAA register).

38.39.2.2.4 Function Active and Power State - PFPCI_FACTPS (0x0009C180; RO)

This register is provided for internal firmware use, notifying firmware about function state.

Field	Bit(s)	Init.	Type	Description
FUNC_POWER_STATE	1:0	00b	RO	**Function Power State** Power state indication of the function. 00b = Dr. 01b = D3. 10b = D0a. 11b = D0u.
RESERVED	2	0b	RSV	Reserved.
FUNC_AUX_EN	3	0b	RO	**Function Aux Enable** Reflects the *Auxiliary Power PM Enable* bit from the PCI configuration space.
RESERVED	31:4	0x0	RSV	Reserved.

38.39.2.2.5 PCIe VM Pending Index - PFPCI_VMINDEX (0x0009C300; RW)

Field	Bit(s)	Init.	Type	Description
VMINDEX	8:0	0x0	RW	**VM Index** Software sets the VMINDEX that its transaction pending flag should be reflected in the PFPCI_VMPEND register (Section 38.39.2.2.6). The VM index is an absolute index in the range of 0 through 383. It can only be set by software to VMs that the PF owns and only to VMs that are not assigned to a VF.
RESERVED	31:9	0x0	RSV	Reserved.

38.39.2.2.6 PCIe VM Pending Status - PFPCI_VMPEND (0x0009C380; RO)

Field	Bit(s)	Init.	Type	Description
PENDING	0	0b	RO	**PCIe Transaction Pending Status** The reported VM is controlled by the *VMINDEX* field in the PFPCI_VMINDEX register (Section 38.39.2.2.5). This flag is set to 1b as long as there is at least one PCIe transaction pending for its completion.
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.2.7 PCIe Default Revision ID - GLPCI_DREVID (0x0009C480; RO)

Field	Bit(s)	Init.	Type	Description
DEFAULT_REVID	7:0	0x0	RO	**Default Revision ID** Mirroring of default Rev ID prior to an NVM load.
RESERVED	31:8	0x0	RSV	Reserved.

38.39.2.2.8 PCIe Statistic Control Register #1 - GLPCI_GSCL_1 (0x0009C48C; RW)

This register controls the operation of the PCIe performance counters.

Field	Bit(s)	Init.	Type	Description
GIO_COUNT_EN_0	0	0b	RW	**GIO Counter Enable 0** Enables PCIe statistic counter number 0.
GIO_COUNT_EN_1	1	0b	RW	**GIO Counter Enable 1** Enables PCIe statistic counter number 1.
GIO_COUNT_EN_2	2	0b	RW	**GIO Counter Enable 2** Enables PCIe statistic counter number 2.
GIO_COUNT_EN_3	3	0b	RW	**GIO Counter Enable 3** Enables PCIe statistic counter number 3.
LBC_ENABLE_0	4	0b	RW	**Leaky Bucket Counter Enable 0** 0b = Leaky bucket mode is disabled and the counter is incremented by one for each event. 1b = Statistics counter 0 operates in leaky bucket mode.
LBC_ENABLE_1	5	0b	RW	**Leaky Bucket Counter Enable 1** 0b = Leaky bucket mode is disabled and the counter is incremented by one for each event. 1b = Statistics counter 1 operates in leaky bucket mode.
LBC_ENABLE_2	6	0b	RW	**Leaky Bucket Counter Enable 2** 0b = Leaky bucket mode is disabled and the counter is incremented by one for each event. 1b = Statistics counter 2 operates in leaky bucket mode.
LBC_ENABLE_3	7	0b	RW	**Leaky Bucket Counter Enable 3** 0b = Leaky bucket mode is disabled and the counter is incremented by one for each event. 1b = Statistics counter 3 operates in leaky bucket mode.
RESERVED	27:8	0x0	RSV	Reserved.
GIO_64_BIT_EN	28	0b	RW	**GIO 64-Bit Enable** Enables two 64-bit counters instead of four 32-bit counters.
GIO_COUNT_RESET	29	0b	RW1S	**GIO Counter Reset** Reset indication of PCIe statistic counters. Reading this bit returns a 0b.
GIO_COUNT_STOP	30	0b	RW1S	**GIO Counter Stop** Stop indication of PCIe statistic counters. Reading this bit returns a 0b.
GIO_COUNT_START	31	0b	RW1S	**GIO Counter Start** Start indication of PCIe statistic counters. Reading this bit returns a 0b.

38.39.2.2.9 PCIe Statistic Control Register #2 - GLPCI_GSCL_2 (0x0009C490; RW)

This register defines the events counted by the performance counters.

Field	Bit(s)	Init.	Type	Description
GIO_EVENT_NUM_0	7:0	0x0	RW	**GIO Event Number 0** Event number that counter 0 counts (GSCN_0).
GIO_EVENT_NUM_1	15:8	0x0	RW	**GIO Event Number 1** Event number that counter 1 counts (GSCN_1).
GIO_EVENT_NUM_2	23:16	0x0	RW	**GIO Event Number 2** Event number that counter 2 counts (GSCN_2).
GIO_EVENT_NUM_3	31:24	0x0	RW	**GIO Event Number 3** Event number that counter 3 counts (GSCN_3).

38.39.2.2.10 PCIe Statistic Control Registers #5...#8 - GLPCI_GSCL_5_8[n] (0x0009C494 + 0x4*n, n=0...3; RW)

These registers control the operation of the leaky bucket counter n.

GSCL_5 corresponds to n=0
GSCL_6 corresponds to n=1
GSCL_7 corresponds to n=2
GSCL_8 corresponds to n=3

Field	Bit(s)	Init.	Type	Description
LBC_THRESHOLD_N	15:0	0x0	RW	**Leaky Bucket Counter Threshold N** Threshold for the leaky bucket counter n.
LBC_TIMER_N	31:16	0x0	RW	**Leaky Bucket Counter Timer N** Time period between decrementing the value in leaky bucket counter n. The time period is defined in µs units.

38.39.2.2.11 PCIe Statistic Counter Registers #0...#3 - GLPCI_GSCN_0_3[n] (0x0009C4A4 + 0x4*n, n=0...3; RO)

These registers contain the performance counters 0-3.

GSCL_0 corresponds to n=0
GSCL_1 corresponds to n=1
GSCL_2 corresponds to n=2
GSCL_3 corresponds to n=3

Field	Bit(s)	Init.	Type	Description
EVENT_COUNTER	31:0	0x0	RO	**Event Counter** A 32-bit event counter. These registers are stuck at their maximum value of 0xFF...F. Refer to the section on Performance and Statistics Counters.

38.39.2.2.12 PCIe PQs Max Used Space - GLPCI_PQ_MAX_USED_SPC (0x0009C4EC; RO)

Field	Bit(s)	Init.	Type	Description
GLPCI_PQ_MAX_USED_SPC_12	7:0	0x0	RO	**PCIe PQ Max Used Space 12**
GLPCI_PQ_MAX_USED_SPC_13	15:8	0x0	RO	**PCIe PQ Max Used Space 13**
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.2.13 PCIe Mux Selector for PFB - GLPCI_PM_MUX_PFB (0x0009C4F0; RW)

Field	Bit(s)	Init.	Type	Description
PFB_PORT_SEL	4:0	0x0	RW	**PFB Port Select**
RSVD_0	15:5	0x0	RSV	Reserved.
INNER_PORT_SEL	18:16	000b	RW	**Inner Port Select**
RSVD_1	31:19	0x0	RSV	Reserved.

38.39.2.2.14 PCIe Mux Selector for NPQs - GLPCI_PM_MUX_NPQ (0x0009C4F4; RW)

Field	Bit(s)	Init.	Type	Description
NPQ_NUM_PORT_SEL	2:0	000b	RW	**NPQ Number Port Select**
RSVD_0	15:3	0x0	RSV	Reserved.
INNER_NPQ_SEL	20:16	0x0	RW	**Inner NPQ Select**
RSVD_1	31:21	0x0	RSV	Reserved.

38.39.2.2.15 PCIe Regs Spare Bits 0 - GLPCI_SPARE_BITS_0 (0x0009C4F8; RW)

Field	Bit(s)	Init.	Type	Description
SPARE_BITS	31:0	0x0	RW	**Spare Bits**

38.39.2.2.16 PCIe Regs Spare Bits 1 - GLPCI_SPARE_BITS_1 (0x0009C4FC; RW)

Field	Bit(s)	Init.	Type	Description
SPARE_BITS	31:0	0x0	RW	**Spare Bits**

38.39.2.2.17 PCIe NPQ Current Status of Allowed Resources for RLAN - GLPCI_CUR_RLAN_ALWD (0x0009C500; RO)

This register reports the current value of resources that a client might use.

Field	Bit(s)	Init.	Type	Description
DATA_LINES	15:0	0x1BE	RO	**Data Lines** Amount of data lines available.
OSR	31:16	0x40	RO	**Outstanding Requests** Amount of outstanding requests available.

38.39.2.2.18 PCIe NPQ Current Status of Allowed Resources for TLAN - GLPCI_CUR_TLAN_ALWD (0x0009C504; RO)

This register reports the current value of resources that a client might use.

Field	Bit(s)	Init.	Type	Description
DATA_LINES	15:0	0x1BE	RO	**Data Lines** Amount of data lines available.
OSR	31:16	0x11	RO	**Outstanding Requests** Amount of outstanding requests available.

38.39.2.2.19 PCIe NPQ Current Status of Allowed Resources for RXPE - GLPCI_CUR_RXPE_ALWD (0x0009C508; RO)

This register reports the current value of resources that a client might use.

Field	Bit(s)	Init.	Type	Description
DATA_LINES	15:0	0xE9	RO	**Data Lines** Amount of data lines available.
OSR	31:16	0x18	RO	**Outstanding Requests** Amount of outstanding requests available.

38.39.2.2.20 PCIe NPQ Statistic Control Register #0 - GLPCI_GSCL_0 (0x0009CA8C; RO)

This register controls the operation of the PCIe performance counters.

Field	Bit(s)	Init.	Type	Description
RESERVED	28:0	0x0	RSV	Reserved.
GIO_COUNT_RESET	29	0b	RW1S	Reset indication of PCIe statistic counters. Reading this bit returns a 0b.
GIO_COUNT_STOP	30	0b	RW1S	Stop indication of PCIe statistic counters. Reading this bit returns a 0b.
GIO_COUNT_START	31	0b	RW1S	Start indication of PCIe statistic counters. Reading this bit returns a 0b.

38.39.2.2.21 PCIe NPQ Current Status of Allowed Resources for TXPE - GLPCI_CUR_TXPE_ALWD (0x0009C50C; RO)

This register reports the current value of resources that a client might use.

Field	Bit(s)	Init.	Type	Description
DATA_LINES	15:0	0xE9	RO	**Data Lines** Amount of data lines available.
OSR	31:16	0x18	RO	**Outstanding Requests** Amount of outstanding requests available.

38.39.2.2.22 PCIe NPQ Current Status of Allowed Resources for PMAT - GLPCI_CUR_PMAT_ALWD (0x0009C510; RO)

This register reports the current value of resources that a client might use.

Field	Bit(s)	Init.	Type	Description
DATA_LINES	15:0	0x4C0	RO	**Data Lines** Amount of data lines available.
OSR	31:16	0xBC	RO	**Outstanding Requests** Amount of outstanding requests available.

38.39.2.2.23 PCIe NPQ Current Status of Allowed Resources for MNG - GLPCI_CUR_MNG_ALWD (0x0009C514; RO)

This register reports the current value of resources that a client might use.

Field	Bit(s)	Init.	Type	Description
DATA_LINES	15:0	0x1B3	RO	**Data Lines** Amount of data lines available.
OSR	31:16	0x9	RO	**Outstanding Requests** Amount of outstanding requests available.

38.39.2.2.24 PCIe NPQ Current Status of Allowed Resources for TDPU - GLPCI_CUR_TDPU_ALWD (0x0009C518; RO)

This register reports the current value of resources that a client might use.

Field	Bit(s)	Init.	Type	Description
DATA_LINES	15:0	0x21F	RO	**Data Lines** Amount of data lines available.
OSR	31:16	0x50	RO	**Outstanding Requests** Amount of outstanding requests available.

38.39.2.2.25 PCIe NPQ Current Status of Reserved Resources for RLAN - GLPCI_CUR_RLAN_RSVD (0x0009C580; RO)

This register reports the current value of resources that are reserved for clients.

Field	Bit(s)	Init.	Type	Description
DATA_LINES	15:0	0x0	RO	**Data Lines** Amount of data lines available.
OSR	31:16	0x0	RO	**Outstanding Requests** Amount of outstanding requests available.

38.39.2.2.26 PCIe NPQ Current Status of Reserved Resources for TLAN - GLPCI_CUR_TLAN_RSVD (0x0009C584; RO)

This register reports the current value of resources that are reserved for clients.

Field	Bit(s)	Init.	Type	Description
DATA_LINES	15:0	0x0	RO	**Data Lines** Amount of data lines available.
OSR	31:16	0x0	RO	**Outstanding Requests** Amount of outstanding requests available.

38.39.2.2.27 PCIe NPQ Current Status of Reserved Resources for RXPE - GLPCI_CUR_RXPE_RSVD (0x0009C588; RO)

This register reports the current value of resources that are reserved for clients.

Field	Bit(s)	Init.	Type	Description
DATA_LINES	15:0	0x0	RO	**Data Lines** Amount of data lines available.
OSR	31:16	0x0	RO	**Outstanding Requests** Amount of outstanding requests available.

38.39.2.2.28 PCIe NPQ Current Status of Reserved Resources for TXPE - GLPCI_CUR_TXPE_RSVD (0x0009C58C; RO)

This register reports the current value of resources that are reserved for clients.

Field	Bit(s)	Init.	Type	Description
DATA_LINES	15:0	0x0	RO	**Data Lines** Amount of data lines available.
OSR	31:16	0x0	RO	**Outstanding Requests** Amount of outstanding requests available.

38.39.2.2.29 PCIe NPQ Current Status of Reserved Resources for PMAT - GLPCI_CUR_PMAT_RSVD (0x0009C590; RO)

This register reports the current value of resources that are reserved for clients.

Field	Bit(s)	Init.	Type	Description
DATA_LINES	15:0	0x0	RO	**Data Lines** Amount of data lines available.
OSR	31:16	0x0	RO	**Outstanding Requests** Amount of outstanding requests available.

38.39.2.2.30 PCIe NPQ Current Status of Reserved Resources for MNG - GLPCI_CUR_MNG_RSVD (0x0009C594; RO)

This register reports the current value of resources that are reserved for clients.

Field	Bit(s)	Init.	Type	Description
DATA_LINES	15:0	0x0	RO	**Data Lines** Amount of data lines available.
OSR	31:16	0x0	RO	**Outstanding Requests** Amount of outstanding requests available.

38.39.2.2.31 PCIe NPQ Current Status of Reserved Resources for TDPU - GLPCI_CUR_TDPU_RSVD (0x0009C598; RO)

This register reports the current value of resources that are reserved for clients.

Field	Bit(s)	Init.	Type	Description
DATA_LINES	15:0	0x0	RO	**Data Lines** Amount of data lines available.
OSR	31:16	0x0	RO	**Outstanding Requests** Amount of outstanding requests available.

38.39.2.2.32 PCIe VF Flush Done - PFPCI_VF_FLUSH_DONE[VF] (0x0009C600 + 0x4*VF, VF=0...127; RO)

Field	Bit(s)	Init.	Type	Description
FLUSH_DONE	0	0b	RO	**Flush Done**
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.2.33 PCIe PF Flush Done - PFPCI_PF_FLUSH_DONE (0x0009C800; RO)

Field	Bit(s)	Init.	Type	Description
FLUSH_DONE	0	0b	RO	**Flush Done**
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.2.34 PCIe VM Flush Done - PFPCI_VM_FLUSH_DONE (0x0009C880; RO)

Field	Bit(s)	Init.	Type	Description
FLUSH_DONE	0	0b	RO	**Flush Done**
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.2.35 PCIe NPQ Config - GLPCI_NPQ_CFG (0x0009CA00; RW)

This register controls some parameters of NPQ.

Field	Bit(s)	Init.	Type	Description
EXTEND_TO	0	0b	RW	**Extend Timeout** Extends timeout in the permitted range.
SMALL_TO	1	0b	RW	**Small Timeout** Reduce timeout value for simulation
WEIGHT_AVG	5:2	0x2	RW	**Weight Average** Controls the weight of the average filter for average round trip calculation.
NPQ_SPARE	15:6	0x0	RW	**NPQ Spare Bits**
NPQ_ERR_STAT	19:16	0x0	RO	**NPQ Error Stats**
RESERVED	31:20	0x0	RSV	Reserved.

38.39.2.2.36 PCIe NPQ Current Status of Common Bucket - GLPCI_CUR_CLNT_COMMON (0x0009CA18; RO)

This register reports the current value of common resources in NPQ.

Field	Bit(s)	Init.	Type	Description
DATA_LINES	15:0	0x53B	RO	**Data Lines** Amount of data lines available.
OSR	31:16	0xD0	RO	**Outstanding Requests** Amount of outstanding requests available.

38.39.2.2.37 PCIe NPQ Current Status of Pipe Monitor - GLPCI_CUR_CLNT_PIPEMON (0x0009CA20; RO)

This register reports the current value of pipe monitor in NPQ.

Field	Bit(s)	Init.	Type	Description
DATA_LINES	15:0	0x53B	RO	**Data Lines** Amount of data lines available.
RESERVED	31:16	0x0	RO	Reserved.

38.39.2.2.38 PCIe NPQ Watermark of Common Bucket - GLPCI_CUR_WATMK_CLNT_COMMON (0x0009CA28; RO)

This register reports a watermark of the common resources in NPQ.

Field	Bit(s)	Init.	Type	Description
DATA_LINES	15:0	0x53B	RO	**Data Lines** Amount of data lines available.
OSR	31:16	0x0	RO	**Outstanding Requests** Amount of outstanding requests available.

38.39.2.2.39 PCIe NPQ Watermark of Pipe Monitor - GLPCI_WATMK_CLNT_PIPEMON (0x0009CA30; RO)

This register reports the current value of the pipe monitor in NPQ.

Field	Bit(s)	Init.	Type	Description
DATA_LINES	15:0	0x53B	RO	**Data Lines** Amount of data lines available.
RESERVED	31:16	0x0	RO	Reserved.

38.39.2.2.40 PCIe NPQ Watermark of Allowed Resources for RLAN - GLPCI_WATMK_RLAN_ALWD (0x0009CB00; RO)

This register reports the watermark of resources that client are allowed to use.

Field	Bit(s)	Init.	Type	Description
DATA_LINES	15:0	0x1BE	RO	**Data Lines** Amount of data lines available.
OSR	31:16	0x40	RO	**Outstanding Requests** Amount of outstanding requests available.

38.39.2.2.41 PCIe NPQ Watermark of Allowed Resources for TLAN - GLPCI_WATMK_TLAN_ALWD (0x0009CB04; RO)

This register reports the watermark of resources that clients are allowed to use.

Field	Bit(s)	Init.	Type	Description
DATA_LINES	15:0	0x1BE	RO	**Data Lines** Amount of data lines available.
OSR	31:16	0x11	RO	**Outstanding Requests** Amount of outstanding requests available.

38.39.2.2.42 PCIe NPQ Watermark of Allowed Resources for RXPE - GLPCI_WATMK_RXPE_ALWD (0x0009CB08; RO)

This register reports the watermark of resources that client are allowed to use.

Field	Bit(s)	Init.	Type	Description
DATA_LINES	15:0	0xE9	RO	**Data Lines** Amount of data lines available.
OSR	31:16	0x18	RO	**Outstanding Requests** Amount of outstanding requests available.

38.39.2.2.43 PCIe NPQ Watermark of Allowed Resources for TXPE - GLPCI_WATMK_TXPE_ALWD (0x0009CB0C; RO)

This register reports the watermark of resources that client are allowed to use.

Field	Bit(s)	Init.	Type	Description
DATA_LINES	15:0	0xE9	RO	**Data Lines** Amount of data lines available.
OSR	31:16	0x18	RO	**Outstanding Requests** Amount of outstanding requests available.

38.39.2.2.44 PCIe NPQ Watermark of Allowed Resources for PMAT - GLPCI_WATMK_PMAT_ALWD (0x0009CB10; RO)

This register reports the watermark of resources that clients are allowed to use.

Field	Bit(s)	Init.	Type	Description
DATA_LINES	15:0	0x4C0	RO	**Data Lines** Amount of data lines available.
OSR	31:16	0xBC	RO	**Outstanding Requests** Amount of outstanding requests available.

38.39.2.2.45 PCIe NPQ Watermark of Allowed Resources for MNG - GLPCI_WATMK_MNG_ALWD (0x0009CB14; RO)

This register reports the watermark of resources that clients are allowed to use.

Field	Bit(s)	Init.	Type	Description
DATA_LINES	15:0	0x1B3	RO	**Data Lines** Amount of data lines available.
OSR	31:16	0x9	RO	**Outstanding Requests** Amount of outstanding requests available.

38.39.2.2.46 PCIe NPQ Watermark of Allowed Resources for TDPU - GLPCI_WATMK_TPDU_ALWD (0x0009CB18; RO)

This register reports the watermark of resources that clients are allowed to use.

Field	Bit(s)	Init.	Type	Description
DATA_LINES	15:0	0x21F	RO	**Data Lines** Amount of data lines available.
OSR	31:16	0x50	RO	**Outstanding Requests** Amount of outstanding requests available.

38.39.2.2.47 PCIe PF Configuration - PFPCI_CNF (0x000BE000; RO)

Contains per-PF configuration loaded from the NVM.

Field	Bit(s)	Init.	Type	Description
RESERVED	1:0	00b	RSV	Reserved.
MSI_EN	2	1b	RW	**MSI Enable** Enables the MSI capability structure for this PCI function. 0b = MSI is disabled. 1b = MSI is enabled.
EXROM_DIS	3	0b	RW	**Expansion ROM Disable** 0b = The Expansion ROM BAR in the PCI configuration space is enabled. 1b = The Expansion ROM BAR in the PCI configuration space is disabled.
IO_BAR	4	0b	RW	**I/O BAR Support** 0b = I/O BAR is not supported. 1b = I/O BAR is supported.
INT_PIN	6:5	00b	RW	**Interrupt Pin** Controls the value advertised in the Interrupt Pin field of the PCI configuration header for this function. 00b = INTA# 01b = INTB# 10b = INTC# 11b = INTD# The value advertised in the PCI configuration header is the value loaded from the NVM plus one.
RESERVED	31:7	0x0	RSV	Reserved.

38.39.2.2.48 PCIe PF Device ID - PFPCI_DEVID (0x000BE080; RO)

Contains the per-PF device ID.

Field	Bit(s)	Init.	Type	Description
PF_DEV_ID	15:0	0x374C	RW	**PF Device ID** Contains the device ID for this PF.
VF_DEV_ID	31:16	0x374D	RW	**VF Device ID** Contains the device ID for the VFs of this PF.

38.39.2.2.49 PCIe Functions Configuration 2 - PFPCI_FUNC2 (0x000BE180; RO)

Field	Bit(s)	Init.	Type	Description
RESERVED	31:0	0x0	RSV	Reserved.

38.39.2.2.50 PCIe Functions Configuration - PFPCI_FUNC (0x000BE200; RO)

Field	Bit(s)	Init.	Type	Description
FUNC_DIS	0	1b	RW	**Function Disable** Defines if the PCI function is enabled from the NVM. Exception: This bit is RO for PF0. It is always enabled and cannot be disabled from the NVM. 0b = Enabled 1b = Disabled Default: 0b = PF0. 1b = Other functions.
ALLOW_FUNC_DIS	1	0b	RW	**Allow Function Disable** 0b = Asserting PCI_DIS_N has no effect on this PCI function. 1b = Asserting PCI_DIS_N disables this PCI function.
DIS_FUNC_ON_PORT_DIS	2	0b	RW	**Disable Function On Port Disable** Defines whether this PF is disabled when the LAN_DIS_N pin is asserted. 0b = Asserting LAN_DIS_N has no effect on this PCI function. 1b = Asserting LAN_DIS_N disables this PCI function.
RESERVED	31:3	0x0	RSV	Reserved.

38.39.2.2.51 PCIe Function Status 1 - PFPCI_STATUS1 (0x000BE280; RO)

Field	Bit(s)	Init.	Type	Description
FUNC_VALID	0	0b	RO	**Function Valid** 0b = Function is disabled. 1b = Function is enabled. This bit is valid to firmware even when the function is disabled.
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.2.52 PCIe PM - PFPCI_PM (0x000BE300; RW)

Field	Bit(s)	Init.	Type	Description
PME_EN	0	0b	RW	**PME Enable** Used by the software device driver to generate a PME event without writing to the PMCSR in the PCIe configuration space. The internal PME Enablement is a logic OR function of the following: *PME_EN* flag in the PMCSR, *PME_EN* flag in the PFPCI_PM CSR, and *APME* flag in the PFPM_APM CSR. The bit is reset on Sticky Reset (STRST). Bit is reset only on power-on reset (LAN_PWR_GOOD). When *AUX_PWR* = 0b, this bit is also reset when de-asserting PE_RST_N.
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.2.53 PCIe Storage Class - PFPCI_CLASS (0x000BE400; RO)

Contains per-PF configuration loaded from the NVM.

Field	Bit(s)	Init.	Type	Description
STORAGE_CLASS	0	0b	RW	**Storage Class** 0b = The class code of this function is set to 0x020000 (LAN). 1b = The class code of this function is set to 0x010000 (SCSI).
RESERVED	1	0b	RSV	Reserved.
PF_IS_LAN	2	0b	RW	**PF Is LAN** 0b = Reserved. 1b = LAN function.
RESERVED	31:3	0x0	RSV	Reserved.

38.39.2.2.54 PCI BAR Control - GLPCI_LBARCTRL (0x000BE484; RO)

Field	Bit(s)	Init.	Type	Description
PREFBAR	0	1b	RW	**Prefetchable BAR** Prefetchable bit indication in the memory BAR and MSI-X BAR (should be set when 64-bit BARs are used). 0b = BARs are marked as non prefetchable. 1b = BARs are marked as prefetchable.
BAR32	1	0b	RW	**BAR 32-bit Enable** 0b = 64-bit BAR addressing mode is selected. 1b = 32-bit BARs are enabled.
RESERVED	2	0b	RSV	Reserved.
FLASH_EXPOSE	3	1b	RW	**Flash Exposed** When set, the Flash memory is accessible through the memory BAR.
PE_DB_SIZE	5:4	00b	RW	**PE DB Size** Determines the size of the memory space allocated to the protocol engine door bells in the PF BARs 00b = Memory space is not allocated for PE door bells. 01b = A 64 KB area is allocated. 10b = A (4 MB + 64 KB) area is allocated. 11b = Reserved.
FL_SIZE	8:6	111b	RW	**Flash Size** Indicates the size of the external Flash as = 64 KB x (2 ** FL_SIZE). 101b = 2 MB. 110b = 4 MB. 111b = 8 MB. All other values are reserved.
RESERVED	10:9	00b	RSV	Reserved.
EXROM_SIZE	13:11	011b	RW	**Expansion ROM Size** Indicates the size of the expansion ROM BAR as = 64 KB x (2 ** EXROM_SIZE). 000b = 64 KB. 001b = 128 KB. 010b = 256 KB. 011b = 512 KB. 100b = 1 MB. 101b = 2 MB. 110b = 4 MB. 111b = 8 MB. Default value is 512 KB.
RESERVED	31:14	0x0	RSV	Reserved.

38.39.2.2.55 PCIe Subsystem ID - GLPCI_SUBVENID (0x000BE48C; RO)

Field	Bit(s)	Init.	Type	Description
SUB_VEN_ID	15:0	0x8086	RW	**Subsystem Vendor ID** Loaded to the PCI configuration Subsystem Vendor ID register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.2.56 PCIe Power Data Register - GLPCI_PWRDATA (0x000BE490; RO)

Field	Bit(s)	Init.	Type	Description
D0_POWER	7:0	0x0	RW	**D0 Power** The value in this field is reflected in the PCI Power Management Data register of the LAN functions for D0 power consumption and dissipation (Data_Select = 0 or 4).
COMM_POWER	15:8	0x0	RW	**Common Power** The value in this field is reflected in the PCI Power Management Data register of function 0 when the Data_Select field is set to eight (common function).
D3_POWER	23:16	0x0	RW	**D3 Power** The value in this field is reflected in the PCI Power Management Data register of the LAN functions for D3 power consumption and dissipation (Data_Select = 3 or 7).
DATA_SCALE	25:24	00b	RW	**Data Scale** The value in this field reflects the Data_Scale field in the PCI PMCSR register.
RESERVED	31:26	0x0	RSV	Reserved.

38.39.2.2.57 PCIe Global Config 2 - GLPCI_CNF2 (0x000BE494; RO)

This register contains global status fields of PCIe configuration.

Field	Bit(s)	Init.	Type	Description
RO_DIS	0	0b	RW	**Relaxed Ordering Disable** 0b = Relaxed ordering is specified per request type. 1b = The device does not request any relaxed ordering transactions.
CACHELINE_SIZE	1	0b	RW	**Cache Line Size** Determines the system cache line size. 0b = 64 bytes. 1b = 128 bytes. This field is loaded from the NVM.
MSI_X_PF_N	12:2	0x80	RW	**MSI_X PF Table Size** System software reads this field to determine the MSI-X table size N, which is encoded as N-1. This field is loaded from the NVM *MSI_X_N_PF* field and reflects the same field in the PCIe MSI-X configuration (Message Control register).
MSI_X_VF_N	23:13	0x04	RW	**MSI_X VF Table Size** System software reads this field to determine the MSI-X table size N for VFs, which is encoded as N-1. This field is loaded from the NVM *MSI_X_N_VF* field and reflects the same field in the PCIe MSI-X configuration (VF MSI-X Control register).
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.2.58 PCIe Serial Number MAC Address Low - GLPCI_SERL (0x000BE498; RO)

Field	Bit(s)	Init.	Type	Description
SER_NUM_L	31:0	0x0	RW	**Serial Number Low** The low DWord of the Ethernet MAC address used to generate the PCIe serial number. The register contents is loaded from NVM. The location in NVM is pointed by the 4th item in the Auto-Generated Pointers Module. It is a per device manufacturing value which represents the whole device. It can be set identical to the concatenated [PFPM_SAL1|PFPM_SAL0] words of the EMP Settings Module.

38.39.2.2.59 PCIe Serial Number MAC Address High - GLPCI_SERH (0x000BE49C; RO)

Field	Bit(s)	Init.	Type	Description
SER_NUM_H	15:0	0x0	RW	**Serial Number High** The high word of the Ethernet MAC address used to generate the PCIe serial number. The register contents is loaded from the NVM. The location in NVM is pointed by the 5th item in the auto-generated pointers module. It is a per device manufacturing value which represents the entire device. It can be set identical to the concatenated [PFPM_SAH1|PFPM_SAH0] words of the EMP settings module.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.2.60 PCIe Capabilities Control - GLPCI_CAPCTRL (0x000BE4A4; RW)

Determines PCIe capabilities supported by the device, and that software is allowed to enable or disable.

Field	Bit(s)	Init.	Type	Description
VPD_EN	0	0b	RW	**VPD Enable** 0b = The PCIe VPD capability is not present and is not exposed. 1b = The PCIe VPD capability is present and exposed.
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.2.61 PCIe Capabilities Support - GLPCI_CAPSUP (0x000BE4A8; RO)

Determines PCIe capabilities supported by the device.

Field	Bit(s)	Init.	Type	Description
PCIE_VER	0	1b	RW	**PCIE Capability Version** Determines the PCIe capability version. 0b = Capability version: 0x1. 1b = Capability version: 0x2.
RESERVED	2:1	0b	RSV	Reserved.
TPH_EN	3	1b	RW	**TPH Enable** A value of 1b indicates support for the PCIe TPH requester capability.
ARI_EN	4	1b	RW	**ARI Enable** A value of 1b indicates support for PCIe ARI capability.
IOV_EN	5	1b	RW	**IOV Enable** A value of 1b indicates support for PCIe SR-IOV capability.
ACS_EN	6	1b	RW	**ACS Enable** A value of 1b indicates support for PCIe ACS capability.
SEC_EN	7	1b	RW	**Secondary Enable** A value of 1b indicates support for the secondary PCIe extended capability.

Field	Bit(s)	Init.	Type	Description
RESERVED	15:8	0x0	RSV	Reserved.
ECRC_GEN_EN	16	1b	RW	**ECRC Generation Enable** Loaded into the ECRC. Generation capable bit of the PCIe configuration registers.
ECRC_CHK_EN	17	1b	RW	**ECRC Check Enable** Loaded into the ECRC. Check capable bit of the PCIe configuration registers.
IDO_EN	18	1b	RW	**IDO Enable** Enables ID-based ordering (IDO).
MSI_MASK	19	1b	RW	**MSI Per-vector Masking Setting** Loaded to the masking bit (bit 8) in the message control of the MSI configuration capability structure.
CSR_CONF_EN	20	1b	RW	**CSR Configuration Enable** Enables access to CSRs via the PCI configuration space. Refer to the section on Configuration Access to Internal Registers.
WAKUP_EN	21	0b	RW	**Wake-up Enable** If set, wake-up in D3 is exposed in *PME_Support* field in the PMCR.
RESERVED	29:22	0x0	RSV	Reserved.
LOAD_SUBSYS_ID	30	0b	RW	**Load Subsystem IDs** When set to 1b, indicates that the device loads its PCIe sub-system ID and sub-system vendor ID from the NVM.
LOAD_DEV_ID	31	0b	RW	**Load Device ID** When set to 1b, indicates that the device loads its PCI device IDs from the NVM.

38.39.2.2.62 PCIe Link Capabilities - GLPCI_LINKCAP (0x000BE4AC; RO)

Determines PCIe Link capabilities supported by the device.

Field	Bit(s)	Init.	Type	Description
LINK_SPEEDS_VECTOR	5:0	0x0	RW	**Supported Link Speeds Vector** Loaded to the Link Capabilities 2 register in the PCIe capability values: Bit[0] = 5.0 GT/s. Bit[1] = 8.0 GT/s. Bits[5:2] = Reserved. 2.5 GT/s is always supported.
MAX_PAYLOAD	8:6	100b	RW	**Max Payload Size Supported** Loaded to the PCIe Device Capabilities register. Support values are: 000b = 128 bytes. 001b = 256 bytes. 010b = 512 bytes. 001b = 1 KB. 100b = 2 KB. Otherwise, keep the hardware default.
MAX_LINK_WIDTH	12:9	0x07	RW	**Max Link Width** Loaded to the PCIe Link Capabilities register. 0001b = Limit max link width to x1. 0011b = Limit max link width to x4. 0100b = Limit max link width to x8. 0111b = Do not limit max link width. Negotiate to the max width supported by the link. All other values are reserved.
RESERVED	31:13	0x0	RSV	Reserved.

38.39.2.2.63 PCIe PM Support - GLPCI_PMSUP (0x000BE4B0; RO)

This register contains parameters that define PCIe power management support.

Field	Bit(s)	Init.	Type	Description
RESERVED	7:0	0x0	RSV	Reserved.
L0S_ACC_LAT	10:8	011b	RW	**L0s Acceptable Latency** Loaded to the Endpoint L0s Acceptable Latency field in the PCIe Device Capabilities register.
L1_ACC_LAT	13:11	110b	RW	**L1 Acceptable Latency** Loaded to the Endpoint L1 Acceptable Latency field in the PCIe Device Capabilities register.
RESERVED	14	0b	RSV	Reserved.
OBFF_SUP	16:15	00b	RW	**OBFF Supported** Loaded to the OBFF Supported field in the PCIe Device Capabilities 2 register. Must be set to 0b in the NVM (OBFF is not supported).
RESERVED	31:17	0x0	RSV	Reserved.

38.39.2.2.64 PCIe Revision ID - GLPCI_REVID (0x000BE4B4; RO)

This register is shared by all physical functions.

Field	Bit(s)	Init.	Type	Description
NVM_REVID	7:0	0x0	RW	**NVM Revision ID** Value of the Rev ID loaded from the NVM.
RESERVED	31:8	0x0	RSV	Reserved.

38.39.2.2.65 PCIe VF Capabilities Support - GLPCI_VFSUP (0x000BE4B8; RO

Field	Bit(s)	Init.	Type	Description
VF_PREFETCH	0	1b	RW	**VF Prefetchable** 0b = IOV memory BAR and MSI-X BAR are declared as non-prefetchable. 1b = IOV memory BAR and MSI-X BAR are declared as prefetchable.
VR_BAR_TYPE	1	1b	RW	**VF BAR Type** 0b = VF BARs advertise 32-bit size. 1b = VF BARs advertise 64-bit size.
RESERVED	31:2	0x0	RSV	Reserved.

38.39.2.2.66 PCIe Global Config - GLPCI_CNF (0x000BE4C0; RO)

Field	Bit(s)	Init.	Type	Description
RESERVED	1:0	00b	RSV	Reserved.
WAKE_PIN_EN	2	0b	RW	**Wake Pin Enable** When set to 1b, enables the use of the WAKE pin for a PME event in all power states.
RESERVED	31:3	0x0	RSV	Reserved.

38.39.2.2.67 PCIe Upper Address - GLPCI_UPADD (0x000BE4F8; RW)

This register is used to block PCIe master accesses above some address.

Field	Bit(s)	Init.	Type	Description
RESERVED	0	0b	RSV	Reserved.
ADDRESS	31:1	0x0	RW	**Address** Bits[31:1] correspond to Bits[63:33] in the PCIe address space, respectively.

38.39.2.2.68 PCIe Vendor ID - GLPCI_VENDORID (0x000BE518; RO)

Field	Bit(s)	Init.	Type	Description
VENDORID	15:0	0x8086	RW	**Vendor ID** Contains the vendor ID exposed in offset 0x0 in the config space of all functions. A value of 0xFFFF is ignored.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.3 PF - MAC Registers

38.39.2.3.1 Port MAC Address Low - PRTGL_SAL (0x001E2120; RO)

This register contains the NVM-loaded lower bits of the 48-bit per-PF Ethernet MAC address. All 32 bits are valid.

Field	Bit(s)	Init.	Type	Description
FC_SAL	31:0	0x0	RW	**Station Address Low** Used for flow control packet processing. The lower 32 bits of the 48-bit Ethernet MAC address. This field is defined in big endian (LS byte of SAL is first on the wire).

38.39.2.3.2 Port MAC Address High - PRTGL_SAH (0x001E2140; RO)

This register contains the NVM-loaded upper bits of the 48-bit per-PF Ethernet MAC address. All 32 bits are valid. The complete address is (PFPM_SAH, PFPM_SAL). PFPM_SAH.*AV* determines whether this address is valid.

Field	Bit(s)	Init.	Type	Description
FC_SAH	15:0	0x0	RW	**Station Address High** Used for flow control packet processing. The upper 16 bits of the 48-bit Ethernet MAC address. This field is defined in big endian (MS byte of SAH is last on the wire).
MFS	31:16	0x2600	RW	**Maximum Frame Size** Defines the maximum receive frame size in bytes from MAC addresses up to and including the CRC. Frames received that are larger than this value might be dropped based on the SBP configuration. This configuration has no effect on maximum transmit frame size.

38.39.2.3.3 Link Down Counter - PRTMAC_LINK_DOWN_COUNTER (0x001E2440; RW1C)

Field	Bit(s)	Init.	Type	Description
LINK_DOWN_COUNTER	15:0	0x0	RW1C	**Link Down Counter** Increments on link down event. Does not do wrap around.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.4 PF - Power Management Registers

38.39.2.4.1 General Control - PRTPM_GC (0x000B8140; RO)

Field	Bit(s)	Init.	Type	Description
EMP_LINK_ON	0	0b	RW	**EMP Link On** 0b = This Ethernet port is not required for EMP functionality. 1b = This Ethernet port is required to be up for EMP functionality.
MNG_VETO	1	0b	RW	**Manageability Veto** Manageability Veto for link LPLU and reset. Access is read/write by manageability, read only to the host. Impact on LPLU: 0b = No specific constraints on link from manageability. 1b = Hold off any low-power link mode changes. This is done to avoid link loss and interrupting manageability activity. Impact on reset: Reset impact is global for the device and is received as a logical OR between the per port bits. When at least one of the per port bits is set, PCI reset triggers internal CORER which does not impact the MAC and PHY interfaces. When all bits are cleared, PCI reset triggers internal GLOBR which initialize also the MAC and PHY interfaces.
RATD	2	0b	RW	**Restart Auto-negotiation on Transition to Dx** Enables the functionality to restart KX/KR backplane auto-negotiation on transitions to Dx(Dr/D3). 0b = Does not restart auto-negotiation when all port's functions moves to the Dx state. 1b = Restarts auto-negotiation to reach a low-power link mode (1 GbE link) when all port's functions transitions to the Dx state.
LCDMP	3	0b	RW	**Lowest Common Denominator (LCD) on Dx(Dr/D3) without Main Power** 0b = No specific action. 1b = Move to lowest common denominator link speed when main power is removed. When *RATD* bit is also set to 1b, it causes the link mode to auto-negotiate to the lowest common denominator (if enabled) when the main power (MAIN_PWR_OK) is removed.
RESERVED	30:4	0x0	RSV	Reserved.
LPLU_ASSERTED	31	0b	RW	**LPLU Asserted** Reflects the LPLU status of the port

38.39.2.5 PF - Wake Up and Proxying Registers

38.39.2.5.1 Wake Up Control Register - PFPM_WUC (0x0006B200; RW)

The *PME_En* and *PME_Status* bits of this register are reset when LAN_PWR_GOOD is 0b. When AUX_PWR = 0b, these register bits also reset by de-asserting PE_RST_N and during a D3 to D0 transition.

Note: For each PF only the WUC register placed in the first WoL and Proxying register set is valid. The values programmed to the WUC registers in higher numbered WoL and Proxying register sets is ignored.

Field	Bit(s)	Init.	Type	Description
RESERVED	4:0	0x0	RSV	Reserved.
EN_APM_D0	5	0b	RW	**Enable APM Wake Also On D0** 0b = Enable wake only when function is in D3/Dr. 1b = Enable wake also in D0.
RESERVED	31:6	0x0	RSV	Reserved.

38.39.2.5.2 Wake Up Filter Control Register - PFPM_WUFC (0x0006B400; RW)

This register is used to enable each of the pre-defined and flexible filters for wake-up support. A value of 1b means the filter is turned on, a value of 0b means the filter is turned off.

Field	Bit(s)	Init.	Type	Description
LNKC	0	0b	RW	**Link Status Change Wake-up Enable**
MAG	1	0b	RW	**Magic Packet Wake-up Enable**
RESERVED	2	0b	RSV	Reserved.
MNG	3	0b	RW	**Manageability** Manageability wake up enable.
RESERVED	30:4	0x0	RSV	Reserved.
FW_RST_WK	31	0b	RW	**Enable Wake On Firmware Reset Assertion** When set, a firmware reset causes system wake so that the software device driver can re-send proxying information to firmware.

38.39.2.5.3 Wake Up Status Register - PFPM_WUS (0x0006B600; RW1C)

This register is used to record statistics about all wake-up packets received. If a packet matches multiple criteria, multiple bits could be set. Writing a 1b to any bit clears that bit. This register is not cleared when RST# is asserted. It is only cleared when LAN_PWR_GOOD is de-asserted, or when cleared by the software device driver.

Note: If additional packets are received that match one of the wake-up filters, after the original wake-up packet is received, the WUS register is not updated with the new match detection until the register is cleared.

Field	Bit(s)	Init.	Type	Description
LNKC	0	0b	RW1C	**Link status Changed**
MAG	1	0b	RW1C	**Magic Packet Received**
PME_STATUS	2	0b	RW1C	**PME Status** This bit is set when device receives a wake-up event. It is the same as the *PME_Status* bit in the PMCSR. Writing a 1b to this bit also clears the *PME_STATUS* bit in the PMCSR. This bit is reset only on power-on reset (LAN_PWR_GOOD). When AUX_PWR = 0b, this bit is also reset on de-assertion of PE_RST_N.
MNG	3	0b	RW1C	**Manageability** Manageability wake-up status.
RESERVED	30:4	0x0	RSV	Reserved.

Field	Bit(s)	Init.	Type	Description
FW_RST_WK	31	0b	RW1C	**Wake Due To Firmware Reset Assertion Event** When set to 1b, indicates that firmware reset assertion caused system wake so that the software device driver can re-send proxying information to firmware.

38.39.2.5.4 WU on MNG Control - GLPM_WUMC (0x0006C800; RO)

Field	Bit(s)	Init.	Type	Description
RESERVED	15:0	0x0	RW	Reserved.
MNG_WU_PF	31:16	0x0	RW	**Manageability Wake Up Per PF** *MNG_WU_PF* EMP can set a bit in this field to indicate a management-initiated wake-up event (bit per PF).

38.39.2.5.5 APM Control Register - PFPM_APM (0x000B8080; RW)

Field	Bit(s)	Init.	Type	Description
APME	0	0b	RW	**Advance Power Management Enable** If set to 1b, APM wake up is enabled. Bit is reset on Power-on reset (LAN_PWR_GOOD) only.
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.5.6 Station Address Low - PRTPM_SAL[n] (0x001E4440 + 0x20*n, n=0...3; RO)

This register contains the lower bits of the NVM pre-loaded 48-bit Ethernet MAC address. All 32 bits are valid.

Field	Bit(s)	Init.	Type	Description
PFPM_SAL	31:0	0x0	RW	**Station Address Low** The lower 32 bits of the 48-bit NVM pre-assigned Ethernet MAC address. Field is defined in big endian (LS byte of SAL is first on the wire).

38.39.2.5.7 Station Address High - PRTPM_SAH[n] (0x001E44C0 + 0x20*n, n=0...3; RO)

This register contains the upper bits of the NVM pre-loaded 48-bit Ethernet MAC address. The complete address is (PFPM_SAH, PFPM_SAL). PFPM_SAH.*AV* determines whether this address is valid and compared against the incoming packet. After reset, if the NVM is present, the station address is loaded from the NVM, and its *Address Valid* field is 1b.

Field	Bit(s)	Init.	Type	Description
PFPM_SAH	15:0	0x0	RW	**Station Address High** The upper 16 bits of the 48-bit Ethernet MAC address. Field is defined in big Indian (MS byte of PRTPM_SAH is Last on the wire).
RESERVED	25:16	0x0	RSV	Reserved.
PF_NUM	29:26	0x0	RW	**PF Number** PF number to be used for reporting the waking PF. Value is written by firmware.
MC_MAG_EN	30	0b	RW	**Multicast Magic Packet Enable** Enable promiscuous multicast for magic packets. If this bit is set to 1b, every multicast magic packet generates a WoL event if enabled in PFPM_WUFC.*MAG* (Section 38.39.2.5.2).

Field	Bit(s)	Init.	Type	Description
AV	31	0b	RW	**Address Valid** If the NVM is present, the station address is assigned by firmware after loading from the NVM, and its *Address Valid* field is set to 1b.

38.39.2.6 PF - NVM Registers

38.39.2.6.1 Unit Load Status - GLNVM_ULD (0x000B6008; RO)

This register provides indications on the completion of loading the Shadow RAM and Alternate Module into the device units.

Field	Bit(s)	Init.	Type	Description
PCIER_DONE	0	0b	RW	**PCIe Reset Done** The PCIe reset process is done (all related registers are loaded).
PCIER_DONE_1	1	0b	RW	**PCIe Reset Done 1** The PCIe reset process is done (all related registers are loaded). Mirror of Bit[0].
RESERVED	2	1b	RSV	Reserved.
CORER_DONE	3	0b	RW	**Core Reset Done** The core reset process is done (all related registers are loaded).
GLOBR_DONE	4	0b	RW	**Global Reset Done** The global reset process is done (all related registers are loaded).
POR_DONE	5	0b	RW	**Power On Reset Done** The power-on reset process is done (all related registers are loaded).
RESERVED	7:6	11b	RSV	Reserved.
POR_DONE_1	8	0b	RW	**Power On Reset Done 1** The power-on reset process is done (all related registers are loaded). Mirror of Bit[5].
PCIER_DONE_2	9	0b	RW	**PCIe Reset Done 2** The PCIe reset process is done (all related registers are loaded). Mirror of Bit[0].
PE_DONE	10	0b	RW	**PE Reset Done** The PE core reset process is done (all related registers are loaded).
RESERVED2	31:11	0x0	RSV	Reserved.

38.39.2.6.2 Protected CSR list - GLNVM_PROTCSR[n] (0x000B6010 + 0x4*n, n=0...59; RO)

Field	Bit(s)	Init.	Type	Description
ADDR_BLOCK	23:0	0xFFFFFF	RW	**CSR Blocked Address** Contains the address of a blocked register included in the CSR protected list NVM module. Blocked registers cannot be loaded from a CSR format module (type 1/2/3). The register is loaded from shadow RAM at POR events only, and only from the CSR protected list module in the NVM. It can be written by EMP. It can also be written by a host only when in blank Flash programming mode.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.6.3 Global NVM General Status Register - GLNVM_GENS (0x000B6100; RO)

This register cannot be loaded from the NVM via one of the CSR format modules.

Field	Bit(s)	Init.	Type	Description
NVM_PRES	0	0b	RO	**NVM Present** Setting this bit to 1b indicates that a Flash part is present and that a correct validity field was found in one of the two basic banks (such as a validity field value read is 01b).
RESERVED	4:1	0x0	RSV	Reserved.
SR_SIZE	7:5	110b	RO	**Shadow RAM Size** Defines the size of the internal shadow RAM. This is equal to the size of the internal shadow RAM in power of 2 KB units. Initial value is 110b = 64 KB.
BANK1VAL	8	0b	RW	**Basic Bank 1 Valid** Indicates that the content of basic bank 0 is valid. Indicates that the content of the basic bank 1 of the Flash device is valid. Meaningful only when *NVM_PRES* bit is read as 1b. It is written by hardware once at power-up, and then toggled only by EMP.
RESERVED	31:9	0x0	RSV	Reserved.

38.39.2.6.4 Flash ID Register - GLNVM_FLASHID (0x000B6104; RO)

Field	Bit(s)	Init.	Type	Description
FLASHID	23:0	0x0	RO	**Flash ID** It is formed by the 3-bytes JEDEC-ID of the device. • Byte 0 (Bits[7:0]) is the first byte read from the Flash after the read JEDEC-ID Op Code (0x9F) is issued to it. It contains the manufacturer's ID. • Byte 1 (Bits[15:8]) is the second byte read from the flash after the read JEDEC-ID opcode (0x9F) is issued to it. It contains the memory type. • Byte 2 (Bits[23:16]) is the third byte read from the flash after the read JEDEC-ID opcode (0x9F) is issued to it. It contains the memory capacity.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.6.5 Flash Access Register - GLNVM_FLA (0x000B6108; RO)

This register is writable by the host only when the device is in the blank flash programming mode. It cannot be loaded from the NVM via one of the CSR format modules.

Field	Bit(s)	Init.	Type	Description
RESERVED	5:0	0x0	RSV	Reserved.
LOCKED	6	1b	RW	**Locked** Normal NVM programming mode. 0b = The device is in blank Flash programming mode. 1b = The device is in normal NVM programming mode. The bit can be cleared by EMP or by hardware.
RSV1	31:7	0x0	RSV	Reserved.

38.39.2.6.6 Auto-load Timers - GLNVM_ALTIMERS (0x000B6140; RO)

Field	Bit(s)	Init.	Type	Description
PCI_ALTIMER	11:0	0x010	RW	**PCIe Auto-Load Timer** Default is 16 to 15 ms. resolution is in ms.
GEN_ALTIMER	31:12	0x0012C	RW	**General Auto-Load Timer** Default is ~ 300 ms. resolution is in ms.

38.39.2.6.7 Unit Load Timeout - GLNVM_ULT (0x000B6154; RO)

This register cannot be loaded from the NVM via one of the CSR format modules.

Field	Bit(s)	Init.	Type	Description
CONF_PCIR_AE	0	0b	RW	When set, indicates that the PCIe auto-load timer has ended before the respective NVM module has been initialized.
CONF_PCIRTL_AE	1	0b	RW	When set, indicates that the PCIe auto-load timer has ended before the respective NVM module has been initialized.
RESERVED	2	0b	RSV	Reserved.
CONF_CORE_AE	3	0b	RW	When set, indicates that the general auto-load timer has ended before the respective NVM module has been initialized.
CONF_GLOBAL_AE	4	0b	RW	When set, indicates that the general auto-load timer has ended before the respective NVM module has been initialized.
CONF_POR_AE	5	0b	RW	When set, indicates that the general auto-load timer has ended before the respective NVM module has been initialized.
RESERVED	7:6	0b	RW	Reserved.
CONF_EMP_AE	8	0b	RW	When set, indicates that the general auto-load timer has ended before the respective NVM module has been initialized.
CONF_PCIALT_AE	9	0b	RW	When set, indicates that the PCIe auto-load timer has ended before the respective NVM module has been initialized.
RESERVED_4	31:10	0x0	RW	Reserved.

38.39.2.6.8 MEM_INIT_DONE_STAT - MEM_INIT_DONE_STAT (0x000B615C; RO)

Each bit in this register is connected to the corresponding signal indication from the block, indicating memory initialization after reset is done.

Field	Bit(s)	Init.	Type	Description
CMLAN_MEM_INIT_DONE	0	0b	RO	**CMLAN Memory Initialization Done** CMLAN memory initialization done indication (from MNG.V pin).
PMAT_MEM_INIT_DONE	1	0b	RO	**PMAT Memory Initialization Done** PMAT memory initialization done indication (from MNG.V pin).
RCU_MEM_INIT_DONE	2	0b	RO	**RCU Memory Initialization Done** RCU memory initialization done indication (from MNG.V pin).
TDPU_MEM_INIT_DONE	3	0b	RO	**TDPU Memory Initialization Done** TDPU memory initialization done indication (from MNG.V pin).
TLAN_MEM_INIT_DONE	4	0b	RO	**TLAN Memory Initialization Done** TLAN memory initialization done indication (from MNG.V pin).
RLAN_MEM_INIT_DONE	5	0b	RO	**RLAN Memory Initialization Done** RLAN memory initialization done indication (from MNG.V pin).
RDPU_MEM_INIT_DONE	6	0b	RO	**RDPU Memory Initialization Done** RDPU memory initialization done indication (from MNG.V pin).

Field	Bit(s)	Init.	Type	Description
PPRS_MEM_INIT_DONE	7	0b	RO	**PPRS Memory Initialization Done** PPRS memory initialization done indication (from MNG.V pin).
RPB_MEM_INIT_DONE	8	0b	RO	**RPB Memory Initialization Done** RPB memory initialization done indication (from MNG.V pin).
TPB_MEM_INIT_DONE	9	0b	RO	**TUPM Memory Initialization Done** TUPM memory initialization done indication (from MNG.V pin).
FOC_MEM_INIT_DONE	10	0b	RO	**FOC Memory Initialization Done** FOC memory initialization done indication (from MNG.V pin).
TSCD_MEM_INIT_DONE	11	0b	RO	**TSCD Memory Initialization Done** TSCD memory initialization done indication (from MNG.V pin).
TCB_MEM_INIT_DONE	12	0b	RO	**TCB Memory Initialization Done** TCB memory initialization done indication (from MNG.V pin).
RCB_MEM_INIT_DONE	13	0b	RO	**RCB Memory Initialization Done** RCB memory initialization done indication (from MNG.V pin).
WUC_MEM_INIT_DONE	14	0b	RO	**WUC Memory Initialization Done** WUC memory initialization done indication (from MNG.V pin).
STAT_MEM_INIT_DONE	15	0b	RO	**STAT Memory Initialization Done** STAT memory initialization done indication (from MNG.V pin).
ITR_MEM_INIT_DONE	16	0b	RO	**ITR Memory Initialization Done** ITR memory initialization done indication (from MNG.V pin).
RESERVED	31:17	0x0	RSV	Reserved.

38.39.2.6.9 GLNVM_AL_REQ - GLNVM_AL_REQ (0x000B6164; RO)

Auto-load requests.

Field	Bit(s)	Init.	Type	Description
POR	0	0b	RW	Reset indication.
PCIE_IMIB	1	0b	RW	Reset indication.
GLOBR	2	0b	RW	Reset indication.
CORER	3	0b	RW	Reset indication.
PE	4	0b	RW	Reset indication.
PCIE_IMIB_ASSERT	5	0b	RW	Reset indication.
RESERVED	31:6	0x0	RSV	Reserved.

38.39.2.7 PF - Analyzer Registers

38.39.2.7.1 L2 Tag - Enable - PRT_L2TAGSEN[PRT] (0x001C0B20 + 0x4*PRT, PRT=0...3; RW)

Field	Bit(s)	Init.	Type	Description
ENABLE	7:0	0x0	RW	**Enable** Defines the L2 tags expected on this port.
RESERVED	31:8	0x0	RSV	Reserved.

38.39.2.8 PF - Switch Registers

38.39.2.8.1 Switching Table Default Action Enable Bitmap - GL_SWR_DEF_ACT_EN[n] (0x0026CFB8 + 0x4*n, n=0...1; RO)

Switching table default action enable bitmap corresponding to GL_SWR_DEF_ACT (Section 38.39.2.8.2).

Field	Bit(s)	Init.	Type	Description
DEF_ACT_EN_BITMAP	31:0	0x0	RW	**Default Action Enable Bitmap** Switching table default action enabling bitmap.

38.39.2.8.2 Switching Table Default Action - GL_SWR_DEF_ACT[n] (0x00270200 + 0x4*n, n=0...35; RO)

For each switching table that does not hit for a packet, the default action takes place in case its corresponding enable bit is set in GL_SWR_DEF_ACT_EN register (Section 38.39.2.8.1).

Field	Bit(s)	Init.	Type	Description
DEF_ACTION	31:0	0x0	RW	**Default Action** 32-bit switching action per table.

38.39.2.9 PF - Interrupt Registers

38.39.2.9.1 VF Interrupt Throttling for Interrupt N - VFINT_ITRN[n,INTVF] (0x00020000 + 0x800*n + 0x4*INTVF, n=0...2, INTVF=0...511; RW)

Register index ‘n’ relates to interrupt ‘n+1’, while interrupt zero is controlled by the VFINT_ITR0 register (Section 38.39.2.9.6).

Field	Bit(s)	Init.	Type	Description
INTERVAL	11:0	0x0	RW	**Interval** ITR ‘n’ interval, where ‘n’ is the register index = 0,1,2 for the three ITRs per interrupt. It is defined in 2 us units, enabling interval range from zero to 8160 us (0xFF0). Setting the INTERVAL to zero enables immediate interrupt. This register can also be programmed by setting the *INTERVAL* field in the matched xxINT_DYN_CTLx register.
RESERVED	31:12	0x0	RSV	Reserved.

38.39.2.9.2 VF Interrupt N Dynamic Control - VFINT_DYN_CTLN[INTVF] (0x00024800 + 0x4*INTVF, INTVF=0…511; RW)

Register index ‘n’ relates to interrupt ‘n+1’, while interrupt zero is controlled by the VFINT_DYN_CTL0 register (Section 38.39.2.9.8).

Field	Bit(s)	Init.	Type	Description
INTENA	0	0b	RW	**Interrupt Enable** 0b = Interrupt disabled. 1b = Interrupt enabled. Refer to auto-clear policy in the Interrupt Enablement section. This bit is meaningful only if the *INTENA_MSK* flag in this register is not set.
CLEARPBA	1	0b	RW1C	**Clear PBA** Setting this bit clears the matched PBA bit. This bit is auto-cleared by hardware.
SWINT_TRIG	2	0b	RW1C	**Software Interrupt Trigger** When this bit is set, a software interrupt is triggered. This bit is auto-cleared by hardware.
ITR_INDX	4:3	00b	RW1C	**ITR Index** Defines the ITR index to be updated, as follows: 00b = ITR0. 01b = ITR1. 10b = ITR2. 11b = No ITR update. This field is auto-cleared by hardware.
INTERVAL	16:5	0x0	RW1C	**Interval** The interval for the ITR defined by the *ITR_INDX* field in this register. This field is auto-cleared by hardware.
RESERVED	23:17	0x0	RSV	Reserved.
SW_ITR_INDX_ENA	24	0b	RW1C	**Software ITR Index Enable** Enables the programming of the *SW_ITR_INDX* field in this register. This flag is auto-cleared by hardware.
SW_ITR_INDX	26:25	00b	RW	**Software ITR Index** ITR index of the software interrupt: 00b = ITR0. 01b = ITR1. 10b = ITR2. 11b = No ITR. When programming this field, the *SW_ITR_INDX_ENA* flag in this register should be set as well.
RESERVED	29:27	000b	RSV	Reserved.
WB_ON_ITR	30	0b	RW	When this bit is set, completed descriptors are indicated to host memory on ITR completion (or No ITR) regardless of the interrupt enablement in this register.
INTENA_MSK	31	0b	RW1C	**Interrupt Enable Mask** When this bit is set, the *INTENA* setting does not impact the device setting. This bit is auto-cleared by hardware.

38.39.2.9.3 Protected VF Interrupt N Linked List - VPINT_LNKLSTN[INTVF] (0x00025000 + 0x4*INTVF, INTVF=0...511; RW)

Register index 'n' relates to interrupt 'n+1', while interrupt zero is controlled by the VPINT_LNKLST0 register (Section 38.39.2.9.9).

Field	Bit(s)	Init.	Type	Description
FIRSTQ_INDX	10:0	0x7FF	RW	**First Queue Index** First queue index in the MSI-X cause list Transmit and receive queue indexes are within the PF space. CEQ indexes are within the function's space. Setting the index to 0x7FF points to an empty linked list (might be useful for other cause interrupt).
FIRSTQ_TYPE	12:11	00b	RW	**First Queue Type** It can be one of the following: 00b = Receive queues. 01b = Transmit queues. 10b = PE completion event queues. 11b = Reserved.
RESERVED	31:13	0x0	RSV	Reserved.

38.39.2.9.4 Protected VF Interrupt N Rate Limit - VPINT_RATEN[INTVF] (0x00025800 + 0x4*INTVF, INTVF=0...511; RW)

These registers affect the VF, but are exposed only to the parent PF. Register index 'n' relates to interrupt 'n+1', while interrupt zero is controlled by the VPINT_RATE0 register (Section 38.39.2.9.10).

Field	Bit(s)	Init.	Type	Description
INTERVAL	5:0	0x0	RW	**Interval** Time interval defined in 4 us units between consecutive credit incremental. When the interrupt rate limit is enabled by the *INTRL_ENA* flag in this register, the *INTERVAL* must be greater than zero. For accurate rate limit, the *INTERVAL* must be smaller than 0x3C (up to 236 μs).
INTRL_ENA	6	0b	RW	**Interrupt Rate Limit Enable** Enable interrupt rate limit on this interrupt vector.
RESERVED	31:7	0x0	RSV	Reserved.

38.39.2.9.5 VF PE Completion Event Queue Interrupt Cause Control - VPINT_CEQCTL[INTVF] (0x00026800 + 0x4*INTVF, INTVF=0...511; RW)

Field	Bit(s)	Init.	Type	Description
MSIX_INDX	7:0	0x0	RW	**MSI-X Index** MSI-X vector index within the function space. Software should set the *MSIX_INDX* field to values in the range of allocated interrupt vectors to the function.
RESERVED	10:8	000b	RSV	Reserved.
ITR_INDX	12:11	00b	RW	**ITR Index** ITR index of the interrupt cause: 00b = ITR0. 01b = ITR1. 10b = ITR2. 11b = No ITR.

Field	Bit(s)	Init.	Type	Description
MSIX0_INDX	15:13	000b	RW	**MSI-X 0 Index** The index of the QUEUE_x bits in the ICR0 register ('x' = 0,...7), which is set as a result of an event on the queue. This field is relevant only if *MSIX_INDX* equals zero.
NEXTQ_INDX	26:16	0x0	RW	**Next Queue Index** Next queue index in the MSI-X cause list. Transmit and receive queue indexes are within the PF space. CEQ indexes are within the function's space. Setting the index to 0x7FF is a NULL pointer indicating the end of the linked list.
NEXTQ_TYPE	28:27	00b	RW	**Next Queue Type** It can be one of the following: 00b = Receive queues. 01b = Transmit queues. 10b = PE completion event queues. 11b = Reserved.
RESERVED	29	0b	RSV	Reserved.
CAUSE_ENA	30	0b	RW	**Cause Enable** Enable interrupt by this queue. When this bit is cleared, interrupts are not generated by the queue. The queue remains in the interrupt linked list and is processed at ITR expiration.
INTEVENT	31	0b	RO	**Interrupt Event indication** Triggered by the specific event on the queue, and cleared when the interrupt is acknowledged by the interrupt signal logic.

38.39.2.9.6 VF Interrupt Throttling for Interrupt Zero - VFINT_ITR0[n,VF] (0x00028000 + 0x400*n + 0x4*VF, n=0...2, VF=0...127; RW)

Field	Bit(s)	Init.	Type	Description
INTERVAL	11:0	0x0	RW	**Interval** ITR 'n' interval, where 'n' is the register index = 0,1,2 for the three ITRs per interrupt. It is defined in 2 us units, enabling interval range from zero to 8160 us (0xFF0). Setting the INTERVAL to zero enables immediate interrupt. This register can also be programmed by setting the *INTERVAL* field in the matched xxINT_DYN_CTLx register.
RESERVED	31:12	0x0	RSV	Reserved.

38.39.2.9.7 VF Interrupt Zero Static Control - VFINT_STAT_CTL0[VF] (0x0002A000 + 0x4*VF, VF=0...127; RW)

In case of MSI or Legacy INTA mode of operation, interrupt zero is the only valid interrupt.

Field	Bit(s)	Init.	Type	Description
RESERVED	1:0	00b	RSV	Reserved.
OTHER_ITR_INDX	3:2	00b	RW	**Other ITR Index** ITR index of the other interrupt causes: 00b = ITR0. 01b = ITR1. 10b = ITR2. 11b = No ITR.
RESERVED	31:4	0x0	RSV	Reserved.

38.39.2.9.8 VF Interrupt Zero Dynamic Control - VFINT_DYN_CTL0[VF] (0x0002A400 + 0x4*VF, VF=0...127; RW)

Field	Bit(s)	Init.	Type	Description
INTENA	0	0b	RW	**Interrupt Enable** 0b = Interrupt disabled. 1b = Interrupt enabled. Refer to auto-clear policy in the Interrupt Enablement section. This bit is meaningful only if the *INTENA_MSK* flag in this register is not set.
CLEARPBA	1	0b	RW1C	**Clear PBA** Setting this bit clears the matched PBA bit. This bit is auto-cleared by hardware.
SWINT_TRIG	2	0b	RW1C	**Software Interrupt Trigger** When this bit is set, a software interrupt is triggered. This bit is auto-cleared by hardware.
ITR_INDX	4:3	00b	RW1C	**ITR Index** Defines the ITR index to be updated, as follows: 00b = ITR0. 01b = ITR1. 10b = ITR2. 11b = No ITR update. This field is auto-cleared by hardware.
INTERVAL	16:5	0x0	RW1C	**Interval** The interval for the ITR defined by the *ITR_INDX* field in this register. This field is auto-cleared by hardware.
RESERVED	23:17	0x0	RSV	Reserved.
SW_ITR_INDX_ENA	24	0b	RW1C	**Software ITR Index Enable** Enables the programming of the *SW_ITR_INDX* field in this register. This flag is auto-cleared by hardware.
SW_ITR_INDX	26:25	00b	RW	**Software ITR Index** ITR index of the software interrupt: 00b = ITR0. 01b = ITR1. 10b = ITR2. 11b = No ITR. When programming this field, the *SW_ITR_INDX_ENA* flag in this register should be set as well.
RESERVED	29:27	000b	RSV	Reserved.
WB_ON_ITR	30	0b	RW	When this bit is set, completed descriptors are indicated to host memory on ITR completion (or No ITR) regardless of the interrupt enablement in this register.
INTENA_MSK	31	0b	RW1C	**Interrupt Enable Mask** When this bit is set, the *INTENA* setting does not impact the device setting. This bit is auto-cleared by hardware.

38.39.2.9.9 Protected VF Interrupt Zero Linked List - VPINT_LNKLST0[VF] (0x0002A800 + 0x4*VF, VF=0...127; RW)

In case of MSI or legacy INTA mode of operation, interrupt zero is the only valid interrupt.

Field	Bit(s)	Init.	Type	Description
FIRSTQ_INDX	10:0	0x7FF	RW	**First Queue Index** First queue index in the MSI-X cause list. Transmit and receive queue indexes are within the PF space. CEQ indexes are within the function's space. Setting the index to 0x7FF points to an empty linked list (might be useful for other cause interrupt).
FIRSTQ_TYPE	12:11	00b	RW	**First Queue Type** It can be one of the following: 00b = Receive queues. 01b = Transmit queues. 10b = PE completion event queues. 11b = Reserved.
RESERVED	31:13	0x0	RSV	Reserved.

38.39.2.9.10 Protected VF Interrupt Zero Rate Limit - VPINT_RATE0[VF] (0x0002AC00 + 0x4*VF, VF=0...127; RW)

This register affects the VF, but is exposed only to the parent PF.

Field	Bit(s)	Init.	Type	Description
INTERVAL	5:0	0x0	RW	**Interval** Time interval defined in 4 us units between consecutive credit incremental. When the interrupt rate limit is enabled by the *INTRL_ENA* flag in this register, the *INTERVAL* must be greater than zero. For accurate rate limit, the *INTERVAL* must be smaller than 0x3C (up to 236 μs).
INTRL_ENA	6	0b	RW	**Interrupt Rate Limit Enable** Enable interrupt rate limit on this interrupt vector.
RESERVED	31:7	0x0	RSV	Reserved.

38.39.2.9.11 VF PE Asynchronous Event Queue Interrupt Cause Control - VPINT_AEQCTL[VF] (0x0002B800 + 0x4*VF, VF=0...127; RW)

Field	Bit(s)	Init.	Type	Description
MSIX_INDX	7:0	0x0	RW	**MSI-X Index** MSI-X vector index within the function space. Software should set the *MSIX_INDX* field to values in the range of allocated interrupt vectors to the function.
RESERVED	10:8	000b	RSV	Reserved.
ITR_INDX	12:11	00b	RW	**ITR Index** ITR index of the interrupt cause: 00b = ITR0. 01b = ITR1. 10b = ITR2. 11b = No ITR.
MSIX0_INDX	15:13	000b	RW	**MSI-X 0 Index** The index of the QUEUE_x bits in the ICR0 register ('x' = 0,...7), which is set as a result of an event on the queue. This field is relevant only if *MSIX_INDX* equals zero.

Field	Bit(s)	Init.	Type	Description
RESERVED	29:16	0x0	RSV	Reserved.
CAUSE_ENA	30	0b	RW	**Cause Enable** Enable interrupt by this queue. When this bit is cleared, interrupts are not generated by the queue. The queue remains in the interrupt linked list and is processed at ITR expiration.
RESERVED	31	0b	RSV	Reserved.

38.39.2.9.12 VF Interrupt Zero Cause - VFINT_ICR0[VF] (0x0002BC00 + 0x4*VF, VF=0...127; RCW)

Field	Bit(s)	Init.	Type	Description
INTEVENT	0	0b	RCW	**Interrupt Event indication** This bit is set on assertion of any causes for this interrupt, and cleared when the interrupt is asserted to the rate limit logic.
QUEUE_0	1	0b	RCW	**Queue 0** Queue 0 interrupt for LAN transmit and receive queues and PE CEQs.
QUEUE_1	2	0b	RCW	**Queue 1** Queue 1 interrupt for LAN transmit and receive queues and PE CEQs.
QUEUE_2	3	0b	RCW	**Queue 2** Queue 2 interrupt for LAN transmit and receive queues and PE CEQs.
QUEUE_3	4	0b	RCW	**Queue 3** Queue 3 interrupt for LAN transmit and receive queues and PE CEQs.
RESERVED	29:5	0x0	RSV	Reserved.
ADMINQ	30	0b	RCW	**Admin Queue** Send/receive admin queue interrupt indication.
SWINT	31	0b	RCW	**Software Interrupt indication**

38.39.2.9.13 VF Interrupt Zero Cause Enablement - VFINT_ICR0_ENA[VF] (0x0002C000 + 0x4*VF, VF=0...127; RW)

Field	Bit(s)	Init.	Type	Description
RESERVED	29:0	0x0	RSV	Reserved.
ADMINQ	30	0b	RW	**Admin Queue** Enable this interrupt at 1b.
RESERVED	31	0b	RW	Reserved.

38.39.2.9.14 PF Interrupt Throttling for Interrupt N - PFINT_ITRN[n,INTPF] (0x00030000 + 0x800*n + 0x4*INTPF, n=0...2, INTPF=0...511; RW)

Register index 'n' relates to interrupt 'n+1', while interrupt zero is controlled by the PFINT_ITR0 register (Section 38.39.2.9.19). In case of MSI or legacy INTA, only interrupt zero is valid, so none of these registers impact the device functionality.

Field	Bit(s)	Init.	Type	Description
INTERVAL	11:0	0x0	RW	**Interval** ITR 'n' interval, where 'n' is the register index = 0,1,2 for the three ITRs per interrupt. It is defined in 2 us units, enabling interval range from zero to 8160 us (0xFF0). Setting the INTERVAL to zero enables immediate interrupt. This register can also be programmed by setting the *INTERVAL* field in the matched xxINT_DYN_CTLx register.
RESERVED	31:12	0x0	RSV	Reserved.

38.39.2.9.15 PF Interrupt N Dynamic Control - PFINT_DYN_CTLN[INTPF] (0x00034800 + 0x4*INTPF, INTPF=0...511; RW)

Register index 'n' relates to interrupt 'n+1', while interrupt zero is controlled by the PFINT_DYN_CTL0 register (Section 38.39.2.9.21). In case of MSI or legacy INTA, only interrupt zero is valid, so none of these registers impact the device functionality.

Field	Bit(s)	Init.	Type	Description
INTENA	0	0b	RW	**Interrupt Enable** 0b = Interrupt disabled. 1b = Interrupt enabled. Refer to auto-clear policy in the Interrupt Enablement section. This bit is meaningful only if the *INTENA_MSK* flag in this register is not set.
CLEARPBA	1	0b	RW1C	**Clear PBA** Setting this bit clears the matched PBA bit. This bit is auto-cleared by hardware.
SWINT_TRIG	2	0b	RW1C	**Software Interrupt Trigger** When this bit is set, a software interrupt is triggered. This bit is auto-cleared by hardware.
ITR_INDX	4:3	00b	RW1C	**ITR Index** Defines the ITR index to be updated, as follows: 00b = ITR0. 01b = ITR1. 10b = ITR2. 11b = No ITR update. This field is auto-cleared by hardware.
INTERVAL	16:5	0x0	RW1C	**Interval** The interval for the ITR defined by the *ITR_INDX* field in this register. This field is auto-cleared by hardware.
RESERVED	23:17	0x0	RSV	Reserved.
SW_ITR_INDX_ENA	24	0b	RW1C	**Software ITR Index Enable** Enables the programming of the *SW_ITR_INDX* field in this register. This flag is auto-cleared by hardware.
SW_ITR_INDX	26:25	00b	RW	**Software ITR Index** ITR index of the software interrupt: 00b = ITR0. 01b = ITR1. 10b = ITR2. 11b = No ITR. When programming this field, the *SW_ITR_INDX_ENA* flag in this register should be set as well.
RESERVED	29:27	000b	RSV	Reserved.
WB_ON_ITR	30	0b	RW	When this bit is set, completed descriptors are indicated to host memory on ITR completion (or No ITR) regardless of the interrupt enablement in this register.
INTENA_MSK	31	0b	RW1C	**Interrupt Enable Mask** When this bit is set, the *INTENA* setting does not impact the device setting. This bit is auto-cleared by hardware.

38.39.2.9.16 PF Interrupt N Linked List - PFINT_LNKLSTN[INTPF] (0x00035000 + 0x4*INTPF, INTPF=0...511; RW)

Register index 'n' relates to interrupt 'n+1', while interrupt zero is controlled by the PFINT_LNKLST0 register (Section 38.39.2.9.22). In case of MSI or legacy INTA, only interrupt zero is valid, so none of these registers impact the device functionality.

Field	Bit(s)	Init.	Type	Description
FIRSTQ_INDX	10:0	0x7FF	RW	**First Queue Index** First queue index in the MSI-X cause list Transmit and receive queue indexes are within the PF space. CEQ indexes are within the function's space. Setting the index to 0x7FF points to an empty linked list (might be useful for "other cause" interrupt).
FIRSTQ_TYPE	12:11	00b	RW	**First Queue Type** It can be one of the following: 00b = Receive queues. 01b = Transmit queues. 10b = PE completion event queues. 11b = Reserved.
RESERVED	31:13	0x0	RSV	Reserved.

38.39.2.9.17 PF Interrupt N Rate Limit - PFINT_RATEN[INTPF] (0x00035800 + 0x4*INTPF, INTPF=0...511; RW)

Register index 'n' relates to interrupt 'n+1', while interrupt zero is controlled by the PFINT_RATE0 register (Section 38.39.2.9.23). In case of MSI or legacy INTA, only interrupt zero is valid, so none of these registers impact the device functionality.

Field	Bit(s)	Init.	Type	Description
INTERVAL	5:0	0x0	RW	**Interval** Time interval defined in 4 us units between consecutive credit incremental. When the interrupt rate limit is enabled by the *INTRL_ENA* flag in this register, the *INTERVAL* must be greater than zero. For accurate rate limit, the *INTERVAL* must be smaller than 0x3C (up to 236 μs).
INTRL_ENA	6	0b	RW	**Interrupt Rate Limit Enable** Enable interrupt rate limit on this interrupt vector.
RESERVED	31:7	0x0	RSV	Reserved.

38.39.2.9.18 PF PE Completion Event Queue Interrupt Cause Control - PFINT_CEQCTL[INTPF] (0x00036800 + 0x4*INTPF, INTPF=0...511; RW)

Field	Bit(s)	Init.	Type	Description
MSIX_INDX	7:0	0x0	RW	**MSI-X Index** MSI-X vector index within the function space. Software should set the *MSIX_INDX* field to values in the range of allocated interrupt vectors to the function.
RESERVED	10:8	000b	RSV	Reserved.
ITR_INDX	12:11	00b	RW	**ITR Index** ITR index of the interrupt cause: 00b = ITR0. 01b = ITR1. 10b = ITR2. 11b = No ITR.

Field	Bit(s)	Init.	Type	Description
MSIX0_INDX	15:13	000b	RW	**MSI-X 0 Index** The index of the QUEUE_x bits in the ICR0 register ('x' = 0,...7), which is set as a result of an event on the queue. This field is relevant only if *MSIX_INDX* equals zero.
NEXTQ_INDX	26:16	0x0	RW	**Next Queue Index** Next queue index in the MSI-X cause list. Transmit and receive queue indexes are within the PF space. CEQ indexes are within the function's space. Setting the index to 0x7FF is a NULL pointer indicating the end of the linked list.
NEXTQ_TYPE	28:27	00b	RW	**Next Queue Type** It can be one of the following: 00b = Receive queues. 01b = Transmit queues. 10b = PE completion event queues. 11b = Reserved.
RESERVED	29	0b	RSV	Reserved.
CAUSE_ENA	30	0b	RW	**Cause Enable** Enable interrupt by this queue. When this bit is cleared, interrupts are not generated by the queue. The queue remains in the interrupt linked list and is processed at ITR expiration.
INTEVENT	31	0b	RO	**Interrupt Event indication** Triggered by the specific event on the queue, and cleared when the interrupt is acknowledged by the interrupt signal logic.

38.39.2.9.19 PF Interrupt Throttling for Interrupt Zero - PFINT_ITR0[n] (0x00038000 + 0x80*n, n=0...2; RW)

Field	Bit(s)	Init.	Type	Description
INTERVAL	11:0	0x0	RW	**Interval** ITR 'n' interval, where 'n' is the register index = 0,1,2 for the three ITRs per interrupt. It is defined in 2 us units, enabling interval range from zero to 8160 us (0xFF0). Setting the INTERVAL to zero enables immediate interrupt. This register can also be programmed by setting the *INTERVAL* field in the matched xxINT_DYN_CTLx register.
RESERVED	31:12	0x0	RSV	Reserved.

38.39.2.9.20 PF Interrupt Zero Static Control - PFINT_STAT_CTL0 (0x00038400; RW)

In case of MSI or legacy INTA mode of operation, interrupt zero is the only valid interrupt.

Field	Bit(s)	Init.	Type	Description
RESERVED	1:0	00b	RSV	Reserved.
OTHER_ITR_INDX	3:2	00b	RW	**Other ITR Index** ITR index of the other interrupt causes: 00b = ITR0. 01b = ITR1. 10b = ITR2. 11b = No ITR.
RESERVED	31:4	0x0	RSV	Reserved.

38.39.2.9.21 PF Interrupt Zero Dynamic Control - PFINT_DYN_CTL0 (0x00038480; RW)

In case of MSI or legacy INTA mode of operation, interrupt zero is the only valid interrupt.

Field	Bit(s)	Init.	Type	Description
INTENA	0	0b	RW	**Interrupt Enable** 0b = Interrupt disabled. 1b = Interrupt enabled. Refer to auto-clear policy in the Interrupt Enablement section. This bit is meaningful only if the *INTENA_MSK* flag in this register is not set.
CLEARPBA	1	0b	RW1C	**Clear PBA** Setting this bit clears the matched PBA bit. This bit is auto-cleared by hardware.
SWINT_TRIG	2	0b	RW1C	**Software Interrupt Trigger** When this bit is set, a software interrupt is triggered. This bit is auto-cleared by hardware.
ITR_INDX	4:3	00b	RW1C	**ITR Index** Defines the ITR index to be updated, as follows: 00b = ITR0. 01b = ITR1. 10b = ITR2. 11b = No ITR update. This field is auto-cleared by hardware.
INTERVAL	16:5	0x0	RW1C	**Interval** The interval for the ITR defined by the *ITR_INDX* field in this register. This field is auto-cleared by hardware.
RESERVED	23:17	0x0	RSV	Reserved.
SW_ITR_INDX_ENA	24	0b	RW1C	**Software ITR Index Enable** Enables the programming of the *SW_ITR_INDX* field in this register. This flag is auto-cleared by hardware.
SW_ITR_INDX	26:25	00b	RW	**Software ITR Index** ITR index of the software interrupt: 00b = ITR0. 01b = ITR1. 10b = ITR2. 11b = No ITR. When programming this field, the *SW_ITR_INDX_ENA* flag in this register should be set as well.
RESERVED	29:27	000b	RSV	Reserved.
WB_ON_ITR	30	0b	RW	When this bit is set, completed descriptors are indicated to host memory on ITR completion (or No ITR) regardless of the interrupt enablement in this register.
INTENA_MSK	31	0b	RW1C	**Interrupt Enable Mask** When this bit is set, the *INTENA* setting does not impact the device setting. This bit is auto-cleared by hardware.

38.39.2.9.22 PF Interrupt Zero Linked List - PFINT_LNKLST0 (0x00038500; RW)

In case of MSI or legacy INTA mode of operation, interrupt zero is the only valid interrupt.

Field	Bit(s)	Init.	Type	Description
FIRSTQ_INDX	10:0	0x7FF	RW	**First Queue Index** First queue index in the MSI-X cause list Transmit and receive queue indexes are within the PF space. CEQ indexes are within the function's space. Setting the index to 0x7FF points to an empty linked list (might be useful for other cause interrupt).
FIRSTQ_TYPE	12:11	00b	RW	**First Queue Type** It can be one of the following: 00b = Receive queues. 01b = Transmit queues. 10b = PE completion event queues. 11b = Reserved.
RESERVED	31:13	0x0	RSV	Reserved.

38.39.2.9.23 PF Interrupt Zero Rate Limit - PFINT_RATE0 (0x00038580; RW)

Field	Bit(s)	Init.	Type	Description
INTERVAL	5:0	0x0	RW	**Interval** Time interval defined in 4 us units between consecutive credit incremental. When the interrupt rate limit is enabled by the *INTRL_ENA* flag in this register, the *INTERVAL* must be greater than zero. For accurate rate limit, the *INTERVAL* must be smaller than 0x3C (up to 236 µs).
INTRL_ENA	6	0b	RW	**Interrupt Rate Limit Enable** Enable interrupt rate limit on this interrupt vector.
RESERVED	31:7	0x0	RSV	Reserved.

38.39.2.9.24 PF PE Asynchronous Event Queue Interrupt Cause Control - PFINT_AEQCTL (0x00038700; RW)

Field	Bit(s)	Init.	Type	Description
MSIX_INDX	7:0	0x0	RW	**MSI-X Index** MSI-X vector index within the function space. Software should set the *MSIX_INDX* field to values in the range of allocated interrupt vectors to the function.
RESERVED	10:8	000b	RSV	Reserved.
ITR_INDX	12:11	00b	RW	**ITR Index** ITR index of the interrupt cause: 00b = ITR0. 01b = ITR1. 10b = ITR2. 11b = No ITR.
MSIX0_INDX	15:13	000b	RW	**MSI-X 0 Index** The index of the QUEUE_x bits in the ICR0 register ('x' = 0,...7), which is set as a result of an event on the queue. This field is relevant only if *MSIX_INDX* equals zero.
RESERVED	29:16	0x0	RSV	Reserved.

Field	Bit(s)	Init.	Type	Description
CAUSE_ENA	30	0b	RW	**Cause Enable** Enable interrupt by this queue. When this bit is cleared, interrupts are not generated by the queue. The queue remains in the interrupt linked list and is processed at ITR expiration.
RESERVED	31	0b	RSV	Reserved.

38.39.2.9.25 PF Interrupt Zero Cause - PFINT_ICR0 (0x00038780; RCW)

Field	Bit(s)	Init.	Type	Description
INTEVENT	0	0b	RCW	**Interrupt Event indication** This bit is set on assertion of any causes for this interrupt, and cleared when the interrupt is asserted to the rate limit logic.
QUEUE_0	1	0b	RCW	**Queue 0** Queue 0 interrupt for LAN transmit and receive queues and PE CEQs.
QUEUE_1	2	0b	RCW	**Queue 1** Queue 1 interrupt for LAN transmit and receive queues and PE CEQs.
QUEUE_2	3	0b	RCW	**Queue 2** Queue 2 interrupt for LAN transmit and receive queues and PE CEQs.
QUEUE_3	4	0b	RCW	**Queue 3** Queue 3 interrupt for LAN transmit and receive queues and PE CEQs.
QUEUE_4	5	0b	RCW	**Queue 4** Queue 4 interrupt for LAN transmit and receive queues and PE CEQs.
QUEUE_5	6	0b	RCW	**Queue 5** Queue 5 interrupt for LAN transmit and receive queues and PE CEQs.
QUEUE_6	7	0b	RCW	**Queue 6** Queue 6 i.nterrupt for LAN transmit and receive queues and PE CEQs
QUEUE_7	8	0b	RCW	**Queue 7** Queue 7 interrupt for LAN transmit and receive queues and PE CEQs.
RESERVED	15:9	0x0	RSV	Reserved.
ECC_ERR	16	0b	RCW	**Unrecoverable ECC Error** This bit is set when an unrecoverable error is detected in one of the device memories.
RESERVED	18:17	00b	RSV	Reserved.
MAL_DETECT	19	0b	RCW	**Malicious Programming Detected**
GRST	20	0b	RCW	**Global Resets Requested** (CORER, GLOBR or EMPR)
RESERVED	21	0b	RSV	Reserved.
GPIO	22	0b	RCW	**GPIO Event** Indicates an event on any of the GPIO pins enabled for interrupt by the PFINT_GPIOCTL register. The GPIO state can be fetched on the GLGEN_GPIO_STAT register (Section 38.39.2.1.6). The level transition that generates an interrupt is set for GPIO 'n' by the *INT_MODE* field in the matched GLGEN_GPIO_CTL[n] register (Section 38.39.2.1.4).
TIMESYNC	23	0b	RCW	**TimeSynch** Any of the TimeSync interrupt causes as described in the interrupts section of the TimeSync (IEEE1588 and 802.1AS) section.
RESERVED	25:24	00b	RSV	Reserved.
HMC_ERR	26	0b	RCW	**HMC Error** HMC error as indicated in the PFHMC_ERRORINFO and PFHMC_ERRORDATA registers (Section 38.39.2.13.8 and Section 38.39.2.13.9, respectively).
RESERVED	28:27	00b	RSV	Reserved.

Field	Bit(s)	Init.	Type	Description
VFLR	29	0b	RCW	**VFLR** VFLR was initiated by one of the VFs of the PF. The PF should read the GLGEN_VFLRSTAT register (Section 38.39.2.1.22) for an indication for the VF that generated the VFLR.
ADMINQ	30	0b	RCW	**Admin Queue** Send/receive admin queue interrupt indication.
SWINT	31	0b	RCW	**Software Interrupt Indication**

38.39.2.9.26 PF Interrupt Zero Cause Enablement - PFINT_ICR0_ENA (0x00038800; RW)

Field	Bit(s)	Init.	Type	Description
RESERVED	15:0	0x0	RSV	Reserved.
ECC_ERR	16	0b	RW	**ECC Error** Enable this interrupt at 1b.
RESERVED	18:17	00b	RSV	Reserved.
MAL_DETECT	19	0b	RW	**Malicious Programming Detected** Enable this interrupt at 1b.
GRST	20	0b	RW	**Global Resets Requested** Enable this interrupt at 1b.
PCI_EXCEPTION	21	0b	RW	**PCI Exception** Enable this interrupt at 1b.
GPIO	22	0b	RW	**GPIO Event** Enable this interrupt at 1b.
TIMESYNC	23	0b	RW	**TimeSynch** Enable this interrupt at 1b.
RESERVED	25:24	00b	RSV	Reserved.
HMC_ERR	26	0b	RW	**HMC Error** Enable this interrupt at 1b.
RESERVED	28:27	00b	RSV	Reserved.
VFLR	29	0b	RW	**VFLR** Enable this interrupt at 1b.
ADMINQ	30	0b	RW	**Admin Queue** Enable this interrupt at 1b.
RESERVED	31	0b	RSV	Reserved.

38.39.2.9.27 Receive Queue Interrupt Cause Control - QINT_RQCTL[Q] (0x0003A000 + 0x4*Q, Q=0...1535; RW)

Field	Bit(s)	Init.	Type	Description
MSIX_INDX	7:0	0x0	RW	**MSI-X Index** MSI-X vector index within the function space. Software should set the *MSIX_INDX* field to values in the range of allocated interrupt vectors to the function.
RESERVED	10:8	000b	RSV	Reserved.
ITR_INDX	12:11	00b	RW	**ITR Index** ITR index of the interrupt cause: 00b = ITR0. 01b = ITR1. 10b = ITR2. 11b = No ITR.
MSIX0_INDX	15:13	000b	RW	**MSI-X 0 Index** The index of the QUEUE_x bits in the ICR0 register ('x' = 0,...7), which is set as a result of an event on the queue. This field is relevant only if *MSIX_INDX* equals zero.
NEXTQ_INDX	26:16	0x0	RW	**Next Queue Index** Next queue index in the MSI-X cause list. Transmit and receive queue indexes are within the PF space. CEQ indexes are within the function's space. Setting the index to 0x7FF is a NULL pointer indicating the end of the linked list.
NEXTQ_TYPE	28:27	00b	RW	**Next Queue Type** It can be one of the following: 00b = Receive queues. 01b = Transmit queues. 10b = PE completion event queues. 11b = Reserved.
RESERVED	29	0b	RSV	Reserved.
CAUSE_ENA	30	0b	RW	**Cause Enable** Enable interrupt by this queue. When this bit is cleared, interrupts are not generated by the queue. The queue remains in the interrupt linked list and is processed at ITR expiration.
INTEVENT	31	0b	RO	**Interrupt Event indication** Triggered by the specific event on the queue, and cleared when the interrupt is acknowledged by the interrupt signal logic.

38.39.2.9.28 Transmit Queue Interrupt Cause Control - QINT_TQCTL[Q] (0x0003C000 + 0x4*Q, Q=0...1535; RW)

Field	Bit(s)	Init.	Type	Description
MSIX_INDX	7:0	0x0	RW	**MSI-X Index** MSI-X vector index within the function space. Software should set the *MSIX_INDX* field to values in the range of allocated interrupt vectors to the function.
RESERVED	10:8	000b	RSV	Reserved.
ITR_INDX	12:11	00b	RW	**ITR Index** ITR index of the interrupt cause: 00b = ITR0. 01b = ITR1. 10b = ITR2. 11b = No ITR.

Field	Bit(s)	Init.	Type	Description
MSIX0_INDX	15:13	000b	RW	**MSI-X 0 Index** The index of the QUEUE_x bits in the ICR0 register ('x' = 0,...7), which is set as a result of an event on the queue. This field is relevant only if *MSIX_INDX* equals zero.
NEXTQ_INDX	26:16	0x0	RW	**Next Queue Index** Next queue index in the MSI-X cause list. Transmit and receive queue indexes are within the PF space. CEQ indexes are within the function's space. Setting the index to 0x7FF is a NULL pointer indicating the end of the linked list.
NEXTQ_TYPE	28:27	00b	RW	**Next Queue Type** It can be one of the following: 00b = Receive queues. 01b = Transmit queues. 10b = PE completion event queues. 11b = Reserved.
RESERVED	29	0b	RSV	Reserved.
CAUSE_ENA	30	0b	RW	**Cause Enable** Enable interrupt by this queue. When this bit is cleared, interrupts are not generated by the queue. The queue remains in the interrupt linked list and is processed at ITR expiration.
INTEVENT	31	0b	RO	**Interrupt Event indication** Triggered by the specific event on the queue, and cleared when the interrupt is acknowledged by the interrupt signal logic.

38.39.2.9.29 LAN Port MDIO Number - PFGEN_PORTMDIO_NUM (0x0003F100; RO)

Field	Bit(s)	Init.	Type	Description
PORT_NUM	1:0	00b	RW	**Port Number** Indicates the LAN port connected to this function. 00b = Port 0. 01b = Port 1. 10b = Port 2. 11b = Port 3. This field must be identical to the PFGEN_PORTNUM register (Section 38.39.2.1.31).
RESERVED	31:2	0x0	RSV	Reserved.

38.39.2.9.30 PF General Purpose I/O Interrupt Enablement - PFINT_GPIO_ENA (0x00088080; RW)

Field	Bit(s)	Init.	Type	Description
GPIO0_ENA	0	0b	RW	Enable interrupt on GPIO 0.
GPIO1_ENA	1	0b	RW	Enable interrupt on GPIO 1.
GPIO2_ENA	2	0b	RW	Enable interrupt on GPIO 2.
GPIO3_ENA	3	0b	RW	Enable interrupt on GPIO 3.
GPIO4_ENA	4	0b	RW	Enable interrupt on GPIO 4.
GPIO5_ENA	5	0b	RW	Enable interrupt on GPIO 5.
GPIO6_ENA	6	0b	RW	Enable interrupt on GPIO 6.
GPIO7_ENA	7	0b	RW	Enable interrupt on GPIO 7.
GPIO8_ENA	8	0b	RW	Enable interrupt on GPIO 8.
GPIO9_ENA	9	0b	RW	Enable interrupt on GPIO 9.
GPIO10_ENA	10	0b	RW	Enable interrupt on GPIO 10.
GPIO11_ENA	11	0b	RW	Enable interrupt on GPIO 11.
GPIO12_ENA	12	0b	RW	Enable interrupt on GPIO 12.
GPIO13_ENA	13	0b	RW	Enable interrupt on GPIO 13.
GPIO14_ENA	14	0b	RW	Enable interrupt on GPIO 14.
GPIO15_ENA	15	0b	RW	Enable interrupt on GPIO 15.
GPIO16_ENA	16	0b	RW	Enable interrupt on GPIO 16.
GPIO17_ENA	17	0b	RW	Enable interrupt on GPIO 17.
GPIO18_ENA	18	0b	RW	Enable interrupt on GPIO 18.
GPIO19_ENA	19	0b	RW	Enable interrupt on GPIO 19.
GPIO20_ENA	20	0b	RW	Enable interrupt on GPIO 20.
GPIO21_ENA	21	0b	RW	Enable interrupt on GPIO 21.
GPIO22_ENA	22	0b	RW	Enable interrupt on GPIO 22.
GPIO23_ENA	23	0b	RW	Enable interrupt on GPIO 23.
GPIO24_ENA	24	0b	RW	Enable interrupt on GPIO 24.
GPIO25_ENA	25	0b	RW	Enable interrupt on GPIO 25.
GPIO26_ENA	26	0b	RW	Enable interrupt on GPIO 26.
GPIO27_ENA	27	0b	RW	Enable interrupt on GPIO 27.
GPIO28_ENA	28	0b	RW	Enable interrupt on GPIO 28.
GPIO29_ENA	29	0b	RW	Enable interrupt on GPIO 29.
RESERVED	31:30	00b	RSV	Reserved.

38.39.2.9.31 EMP General Purpose I/O Interrupt Enablement - EMPINT_GPIO_ENA (0x00088188; RO)

Field	Bit(s)	Init.	Type	Description
GPIO0_ENA	0	0b	RW	Enable interrupt on GPIO 0.
GPIO1_ENA	1	0b	RW	Enable interrupt on GPIO 1.
GPIO2_ENA	2	0b	RW	Enable interrupt on GPIO 2.
GPIO3_ENA	3	0b	RW	Enable interrupt on GPIO 3.
GPIO4_ENA	4	0b	RW	Enable interrupt on GPIO 4.

Field	Bit(s)	Init.	Type	Description
GPIO5_ENA	5	0b	RW	Enable interrupt on GPIO 5.
GPIO6_ENA	6	0b	RW	Enable interrupt on GPIO 6.
GPIO7_ENA	7	0b	RW	Enable interrupt on GPIO 7.
GPIO8_ENA	8	0b	RW	Enable interrupt on GPIO 8.
GPIO9_ENA	9	0b	RW	Enable interrupt on GPIO 9.
GPIO10_ENA	10	0b	RW	Enable interrupt on GPIO 10.
GPIO11_ENA	11	0b	RW	Enable interrupt on GPIO 11.
GPIO12_ENA	12	0b	RW	Enable interrupt on GPIO 12.
GPIO13_ENA	13	0b	RW	Enable interrupt on GPIO 13.
GPIO14_ENA	14	0b	RW	Enable interrupt on GPIO 14.
GPIO15_ENA	15	0b	RW	Enable interrupt on GPIO 15.
GPIO16_ENA	16	0b	RW	Enable interrupt on GPIO 16.
GPIO17_ENA	17	0b	RW	Enable interrupt on GPIO 17.
GPIO18_ENA	18	0b	RW	Enable interrupt on GPIO 18.
GPIO19_ENA	19	0b	RW	Enable interrupt on GPIO 19.
GPIO20_ENA	20	0b	RW	Enable interrupt on GPIO 20.
GPIO21_ENA	21	0b	RW	Enable interrupt on GPIO 21.
GPIO22_ENA	22	0b	RW	Enable interrupt on GPIO 22.
GPIO23_ENA	23	0b	RW	Enable interrupt on GPIO 23.
GPIO24_ENA	24	0b	RW	Enable interrupt on GPIO 24.
GPIO25_ENA	25	0b	RW	Enable interrupt on GPIO 25.
GPIO26_ENA	26	0b	RW	Enable interrupt on GPIO 26.
GPIO27_ENA	27	0b	RW	Enable interrupt on GPIO 27.
GPIO28_ENA	28	0b	RW	Enable interrupt on GPIO 28.
GPIO29_ENA	29	0b	RW	Enable interrupt on GPIO 29.
RESERVED	31:30	00b	RSV	Reserved.

38.39.2.10 PF - Virtualization PF Registers

38.39.2.10.1 Malicious Driver TX Data Checks Enable - GL_MDCK_TDAT (0x000442F4)

This register selects which TX data checks are enabled, for a list of the defined bits see the Virtualization section.

Field	Bit(s)	Init.	Type	Description
Reserved	1:0	1b	RW	Reserved.
MAL_ LENGTH _DIS	2	0b	RW	Disable malicious detection of bad header length in the transmit descriptor.
MAL_CMD_ DIS	3	0b	RW	Disable malicious detection of bad combination of commands in the transmit descriptor.
Reserved	31:4	0x0	RSV	Reserved.

38.39.2.10.2 Malicious Driver Detected on TX - VP_MDET_TX[VF] (0x000E6000 + 0x4*VF, VF=0...127; RW1C)

This register records a malicious event detected on the Tx queues. Once read, driver must write 0xFFFF to clear.

Field	Bit(s)	Init.	Type	Description
VALID	0	0b	RW1C	**Valid** A malicious event has been detected on this function.
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.10.3 Malicious Driver Detected on Tx - PF_MDET_TX (0x000E6400; RW1C)

This register records a malicious event detected on the Tx queues. Once read, driver must write 0xFFFF to clear.

Field	Bit(s)	Init.	Type	Description
VALID	0	0b	RW1C	**Valid** A malicious event has been detected on this function.
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.10.4 Malicions Driver RX checks enabled - GL_MDCK_RX (0x0012A50C)

This register selects which RX command checks are enabled, for a list of the defined bits see the Virtualization section.

Field	Bit(s)	Init.	Type	Description
DESC_ADDR	0	1b	RW	Enable malicious event: descriptor fetch failed. For proper operation this flag must be active.
Reserved	31:1	0x0	RSV	Reserved.

38.39.2.10.5 Malicious Driver TX Command Checks Enable - GL_MDCK_TCMD (0x000E648C)

This register selects which TX command checks are enabled, for a list of the defined bits see the Virtualization section.

Field	Bit(s)	Init.	Access Type	CFG Policy	Description
DESC_ADDR	0	1b	RW		Enable malicious event: descriptor fetch failed
RESERVED	1	0b	RSV		
MAX_BUFF	2	1b	RW		Enable malicious event: Single send packet or large send segment is spread on more than 8 data descriptors.
MAX_HEAD	3	1b	RW		Enable malicious event: LSO with header spanning more than 3 buffers.
NO_HEAD	4	1b	RW		Enable malicious event: Zero header length when Large send Offload enabled.
WRONG_SIZE	5	1b	RW		Enable malicious event: A single send packet or large send segment that is not between min / max sizes defined in the GLTLAN_MIN_MAX_PKT register.
RESERVED	6	1b	RW		Reserved

Field	Bit(s)	Init.	Access Type	CFG Policy	Description
ENDLESS_TX	7	1b	RW		Enable malicious event: endless transmit ring. When this flag is cleared, endless transmit ring option is enabled. When this flag is set, endless transmit ring (a tail update larger than the ring size) is considered a malicious event.
BAD_LSO_LEN	8	1b	RW		Enable malicious event: LSO PKT_LEN not equal sum of buffers
BAD_LSO_MSS	9	1b	RW		Enable malicious event: LSO with MSS of 0. Functionality of this flag is a subset of bit 13.
RESERVED	10	0b	RSV		
BAD_FD_DES	11	1b	RW		Enable malicious event: When this bit is set, and a certain queue with FD = 0 in queue context tries to send FD init packets, then the queue will be considered as malicious. SW/NVM should not clear it.
M_CONTEXTS	12	1b	RW		Enable malicious event: 7 or more consecutive non-data descriptors are fetched in a transmit queue. Functionality of this flag is a subset of bit 13.
BAD_DESC_SEQUENCE	13	1b	RW		Enable malicious event: Bad descriptors sequence. Include: MSS min/max; Max number of non-data descriptors; DIF/DIX descriptors (enabled by bit 16); Flex descriptors (enabled by bit 17).
BAD_FC_DESC	14	1b	RW		Enable malicious event: When this bit is set then the queue will be considered as malicious. SW/NVM should not clear it.
NO_PACKET	15	1b	RW		Enable malicious event: Tail update that does not contain at least one full packet.
DIS_DIF_DIX	16	1b	RW		Enable malicious event: DIF/DIX descriptors are considered malicious descriptors.
DIS_FLEX	17	1b	RW		Enable malicious event: Flexible descriptors are considered malicious descriptors.
ZERO_BSIZE	18	1b	RW		Enable Malicious event: a descriptor with a zero value in BSIZE
RESERVED	31:19	0x0	RSV		

38.39.2.10.6 Malicious Driver Tx Event Details - GL_MDET_TX (0x000E6480; RW1C)

This register records the details of the first Tx event detected.

Field	Bit(s)	Init.	Type	Description
QNUM	11:0	0x0	RW1C	**Queue Number** Absolute queue ID on which the event was detected.
VF_NUM	20:12	0x0	RW1C	**VF Number** Absolute VF number on which the event was detected.
PF_NUM	24:21	0x0	RW1C	**PF Number** PF/parent PF number on which the event was detected.
MAL_TYPE	29:25	0x0	RW1C	**Malicious Drive Type** ID of the event that has been recorded. For more information, see the Malicious Driver - Tx descriptor checks table.
RESERVED	30	0b	RSV	Reserved.
VALID	31	0b	RW1C	**Valid** Indicates that an event has been captured.

38.39.2.10.7 Malicious Driver Detected on Rx - VP_MDET_RX[VF] (0x0012A000 + 0x4*VF, VF=0...127; RW1C)

This register records a malicious event detected on the Rx queues. Once read, the software device driver must write 0xFFFF to clear.

Field	Bit(s)	Init.	Type	Description
VALID	0	0b	RW1C	**Valid** A malicious event has been detected on this function.
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.10.8 Malicious Driver Detected on Rx - PF_MDET_RX (0x0012A400; RW1C)

This register records a malicious event detected on the Rx queues. Once read, the software device driver must write 0xFFFF to clear.

Field	Bit(s)	Init.	Type	Description
VALID	0	0b	RW1C	**Valid** A malicious event has been detected on this function.
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.10.9 Malicious Driver Rx Event Details - GL_MDET_RX (0x0012A510; RW1C)

This register records the details of the first Rx event detected.

Field	Bit(s)	Init.	Type	Description
FUNCTION	7:0	0x0	RW1C	**Function** The function that triggered the event.
EVENT	16:8	0x0	RW1C	**Event** ID of the event that has been recorded.
QUEUE	30:17	0x0	RW1C	**Queue** Queue ID on which the event was detected.
VALID	31	0b	RW1C	**Valid** Indicates that an event has been captured.

38.39.2.10.10 PF Resources Allocation - PF_VT_PFALLOC (0x001C0500; RO)

Field	Bit(s)	Init.	Type	Description
FIRSTVF	7:0	0x0	RW	**First VF** The first VF allocated to this PF. Valid only if the *VALID* flag is set. Valid values are 0-127.
LASTVF	15:8	0x0	RW	**Last VF** The last VF allocated to this PF. Valid only if the *VALID* flag is set. Valid values are 0-127.
RESERVED	30:16	0x0	RSV	Reserved.
VALID	31	0b	RW	**Valid** The *FIRSTVF* and *LASTVF* fields in this register are valid. If cleared no VFs are allocated to this PF. If cleared, the SR-IOV capability should not be exposed for this PF.

38.39.2.11 PF - DCB Registers

38.39.2.11.1 Port DCB General Control - PRTDCB_GENC (0x00083000; RW)

Field	Bit(s)	Init.	Type	Description
RESERVED	1:0	00b	RSV	Reserved.
NUMTC	5:2	0x1	RW	**Number of TCs** Number of TCs for the port. This field must be set consistently with the settings made in the Tx-scheduler.
RESERVED	8:6	011b	RW	Reserved.
RESERVED	9	1b	RW	1b = Reserved.
RESERVED	15:10	0x0	RSV	Reserved.
PFCLDA	31:16	0x079D	RW	**PFC Link Delay Allowance** Expressed in 16-byte units. Default value assumes 9.5 KB jumbo frames over a 10 GbE link with a 10GBASE-T PHY and 100-meter CAT6 cable (no optimization done for lower link speeds). For a 40 GbE link, the number is 0x1E70.

38.39.2.11.2 Port DCB General Status - PRTDCB_GENS (0x00083020; RO)

Field	Bit(s)	Init.	Type	Description
DCBX_STATUS	2:0	000b	RW	**DCBX Status** 000b = Not_Started. 001b = In_Progress. 010b = Done. 011b = Multiple_Peers. 100b = Reserved. 101b = Reserved. 110b = Reserved. 111b = Disabled.
RESERVED	31:3	0x0	RSV	Reserved.

38.39.2.11.3 Global DCB General Control - GLDCB_GENC (0x00083044; RW)

Field	Bit(s)	Init.	Type	Description
PCIRTT	15:0	0x009C	RW	**PCIe Round Trip Time** Expressed in 16-byte units. Default is 2 us PCIe round trip time assuming 10 GbE links (no optimization done for lower link speeds).
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.11.4 DCB Transmit ETS Control for TPB - PRTDCB_TETSC_TPB (0x00098060; RW)

Field	Bit(s)	Init.	Type	Description
RESERVED	7:0	0x0	RSV	Reserved.
LLTC	15:8	0x0	RW	**Low Latency TC** 8-bit wide bitmap with a 1-bit entry per each TC. Each entry controls whether the TC is considered to have low latency needs for the transmit path. 0b = TC is defined to be a Bulk TC for transmit. 1b = TC is defined to be a Low Latency TC for transmit.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.11.5 DCB Transmit Frame Monitoring Status Per TC - PRTDCB_TFMSTC[n] (0x000A0040 + 0x20*n, n=0...7; RO)

One register per TC. Register index corresponds to TCID.

Field	Bit(s)	Init.	Type	Description
MSTC	19:0	0x0	RW	**Monitoring Status of the TC** Number of bytes in transit from the host to the TPB (TPB waiting list included) for TC n, where n is the index of the register in the array.
RESERVED	31:20	0x0	RSV	Reserved.

38.39.2.11.6 DCB Transmit Data Pipe Monitor Control - PRTDCB_TDPMC (0x000A0180; RW)

Field	Bit(s)	Init.	Type	Description
RESERVED	31:0	0x0	RSV	Reserved.

38.39.2.11.7 DCB Transmit Command Waiting Status Per TC - PRTDCB_TCMSTC[n] (0x000A2040 + 0x20*n, n=0...7; RO)

One register per TC. Register index corresponds to TCID.

Field	Bit(s)	Init.	Type	Description
MSTC	19:0	0x0	RW	**Monitoring Status of the TC** Number of commands in transit from the host to the TCB (TCB waiting list included) for TC n, where n is the index of the register in the array.
RESERVED	31:20	0x0	RSV	Reserved.

38.39.2.11.8 DCB Transmit Command Pipe Monitor Control - PRTDCB_TCPMC (0x000A21A0; RW)

Field	Bit(s)	Init.	Type	Description
CPM	12:0	0x098	RW	**Command Pipe Monitor** Depth of the per-Port Monitor applied over the Tx Command Pipe. Expressed in commands units.
LLTC	20:13	0x0	RW	**Low Latency TC** 0b = TC is bulk. 1b = TC is low latency.
RESERVED2	31:21	0x0	RSV	Reserved.

38.39.2.11.9 DCB Transmit ETS Control for TCB - PRTDCB_TETSC_TCB (0x000AE060; RW)

Field	Bit(s)	Init.	Type	Description
RESERVED	7:0	0x0	RSV	Reserved.
LLTC	15:8	0x0	RW	**Low Latency TC** 8-bit wide bitmap with a 1-bit entry per each TC. Each entry controls whether the TC is considered to have low latency needs for the transmit path. 0b = TC is defined to be a Bulk TC for transmit. 1b = TC is defined to be a Low Latency TC for transmit.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.11.10 DCB Receive ETS Per TC Control - PRTDCB_RETSTCC[n] (0x00122180 + 0x20*n, n=0...7; RW)

One register per TC. Register index corresponds to TCID.

Field	Bit(s)	Init.	Type	Description
RESERVED	29:0	0x0	RSV	Reserved.
UPINTC_MODE	30	0b	RW	Rx-ETS operating mode when several UPs are attached to TC n, where n is the register index in the array. 0b = Strict Priority (SP) mode. 1b = Round Robin (RR) mode.
ETSTC	31	0b	RW	**ETS TC** Controls the use of ETS as the Transmit Selection Algorithm (TSA) in Rx for TC n, where n is the register index in the array. 0b = TC n uses a strict priority or other TSA in Rx. 1b = TC n uses ETS scheme in Rx.

38.39.2.11.11DCB Receive Per Port Pipe Monitor Control - PRTDCB_RPPMC (0x001223A0; RW)

Field	Bit(s)	Init.	Type	Description
RESERVED	15:0	0x0	RSV	Reserved.
RX_FIFO_SIZE	23:16	0x08	RW	**Rx Command FIFO Size** The number of Rx commands per TC that can be accumulated in RCB before sending back pressure to RCU.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.11.12DCB Receive ETS Control - PRTDCB_RETSC (0x001223E0; RW)

Field	Bit(s)	Init.	Type	Description
RESERVED	0	0b	RSV	Reserved.
NON_ETS_MODE	1	0b	RW	**Rx Non-ETS Operating Mode** 0b = Strict Priority (SP) mode. 1b = Round Robin (RR) mode.
RESERVED	7:2	0x0	RSV	Reserved.
LLTC	15:8	0x0	RW	**Low Latency TC** 8-bit wide bitmap with a 1-bit entry per each TC. Each entry controls whether the TC is considered to have low latency needs for the receive path. 0b = TC is defined to be a Bulk TC for receive. 1b = TC is defined to be a Low Latency TC for receive.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.11.13DCB Receive Per UP PFC Timer Indication - GLDCB_RUPTI (0x00122618; RO)

Field	Bit(s)	Init.	Type	Description
PFCTIMEOUT_UP	31:0	0x0	RW1C	**PFC Time-Out UPs** Bitmap where a bit set to 1b means that the PFC timer corresponding to the port/UP has timed out. Bits[0:7] = Port 0, UP 0 to 7. Bits[8:15] = Port 1, UP 0 to 7. Bits[23:16] = Port 2, UP 0 to 7. Bits[24:31] = Port 3, UP 0 to 7. Writing a bit with 1b restarts the corresponding PFC timer.

38.39.2.11.14DCB TC to PFC Mapping - PRTDCB_TC2PFC (0x001C0980; RW)

Field	Bit(s)	Init.	Type	Description
TC2PFC	7:0	0x0	RW	**TC to PFC** Bitmap that controls the use of Priority Flow Control (PFC) per each TC. Bit n set to: 0b = The device does not issue PFC pause frames with bits set to 1b in the priority_enable_vector for the UPs attached to that TC. It does not react to bits set to 1b for the UPs attached to that TC in the priority_enable_vector of a received PFC pause frame. The TC is referred as a drop UP. 1b = TC n uses PFC in Rx and Tx. The TC is referred as a no-drop TC.
RESERVED	31:8	0x0	RSV	Reserved.

38.39.2.11.15DCB Receive UP to TC Mapping for RCB - PRTDCB_RUP2TC (0x001C09A0; RW)

24-bit wide bitmap with 3-bit entries per each UP. Each entry controls the mapping of a UP to a 3-bit TC index in receive. Higher TC index means higher priority of the TC. The same mapping is used for packets received from the wires and for those looped back internally. It defines on the account of which TC a packet is stored in the Rx packet buffer, and which UPs bits are set in the PFC XOFF/XON frames issued to the link partner when the filling state of a TC requires it. Default mapping maps all UPs to TC0.

Field	Bit(s)	Init.	Type	Description
UP0TC	2:0	000b	RW	TC index to which UP 0 is mapped.
UP1TC	5:3	000b	RW	TC index to which UP 1 is mapped.
UP2TC	8:6	000b	RW	TC index to which UP 2 is mapped.
UP3TC	11:9	000b	RW	TC index to which UP 3 is mapped.
UP4TC	14:12	000b	RW	TC index to which UP 4 is mapped.
UP5TC	17:15	000b	RW	TC index to which UP 5 is mapped.
UP6TC	20:18	000b	RW	TC index to which UP 6 is mapped.
UP7TC	23:21	000b	RW	TC index to which UP 7 is mapped.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.11.16DCB Receive UP in PPRS - PRTDCB_RUP (0x001C0B00; RW)

Field	Bit(s)	Init.	Type	Description
NOVLANUP	2:0	000b	RW	Assigns a default UP value to untagged incoming packets. The default UP is not inserted in the packet itself, but it controls in which a linked list of RPB untagged packets are stored. UP 0 is the default.
RESERVED	31:3	0x0	RSV	Reserved.

38.39.2.11.17MAC Flow Control Register - PRTDCB_MFLCN (0x001E2400; RW)

Field	Bit(s)	Init.	Type	Description
PMCF	0	0b	RW	**Pass MAC Control Frames** Filter out unrecognized pause (flow control opcode doesn't match) and other control frames 0b = Filter unrecognized pause frames. 1b = Pass/forward unrecognized pause frames.
DPF	1	0b	RW	**Discard Pause Frame** 0b = Pause frames are sent to the host. 1b = Pause frames are discarded when *RFCE* or *RPFCM* is set to 1b. Setting this bit to 1b has no effect otherwise (if both *RFCE* and *RPFCM* are set to 0b).
RPFCM	2	0b	RW	**Receive PFC Mode** Indicates that the device responds to the reception of PFC packets. If auto-negotiation is enabled, this bit should be set by software to the negotiated flow control value. This bit is set as a logical OR over the *RPFCE*[7:0] bitmap. It is useful to control forwarding of PFC frames to host, if required. ***Notes:*** 1. Receive PFC and receive link flow control are mutually exclusive, and designers should not configure both of them to be enabled at the same time. 2. This bit should not be set if Bit[3] is set.

Field	Bit(s)	Init.	Type	Description
RFCE	3	0b	RW	**Receive Link Flow Control Enable** Indicates that the device responds to the reception of link flow control packets. If auto-negotiation is enabled, this bit should be set by software to the negotiated flow control value. ***Note:*** This bit should not be set if Bit[2] is set.
RPFCE	11:4	0x0	RW	**Receive PFC Enable bitmap** When bit n is: 0b = PFC indications received for UPn are ignored. 1b = Upon reception of PFC packets for UPn, the device stops transmit over the TC to which UPn is mapped in Tx.
RESERVED	31:12	0x0	RSV	Reserved.

38.39.2.11.18Transmit Flow Control Status - PRTDCB_TFCS (0x001E4560; RO)

Field	Bit(s)	Init.	Type	Description
TXOFF	0	0b	RO	**Transmission Paused** Pause state indication of the transmit function when symmetrical link flow control is enabled.
RESERVED	7:1	0x0	RSV	Reserved.
TXOFF0	8	0b	RO	**TC 0 Transmission Paused** Pause state indication of TC 0 when PFC is enabled.
TXOFF1	9	0b	RO	**TC 1 Transmission Paused** Pause state indication of TC 1 when PFC is enabled.
TXOFF2	10	0b	RO	**TC 2 Transmission Paused** Pause state indication of TC 2 when PFC is enabled.
TXOFF3	11	0b	RO	**TC 3 Transmission Paused** Pause state indication of TC 3 when PFC is enabled.
TXOFF4	12	0b	RO	**TC 4 Transmission Paused** Pause state indication of TC 4 when PFC is enabled.
TXOFF5	13	0b	RO	**TC 5 Transmission Paused** Pause state indication of TC 5 when PFC is enabled.
TXOFF6	14	0b	RO	**TC 6 Transmission Paused** Pause state indication of TC 6 when PFC is enabled.
TXOFF7	15	0b	RO	**TC 7 Transmission Paused** Pause state indication of TC 7 when PFC is enabled.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.11.19Flow Control Transmit Timer Value n - PRTDCB_FCTTVN[n] (0x001E4580 + 0x20*n, n=0...3; RW)

Field	Bit(s)	Init.	Type	Description
TTV_2N	15:0	0xFFFF	RW	**Transmit Timer Value 2n** Timer value included in XOFF frames as timer (2n). The same value is set to UPs attached to the same TC, as defined in the PRTDCB_RUP2TC register (Section 38.39.2.11.15). For legacy 802.3x flow control packets, TTV0 is the only timer that is used.
TTV_2N_P1	31:16	0xFFFF	RW	**Transmit Timer Value 2n+1** Timer value included in XOFF frames as timer (2n+1). The same value is set to UPs attached to the same TC, as defined in PRTDCB_RUP2TC register (Section 38.39.2.11.15).

38.39.2.11.20 Flow Control Refresh Threshold Value - PRTDCB_FCRTV (0x001E4600; RW)

Field	Bit(s)	Init.	Type	Description
FC_REFRESH_TH	15:0	0x7FFF	RW	**Flow Control Refresh Threshold** Used to calculate the actual refresh period for sending the next pause frame if conditions for a pause state are still valid (buffer fullness above low threshold value). The formula for the refresh period for user priority N is: FCTTV[N/2].*TTV*[Nmod2] - FCRTV.*FC_REFRESH_TH*
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.11.21 Flow Control Configuration - PRTDCB_FCCFG (0x001E4640; RW)

Field	Bit(s)	Init.	Type	Description
RESERVED	2:0	000b	RSV	Reserved.
TFCE	4:3	00b	RW	**Transmit Flow Control Enable** Indicate that the 10 GbE controller transmits flow control packets (XON/XOFF frames) based on receive fullness. If auto-negotiation is enabled this bit should be set by software to the negotiated flow control value. 00b = Transmit flow control disabled. 01b = Link flow control enabled. 10b = PFC enabled. 11b = Reserved.
RESERVED	31:5	0x0	RSV	Reserved.

38.39.2.11.22 DCB Transmit PFC Timer Status - PRTDCB_TPFCTS[n] (0x001E4660 + 0x20*n, n=0...7; RW)

One register per TC. Register index corresponds to TCID.

Field	Bit(s)	Init.	Type	Description
PFCTIMER	13:0	0x0	RW	**PFC Timer** Current value of the PFC timer of TC n in Tx, where n is the register index in the array. The amount is expressed in milliseconds. The timer saturates to 0x3FFF. Writing to this field has the effect of loading a new current timer value, which can be useful for diagnostic purposes.
RESERVED	31:14	0x0	RSV	Reserved.

38.39.2.12 PF - Receive Packet Buffer Registers

38.39.2.12.1 RPB Dedicated Pool High Watermark - PRTRPB_DHW[n] (0x000AC100 + 0x20*n, n=0...7; RW)

One register per TC. Register index corresponds to TCID.

Field	Bit(s)	Init.	Type	Description
DHW_TCN	19:0	0x0	RW	**Dedicated Pool High Watermark For TC n** Expressed in bytes.
RESERVED	31:20	0x0	RSV	Reserved.

38.39.2.12.2 RPB Dedicated Pool Low Watermark - PRTRPB_DLW[n] (0x000AC220 + 0x20*n, n=0...7; RW)

One register per TC. Register index corresponds to TCID.

Field	Bit(s)	Init.	Type	Description
DLW_TCN	19:0	0x0	RW	**Dedicated Pool Low Watermark For TC n** Expressed in bytes.
RESERVED	31:20	0x0	RSV	Reserved.

38.39.2.12.3 RPB Dedicated Pool Size - PRTRPB_DPS[n] (0x000AC320 + 0x20*n, n=0...7; RW)

One register per TC. Register index corresponds to TCID.

Field	Bit(s)	Init.	Type	Description
DPS_TCN	19:0	0x0	RW	**Dedicated Pool Size For TC n** Expressed in bytes.
RESERVED	31:20	0x0	RSV	Reserved.

38.39.2.12.4 RPB Shared Pool High Threshold - PRTRPB_SHT[n] (0x000AC480 + 0x20*n, n=0...7; RW)

One register per TC. Register index corresponds to TCID.

Field	Bit(s)	Init.	Type	Description
SHT_TCN	19:0	0x3C800	RW	**Shared Pool High Threshold For TC n** Expressed in bytes.
RESERVED	31:20	0x0	RSV	Reserved.

38.39.2.12.5 RPB Shared Pool High Watermark - PRTRPB_SHW (0x000AC580; RW)

Field	Bit(s)	Init.	Type	Description
SHW	19:0	0x3C800	RW	**Shared Pool High Watermark** Expressed in bytes.
RESERVED	31:20	0x0	RSV	Reserved.

38.39.2.12.6 RPB Shared Pool Low Threshold - PRTRPB_SLT[n] (0x000AC5A0 + 0x20*n, n=0...7; RW)

One register per TC. Register index corresponds to TCID.

Field	Bit(s)	Init.	Type	Description
SLT_TCN	19:0	0x0	RW	**Shared Pool Low Threshold For TC n** Expressed in bytes.
RESERVED	31:20	0x0	RSV	Reserved.

38.39.2.12.7 RPB Shared Pool Low Watermark - PRTRPB_SLW (0x000AC6A0; RW)

Field	Bit(s)	Init.	Type	Description
SLW	19:0	0x0	RW	**Shared Pool Low Watermark** Expressed in bytes.
RESERVED	31:20	0x0	RSV	Reserved.

38.39.2.12.8 RPB Shared Pool Size - PRTRPB_SPS (0x000AC7C0; RW)

Field	Bit(s)	Init.	Type	Description
SPS	19:0	0x3C800	RW	**Shared Pool Size** Number of bytes allocated to the shared pool of the port.
RESERVED	31:20	0x0	RSV	Reserved.

38.39.2.12.9 RPB Dedicated Pool Size for Single Shared Buffer State - GLRPB_DPSS (0x000AC828; RW)

Field	Bit(s)	Init.	Type	Description
DPS_TCN	19:0	0x0	RW	**Dedicated Pool Size For TC n** Dedicated pool size for the single shared buffer state, for all TCs of all ports. Expressed in bytes.
RESERVED	31:20	0x0	RSV	Reserved.

38.39.2.12.10 RPB Global High Watermark - GLRPB_GHW (0x000AC830; RW)

Field	Bit(s)	Init.	Type	Description
GHW	19:0	0xF2000	RW	**Global High Watermark** Expressed in bytes.
RESERVED	31:20	0x0	RSV	Reserved.

38.39.2.12.11 RPB Global Low Watermark - GLRPB_GLW (0x000AC834; RW)

Field	Bit(s)	Init.	Type	Description
GLW	19:0	0x0	RW	**Global Low Watermark** Expressed in bytes.
RESERVED	31:20	0x0	RSV	Reserved.

38.39.2.12.12 RPB Packet High Watermark - GLRPB_PHW (0x000AC844; RW)

Field	Bit(s)	Init.	Type	Description
PHW	19:0	0x1246	RW	**Packet High Watermark** Relative to the total number of packets stored in the RPB.
RESERVED	31:20	0x0	RSV	Reserved.

38.39.2.12.13 RPB Packet Low Watermark - GLRPB_PLW (0x000AC848; RW)

Field	Bit(s)	Init.	Type	Description
PLW	19:0	0x0846	RW	**Packet Low Watermark** Relative to the total number of packets stored in the RPB.
RESERVED	31:20	0x0	RSV	Reserved.

38.39.2.13 PF - Host Memory Cache Registers

38.39.2.13.1 PEOC Cache Attributes - GLPEOC_CACHESIZE (0x000A60E8; RO)

Field	Bit(s)	Init.	Type	Description
WORD_SIZE	7:0	0x40	RO	**Word Size** The cache line size in bytes.
SETS	19:8	0x400	RO	**Sets** The number of cache sets.
WAYS	23:20	0x8	RO	**Ways** The number of cache ways.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.2 PBLOC Cache Attributes - GLPBLOC_CACHESIZE (0x000A80BC; RO)

Field	Bit(s)	Init.	Type	Description
WORD_SIZE	7:0	0x40	RO	**Word Size** The cache line size in bytes.
SETS	19:8	0x400	RO	**Sets** The number of cache sets.
WAYS	23:20	0x8	RO	**Ways** The number of cache ways.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.3 FOC Cache Attributes - GLFOC_CACHESIZE (0x000AA0DC; RO)

Field	Bit(s)	Init.	Type	Description
WORD_SIZE	7:0	0x40	RO	**Word Size** The cache line size in bytes.
SETS	19:8	0x80	RO	**Sets** The number of cache sets.
WAYS	23:20	0x8	RO	**Ways** The number of cache ways.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.4 Private Memory Space Segment Descriptor Command - PFHMC_SDCMD (0x000C0000; RW)

This register is used to access the HMC's segment table. The HMC's segment table is partitioned per PCI function. However, each PCIe PF is able access any segment table entry.

For read operations, PFHMC_SDCMD must be written with *PMSDWR* set to 0b, and *PMSDIDX* must be set to the segment table index to be read. After the write operation completes, PFHMC_SDDATALOW and PFHMC_SDDATAHIGH registers might be read to retrieve the segment descriptor contents (Section 38.39.2.13.5 and Section 38.39.2.13.6, respectively).

For write operations, the PFHMC_SDDATALOW and PFHMC_SDDATAHIGH registers must be written before writing PFHMC_SDCMD, with *PMSDWR* set to 1b and *PMSDIDX* set to the segment table index to be written.

Field	Bit(s)	Init.	Type	Description
PMSDIDX	11:0	0x0	RW	**Private Memory Space Descriptor Index** Relative index of the HMC segment descriptor to be read or written. The actual absolute index to be used to access the segment table is (*PMSDBASE* + *PMSDIDX*), where *PMSDBASE* is from the *PMSDBASE* field of the GLHMC_SDPART or GLHMC_PFPESDPART register associated with this function (Section 38.39.2.13.10 and Section 38.39.2.13.11, respectively). On write operations, if (*PMSDIDX* ≥ *PMSDSIZE*), the write is dropped, where *PMSDSIZE* is from the *PMSDSIZE* field of the GLHMC_SDPART or GLHMC_PFPESDPART register associated with this function. The *PMSDPARTSEL* bit of this register determines which SD partition register is selected for the absolute SD index calculation and for the check that *PMSDIDX* is within the range allowed by this function.
RESERVED	14:12	000b	RSV	Reserved.
PMSDPARTSEL	15	0b	RW	**Private Memory Space Descriptor Partition Select** 0b = The GLHMC_SDPART register is used for calculating the absolute SD Index to access within the segment table, and for checking whether *PMSDIDX* is within this function's allocated range. 1b = The GLHMC_PFPESDPART register is used for calculating the absolute SD Index to access within the segment table, and for checking whether *PMSDIDX* is within this function's allocated range.
RESERVED	30:16	0x0	RSV	Reserved.
PMSDWR	31	0b	RW	**Private Memory Space Descriptor Write or Read** 0b = Read operations. 1b = Write operations.

38.39.2.13.5 Private Memory Space Segment Descriptor Data Low - PFHMC_SDDATALOW (0x000C0100; RW)

This register is used in conjunction with PFHMC_SDCMD and PFHMC_SDDATAHIGH to access the HMC's segment table (Section 38.39.2.13.4 and Section 38.39.2.13.6, respectively).

Field	Bit(s)	Init.	Type	Description
PMSDVALID	0	0b	RW	**Private Memory Space Descriptor Valid** Valid bit of an HMC segment descriptor table entry.
PMSDTYPE	1	0b	RW	**Private Memory Space Descriptor Type** 0b = The segment descriptor is paged (the SD points to a the physical address of a host memory page that contains an array of page descriptors). 1b = The segment descriptor directly points the physical address of a physically contiguous 2 MB memory region.
PMSDBPCOUNT	11:2	0x0	RW	**Private Memory Space Descriptor Backing Page Count** Backing page count of an HMC segment descriptor table entry. Every SD entry in a given FPM space must be set to 512 except the last SD. The last SD can have a value from 1 to 512. This field is used to calculate the end of the FPM space associated with a Segment Descriptor without having to read the valid bit for each individual PD entry.
PMSDDATALOW	31:12	0x0	RW	**Private Memory Space Descriptor Data Low** Bits[31:12] of an HMC segment descriptor table entry.

38.39.2.13.6 Private Memory Space Segment Descriptor Data High - PFHMC_SDDATAHIGH (0x000C0200; RW)

This register is used in conjunction with PFHMC_SDCMD and PFHMC_SDDATALOW to access the HMC's segment table (Section 38.39.2.13.4 and Section 38.39.2.13.5, respectively).

Field	Bit(s)	Init.	Type	Description
PMSDDATAHIGH	31:0	0x0	RW	**Private Memory Space Descriptor Data High** Most significant 32 bits of a segment descriptor.

38.39.2.13.7 Private Memory Space Page Descriptor Invalidate - PFHMC_PDINV (0x000C0300; RW)

This register is used to invalidate cached HMC page descriptors that have been set to the invalid state by software.

Field	Bit(s)	Init.	Type	Description
PMSDIDX	11:0	0x0	RW	**Private Memory Space Descriptor Index** Relative index of the HMC segment descriptor associated with the HMC page descriptor that is to be invalidated. For PFs, the actual index to be used to access the segment table is (*PMSDBASE* + *PMSDIDX*), where *PMSDBASE* is from the *PMSDBASE* field of the GLHMC_SDPART or GLHMC_PFPESDPART register associated with this function (Section 38.39.2.13.10 and Section 38.39.2.13.11, respectively). If (*PMSDIDX* ≥ *PMSDSIZE*), the invalidate request is dropped, where *PMSDSIZE* is from the *PMSDSIZE* field of the GLHMC_SDPART or GLHMC_PFPESDPART register associated with this function. The *PMSDPARTSEL* bit of this register determines which SD partition register is selected for the absolute SD index calculation and for the check that *PMSDIDX* is within the range allowed by this function. For PE enabled VFs, the actual index to be used to invalidate a segment table entry is (*PMSDBASE* + *PMSDIDX*), where *PMSDBASE* is from the GLHMC_VFSDPART register associated with this function (Section 38.39.2.13.73). For VFs, if (*PMSDIDX* ≥ GLHMC_VFSDPART.*PMSDSIZE*), the invalidate request is dropped. The *PMSDPARTSEL* bit is ignored in the GLHMC_VFPDINV registers.
RESERVED	14:12	000b	RSV	Reserved.
PMSDPARTSEL	15	0b	RW	**Private Memory Space Descriptor Partition Select** 0b = The GLHMC_SDPART register is used for calculating the absolute SD Index to invalidate within the segment table, and for checking whether or not *PMSDIDX* is within this function's allocated range. 1b = The GLHMC_PFPESDPART register is used for calculating the absolute SD index to invalidate within the segment table, and for checking whether or not *PMSDIDX* is within this function's allocated range. This field is ignored in the GLHMC_VFPDINV versions of this register.
PMPDIDX	24:16	0x0	RW	**Private Memory Page Descriptor Index** Index of the page descriptor within the page descriptor page indicated by *PMSDIDX*.
RESERVED	31:25	0x0	RSV	Reserved.

38.39.2.13.8 Host Memory Cache Error Information Register - PFHMC_ERRORINFO (0x000C0400; RW)

This register reports the errors detected by the HMC. Errors reported through this register might also trigger interrupts.

Field	Bit(s)	Init.	Type	Description
PMF_INDEX	4:0	0x0	RW	**Private Memory Function Index** Reports the HMC private memory function associated with the error. Writes to this field are ignored.
RESERVED	6:5	00b	RSV	Reserved.
PMF_ISVF	7	0b	RW	**Private Memory Function Is VF** 0b = The private memory function reported in *PMF_INDEX* is associated with a PF. 1b = The private memory function reported in *PMF_INDEX* is associated with a PE enabled VF. Writes to this field are ignored.
HMC_ERROR_TYPE	11:8	0x0	RW	**HMC Error Type** Reports the error type detected by the HMC. The values are: 0 = Private memory function is not valid. 1 = Invalid private memory function index for a PE enabled VF. 2 = Invalid PF for a PE enabled VF. 3 = Invalid LAN queue index. 4 = Object index from transaction was larger than the value specified in the object's GLHMC_*CNT register. 5 = Private memory address extends beyond the limits of the segment descriptors assigned to the PCIe function. 6 = Segment descriptor Invalid. 7 = Segment descriptor too small (only applies to direct mapped SDs). 8 = Page Descriptor Invalid. 9 = Received Unsupported Request (UR) completion from PCIe read of object. 10 = LAN queue is not valid. 11 = Invalid object type was detected. 12 = Reserved. The PFHMC_ERRORDATA register (Section 38.39.2.13.9) can be read to determine the following: • The LAN Queue index. • The HMC object index associated with error types 4 and 12. • The HMC function relative *SD_Index* and *PD_Index* associated with error types 5 through 9.
RESERVED	15:12	0x0	RSV	Reserved.

Field	Bit(s)	Init.	Type	Description
HMC_OBJECT_TYPE	20:16	0x0	RW	**HMC Object Type** Specifies the object type associated with the error. The encodings for the object type are as follows: 00000b = QP_CNTXT. 00001b = ARP_TBL_ENTRY. 00010b = TXFIFO. 00011b = IRRQ. 00100b = MRTE. 00101b = PBLE. 00110b = CQ_CNTXT. 00111b = SRQ_CNTXT. 01000b = APBVT_INUSE. 01001b = FSI_ADR_VCTR. 01010b = FSI_MCAST_GRP. 01011b = XF_FL. 01100b = Q1_FL. 01101b = TIMER. 10000b = LAN_TXQ_CNTXT. 10001b = LAN_RXQ_CNTXT. 10010b = Reserved. 10011b = Reserved. 10110b = QUAD_HTE. 11001b = PD. All other values are reserved.
RESERVED	30:21	0x0	RSV	Reserved.
ERROR_DETECTED	31	0b	RW	**Error Detected** This field is set to 1b when a new error is detected by the HMC. No subsequent errors are recorded until this field is written with a value of 0b. A write of a 0b to this register clears the error and allows a subsequent error to be reported. Writes of 1b to this field are ignored.

38.39.2.13.9 Host Memory Cache Error Data Register - PFHMC_ERRORDATA (0x000C0500; RO)

This register reports the private memory address or HMC object index related to an error detected by the HMC.

Field	Bit(s)	Init.	Type	Description
HMC_ERROR_DATA	29:0	0x0	RO	**HMC Error Data** Reports either the HMC function relative *SD_Index*, *PD_Index*, LAN queue index or HMC object index associated with the error reported in the PFHMC_ERRORINFO register (Section 38.39.2.13.8). • When PFHMC_ERRORINFO.*HMC_ERROR_TYPE* is 3, HMC_ERROR_DATA[27:0] reports the LAN queue index and HMC_ERROR_DATA[29:28] should be zero. • When PFHMC_ERRORINFO.*HMC_ERROR_TYPE* is 4 or 12, HMC_ERROR_DATA[27:0] reports the object index associated with the error, and HMC_ERROR_DATA[29:28] should be zero. • When PFHMC_ERRORINFO.*HMC_ERROR_TYPE* is 5 through 9, HMC_ERROR_DATA[29:9] reports the HMC function relative *SD_Index* and *PD_Index* associated with the error detected by the HMC. HMC_ERROR_DATA[29:18] is set to HMC function relative *SD_Index* for the affected HMC function, HMC_ERROR_DATA[17:9] is set to the *PD_Index*. HMC_ERROR_DATA[8:0] are reserved for error types 5 through 9. • When PFHMC_ERRORINFO.*HMC_ERROR_TYPE* is 0 through 2, or 10 through 11, HMC_ERROR_DATA[29:0] is not valid, and should be zero.
RESERVED	31:30	00b	RSV	Reserved.

38.39.2.13.10 Private Memory Segment Table Partitioning Registers - GLHMC_SDPART[n] (0x000C0800 + 0x4*n, n=0...15; RO)

This register is used to partition the shared Host Memory Cache segment table.

Field	Bit(s)	Init.	Type	Description
PMSDBASE	11:0	0x0	RW	**Private Memory SD Base** Base segment table index for the function n.
RESERVED	15:12	0x0	RSV	Reserved.
PMSDSIZE	28:16	0x0	RW	**Private Memory SD Size** Number of valid segment table entries for the function n.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.11 Private Memory Segment Table Partitioning Registers - GLHMC_PFPESDPART[n] (0x000C0880 + 0x4*n, n=0...15; RO)

This register is used to partition the shared HMC segment table for the PE resources associated with PFs.

Note: This register must be read only in the PF CSR space.

Field	Bit(s)	Init.	Type	Description
PMSDBASE	11:0	0x0	RW	**Private Memory SD Base** Base segment table index for the function n.
RESERVED	15:12	0x0	RSV	Reserved.
PMSDSIZE	28:16	0x0	RW	**Private Memory SD Size** Number of valid segment table entries for the function n.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.12 Private Memory LAN Tx Object Size - GLHMC_LANTXOBJSZ (0x000C2004; RO)

Field	Bit(s)	Init.	Type	Description
PMLANTXOBJSZ	3:0	0x7	RO	**Private Memory LAN Transmit Object Size** Used to calculate the amount of memory to allocate for HMC LAN Tx queue objects. The value in this register is decoded such that the value = log2(ObjSize). 0x7 = 128 bytes.
RESERVED	31:4	0x0	RSV	Reserved.

38.39.2.13.13 Private Memory LAN Queue Maximum - GLHMC_LANQMAX (0x000C2008; RO)

Field	Bit(s)	Init.	Type	Description
PMLANQMAX	10:0	0x600	RO	**Private Memory LAN Queue Maximum** Reports the maximum number of LAN transmit or receive queue resources supported by the HMC. A given PCI function might be restricted to a smaller value by the NVRAM settings and the total number of enabled PCI functions.
RESERVED	31:11	0x0	RSV	Reserved.

38.39.2.13.14 Private Memory LAN Rx Object Size - GLHMC_LANRXOBJSZ (0x000C200C; RO)

Field	Bit(s)	Init.	Type	Description
PMLANRXOBJSZ	3:0	0x5	RO	**Private Memory LAN Receive Object Size** Used to calculate the amount of memory to allocate for HMC LAN Rx queue objects. The value in this register is decoded such that the value = log2(ObjSize). 0x5 = 32 bytes.
RESERVED	31:4	0x0	RSV	Reserved.

38.39.2.13.15 Private Memory PE QP Object Size - GLHMC_PEQPOBJSZ (0x000C201C; RO)

Field	Bit(s)	Init.	Type	Description
PMPEQPOBJSZ	3:0	0x9	RO	**Private Memory PE Queue Pair Object Size** Used to calculate the amount of memory to allocate for HMC PE queue pair objects. 0x9 = 512 bytes.
RESERVED	31:4	0x0	RSV	Reserved.

38.39.2.13.16 Private Memory PE CQ Object Size - GLHMC_PECQOBJSZ (0x000C2020; RO)

Field	Bit(s)	Init.	Type	Description
PMPECQOBJSZ	3:0	0x6	RO	**Private Memory PE Completion Queue Object Size** Used to calculate the amount of memory to allocate for HMC PE completion queue objects. 0x6 = 64 bytes
RESERVED	31:4	0x0	RSV	Reserved.

38.39.2.13.17 Private Memory PE S-RQ Object Size - GLHMC_PESRQOBJSZ (0x000C2024; RO)

Field	Bit(s)	Init.	Type	Description
PMPESRQOBJSZ	3:0	0x6	RO	**Private Memory PE Shared Receive Queue Object Size** Used to calculate the amount of memory to allocate for HMC Shared Receive Queue (S-RQ) objects. 0x6 = 64 bytes.
RESERVED	31:4	0x0	RO	Reserved.

38.39.2.13.18 Private Memory Protocol Engine SRQ Max - GLHMC_PESRQMAX (0x000C2028; RO)

Field	Bit(s)	Init.	Type	Description
PMPESRQMAX	15:0	0x8000	RO	**Private Memory PE Shared Receive Queue Maximum** Reports the maximum number of S-RQs supported by the HMC.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.13.19 Private Memory PE Hash Table Entry Object Size - GLHMC_PEHTEOBJSZ (0x000C202C; RO)

Field	Bit(s)	Init.	Type	Description
PMPEHTEOBJSZ	3:0	0x6	RO	**Private Memory PE Hash Table Entry Object Size** Used to calculate the amount of memory to allocate for HMC PE hash table entry objects. PE hash tables can be configured to be IPv4 or IPv6 per function. 0x6 = 64 bytes.
RESERVED	31:4	0x0	RSV	Reserved.

38.39.2.13.20 Private Memory Protocol Engine Hash Entry Max - GLHMC_PEHTMAX (0x000C2030; RO)

Field	Bit(s)	Init.	Type	Description
PMPEHTMAX	20:0	0x140000	RO	**Private Memory PE Hash Table Maximum** Reports the maximum number of Hash Table Entries supported by the HMC.
RESERVED	31:21	0x0	RSV	Reserved.

38.39.2.13.21 Private Memory PE ARP Table Entry Object Size - GLHMC_PEARPOBJSZ (0x000C2034; RO)

Field	Bit(s)	Init.	Type	Description
PMPEARPOBJSZ	2:0	100b	RO	**Private Memory PE ARP Object Size** Used to calculate the amount of memory to allocate for HMC ARP table entry objects. 100b = 16 bytes.
RESERVED	31:3	0x0	RSV	Reserved.

38.39.2.13.22 Private Memory Protocol Engine ARP Table Entry Max - GLHMC_PEARPMAX (0x000C2038; RO)

Field	Bit(s)	Init.	Type	Description
PMPEARPMAX	16:0	0x10000	RO	**Private Memory PE ARP Maximum** Reports the maximum number of ARP table entries supported by the HMC.
RESERVED	31:17	0x0	RSV	Reserved.

38.39.2.13.23 Private Memory PE Memory Region Table Entry Object Size - GLHMC_PEMROBJSZ (0x000C203C; RO)

Field	Bit(s)	Init.	Type	Description
PMPEMROBJSZ	3:0	0x5	RO	**Private Memory PE Memory Region Object Size** Used to calculate the amount of memory to allocate for HMC memory region table entry objects. 0x5 = 32 bytes
RESERVED	31:4	0x0	RSV	Reserved.

38.39.2.13.24 Private Memory Protocol Engine Memory Registration Max - GLHMC_PEMRMAX (0x000C2040; RO)

Field	Bit(s)	Init.	Type	Description
PMPEMRMAX	22:0	0x400000	RO	**Private Memory PE Memory Registration Maximum** Reports the maximum number of memory registration table entries supported by the HMC.
RESERVED	31:23	0x0	RSV	Reserved.

38.39.2.13.25 Private Memory PE Transmit FIFO Object Size - GLHMC_PEXFOBJSZ (0x000C2044; RO)

Field	Bit(s)	Init.	Type	Description
PMPEXFOBJSZ	3:0	0x5	RO	**Private Memory PE Transmit FIFO Object Size** Used to calculate the amount of memory to allocate for HMC transmit FIFO objects. 0x5 = 32 bytes.
RESERVED	31:4	0x0	RO	Reserved.

38.39.2.13.26 Private Memory Protocol Engine Transmit FIFO Entry mMax - GLHMC_PEXFMAX (0x000C2048; RO)

Field	Bit(s)	Init.	Type	Description
PMPEXFMAX	25:0	0x2000000	RO	**Private Memory PE Transmit FIFO Maximum** Reports the maximum number of transmit FIFO entries supported by the HMC.
RESERVED	31:26	0x0	RSV	Reserved.

38.39.2.13.27 Private Memory Protocol Engine Transmit FIFO Free List xax - GLHMC_PEXFFLMAX (0x000C204C; RO)

Field	Bit(s)	Init.	Type	Description
PMPEXFFLMAX	24:0	0x1000000	RO	**Private Memory PE Transmit FIFO Free List Maximum** Reports the maximum number of transmit FIFO free list entries supported by the HMC.
RESERVED	31:25	0x0	RSV	Reserved.

38.39.2.13.28 Private Memory PE IRRQ Object Size - GLHMC_PEQ1OBJSZ (0x000C2050; RO)

Field	Bit(s)	Init.	Type	Description
PMPEQ1OBJSZ	3:0	0x6	RO	**Private Memory PE Q1 Object Size** Used to calculate the amount of memory to allocate for HMC inbound RDMA read (Q1) objects. 0x6 = 64 bytes.
RESERVED	31:4	0x0	RSV	Reserved.

38.39.2.13.29 Private Memory Protocol Engine Q1 Max - GLHMC_PEQ1MAX (0x000C2054; RO)

Field	Bit(s)	Init.	Type	Description
PMPEQ1MAX	25:0	0x2000000	RO	**Private Memory PE Q1 Maximum** Reports the maximum number of inbound RDMA read queue (Q1) entries supported by the HMC.
RESERVED	31:26	0x0	RSV	Reserved.

38.39.2.13.30 Private Memory Protocol Engine Q1 Free List Max - GLHMC_PEQ1FLMAX (0x000C2058; RO)

Field	Bit(s)	Init.	Type	Description
PMPEQ1FLMAX	25:0	0x2000000	RO	**Private Memory PE Q1 Free List Maximum** Reports the maximum number of inbound RDMA read queue (Q1) free list entries supported by the HMC.
RESERVED	31:26	0x0	RSV	Reserved.

38.39.2.13.31 Private Memory FSI Multicast Group Object Size - GLHMC_FSIMCOBJSZ (0x000C205C; RO)

Field	Bit(s)	Init.	Type	Description
PMFSIMCOBJSZ	3:0	0x6	RO	**Private Memory FSI Multicast Object Size** Used to calculate the amount of memory to allocate for HMC FSI multicast group objects. 0x6 = 64 bytes.
RESERVED	31:4	0x0	RSV	Reserved.

38.39.2.13.32 Private Memory FSI Multicast Group Max - GLHMC_FSIMCMAX (0x000C2060; RO)

Field	Bit(s)	Init.	Type	Description
PMFSIMCMAX	13:0	0x2000	RO	**Private Memory FSI Multicast Maximum** Reports the maximum number of FSI multicast group entries supported by the HMC.
RESERVED	31:14	0x0	RSV	Reserved.

38.39.2.13.33 Private Memory FSI Address Vector Object Size - GLHMC_FSIAVOBJSZ (0x000C2064; RO)

Field	Bit(s)	Init.	Type	Description
PMFSIAVOBJSZ	3:0	0x5	RO	**Private Memory FSI Address Vector Object Size** Used to calculate the amount of memory to allocate for HMC FSI address vector objects. 0x5 = 32 bytes.
RESERVED	31:4	0x0	RSV	Reserved.

38.39.2.13.34 Private Memory FSI Address Vector Max - GLHMC_FSIAVMAX (0x000C2068; RO)

Field	Bit(s)	Init.	Type	Description
PMFSIAVMAX	16:0	0x10000	RO	**Private Memory FSI Address Vector Maximum** Reports the maximum number of FSI address vectors supported by the HMC.
RESERVED	31:17	0x0	RSV	Reserved.

38.39.2.13.35 Private Memory Protocol Engine Physical Buffer List Max - GLHMC_PEPBLMAX (0x000C206C; RO)

Field	Bit(s)	Init.	Type	Description
PMPEPBLMAX	28:0	0x10000000	RO	**Private Memory PE Physical Buffer List Maximum** Reports the maximum number of physical buffer list entries supported by the HMC.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.36 Private Memory PE Timer Object Size - GLHMC_PETIMEROBJSZ (0x000C2080; RO)

Field	Bit(s)	Init.	Type	Description
PMPETIMEROBJSZ	3:0	0x6	RO	**Private Memory PE Timer Object Size** Used to calculate the amount of memory to allocate for HMC PE timer objects. The value in this register is decoded such that the value = log2(ObjSize). 0x6 = 64 bytes.
RESERVED	31:4	0x0	RSV	Reserved.

38.39.2.13.37 Private Memory PE Timer Object Max - GLHMC_PETIMERMAX (0x000C2084; RO)

Field	Bit(s)	Init.	Type	Description
PMPETIMERMAX	28:0	0x10000000	RO	**Private Memory PE Timer Maximum** Reports the maximum number of PE timer objects supported by the HMC.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.38 Protocol Engine PF First SD - GLHMC_PEPFFIRSTSD (0x000C20E4; RO)

This register is set to the number of HMC segment descriptors consumed by LAN for all PFs. This value is programmed from the NVM.

Field	Bit(s)	Init.	Type	Description
GLHMC_PEPFFIRSTSD	11:0	0x0	RW	**PE PF First SD** First SD that is available to the PE to use.
RESERVED	31:12	0x0	RSV	Reserved.

38.39.2.13.39 Private Memory Protocol Engine Queue Pair Max - GLHMC_DBQPMAX (0x000C20EC; RO)

Field	Bit(s)	Init.	Type	Description
GLHMC_DBQPMAX	18:0	0x40000	RO	**Doorbell Queue Pair Maximum** Reports the maximum number of PE queue pairs supported by the HMC and doorbell array.
RESERVED	31:19	0x0	RSV	Reserved.

38.39.2.13.40 Private Memory Protocol Engine Completion Queue Max - GLHMC_DBCQMAX (0x000C20F0; RO)

Field	Bit(s)	Init.	Type	Description
GLHMC_DBCQMAX	17:0	0x020000	RO	**Doorbell Completion Queue Maximum** Reports the maximum number of PE completion queues supported by the HMC and doorbell array.
RESERVED	31:18	0x0	RSV	Reserved.

38.39.2.13.41 FPM PE QP Base - GLHMC_PEQPBASE[n] (0x000C4000 + 0x4*n, n=0...15; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only.

Field	Bit(s)	Init.	Type	Description
FPMPEQPBASE	23:0	0x0	RW	**Function Private Memory PE Queue Pair Base** Reports the FPM space base address for the PE queue pair objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.42 FPM PE QP Object Count - GLHMC_PEQPCNT[n] (0x000C4100 + 0x4*n, n=0...15; RO)

The associated base register (Section 38.39.2.13.41) is updated after commit FPM values CQP operation is performed to indicate to hardware that all of the FPM size registers have been set properly and the FPM map should be recomputed.

Field	Bit(s)	Init.	Type	Description
FPMPEQPCNT	28:0	0x0	RW	**Function Private Memory PE Queue Pair Count** Used to set the FPM space size for the PE queue pair objects.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.43FPM PE CQ Base - GLHMC_PECQBASE[n] (0x000C4200 + 0x4*n, n=0...15; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only.

Field	Bit(s)	Init.	Type	Description
FPMPECQBASE	23:0	0x0	RW	**Function Private Memory PE Completion Queue Base** Reports the FPM space base address for the PE completion queue objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.44FPM PE CQ Object Count - GLHMC_PECQCNT[n] (0x000C4300 + 0x4*n, n=0...15; RO)

The associated base register (Section 38.39.2.13.43) is updated after the commit FPM values CQP operation is performed to indicate to hardware that all of the FPM size registers have been set properly and the FPM map should be recomputed.

Field	Bit(s)	Init.	Type	Description
FPMPECQCNT	28:0	0x0	RW	**Function Private Memory PE Completion Queue Count** Used to set the FPM space size for the PE completion queue objects.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.45FPM PE Shared RQ Base - GLHMC_PESRQBASE[n] (0x000C4400 + 0x4*n, n=0...15; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only.

Field	Bit(s)	Init.	Type	Description
FPMPESRQBASE	23:0	0x0	RW	**Function Private Memory PE Shared Receive Queue Base** Reports the FPM space base address for the PE shared receive queue objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.46FPM PE Shared RQ Size - GLHMC_PESRQCNT[n] (0x000C4500 + 0x4*n, n=0...15; RO)

The associated base register (Section 38.39.2.13.45) is updated after the commit FPM values CQP operation is performed to inform hardware that all of the FPM size registers have been set properly and the FPM map should be recomputed.

Field	Bit(s)	Init.	Type	Description
FPMPESRQCNT	28:0	0x0	RW	**Function Private Memory PE Shared Receive Queue Count** Used to set the FPM space size for the PE shared receive queue objects.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.47FPM PE Hash Table Entry Base - GLHMC_PEHTEBASE[n] (0x000C4600 + 0x4*n, n=0...15; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only.

Field	Bit(s)	Init.	Type	Description
FPMPEHTEBASE	23:0	0x0	RW	**Function Private Memory PE Hash Table Entry Base** Reports the FPM space base address for the PE hash table entry objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.48FPM PE Hash Table Object Count - GLHMC_PEHTCNT[n] (0x000C4700 + 0x4*n, n=0...15; RO)

The associated base register (Section 38.39.2.13.47) is updated after the commit FPM values CQP operation is performed to indicate to hardware that all of the FPM size registers have been set properly and the FPM map should be recomputed.

Field	Bit(s)	Init.	Type	Description
FPMPEHTCNT	28:0	0x0	RW	**Function Private Memory PE Hash Table Count** Used to set the FPM space size for the PE hash table objects.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.49FPM PE ARP Table Base - GLHMC_PEARPBASE[n] (0x000C4800 + 0x4*n, n=0...15; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only.

Field	Bit(s)	Init.	Type	Description
FPMPEARPBASE	23:0	0x0	RW	**Function Private Memory PE ARP Base** Reports the FPM space base address for the PE ARP table objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.50FPM PE ARP Table Object Count - GLHMC_PEARPCNT[n] (0x000C4900 + 0x4*n, n=0...15; RO)

The associated base register (Section 38.39.2.13.49) is updated after the commit FPM values CQP operation is performed to indicate to hardware that all of the FPM size registers have been set properly and the FPM map should be recomputed.

Field	Bit(s)	Init.	Type	Description
FPMPEARPCNT	28:0	0x0	RW	**Function Private Memory PE ARP Count** Used to set the FPM space size for the PE ARP table objects.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.51 FPM PE APBVT In-Use Base - GLHMC_APBVTINUSEBASE[n] (0x000C4A00 + 0x4*n, n=0...15; RO)

There is a single APBVT in-use object for each PCI function and it is a fixed 8 KB in size. Since there is only a single object and it is a fixed size, there is not a corresponding object size or region size register. This register is updated by hardware when commit FPM values CQP operation is performed.

Field	Bit(s)	Init.	Type	Description
FPMAPBINUSEBASE	23:0	0x0	RW	**Function Private Memory Accelerated Port Bit In-Use Base** Reports the base FPM space address of the accelerated port bit in-use object in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.52 FPM PE MRT Base - GLHMC_PEMRBASE[n] (0x000C4C00 + 0x4*n, n=0...15; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only.

Field	Bit(s)	Init.	Type	Description
FPMPEMRBASE	23:0	0x0	RW	**Function Private Memory PE Memory Region Base** Reports the FPM space base address for the PE memory region table objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.53 FPM PE Memory Region Table Object Count - GLHMC_PEMRCNT[n] (0x000C4D00 + 0x4*n, n=0...15; RO)

The associated base register (Section 38.39.2.13.52) is updated after commit FPM values CQP operation is performed to indicate to hardware that all of the FPM size registers have been set properly and the FPM map should be recomputed.

Field	Bit(s)	Init.	Type	Description
FPMPEMRSZ	28:0	0x0	RW	**Function Private Memory PE Memory Region Size** Used to set the FPM space size for the PE memory region table objects.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.54 FPM PE Transmit FIFO Base - GLHMC_PEXFBASE[n] (0x000C4E00 + 0x4*n, n=0...15; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only.

Field	Bit(s)	Init.	Type	Description
FPMPEXFBASE	23:0	0x0	RW	**Function Private Memory PE Transmit FIFO Base** Reports the FPM space base address for the PE transmit FIFO objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.55FPM PE Transmit FIFO Object Count - GLHMC_PEXFCNT[n] (0x000C4F00 + 0x4*n, n=0...15; RO)

The associated base register (Section 38.39.2.13.54) is updated after commit FPM values CQP operation is performed to indicate to hardware that all of the FPM size registers have been set properly and the FPM map should be recomputed.

Field	Bit(s)	Init.	Type	Description
FPMPEXFCNT	28:0	0x0	RW	**Function Private Memory PE Transmit FIFO Count** Used to set the FPM space size for the PE transmit FIFO objects.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.56FPM PE Transmit FIFO Free List Base - GLHMC_PEXFFLBASE[n] (0x000C5000 + 0x4*n, n=0...15; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only.

Field	Bit(s)	Init.	Type	Description
FPMPEXFFLBASE	23:0	0x0	RW	**Function Private Memory PE Transmit FIFO Free List Base** Reports the FPM space base address for the PE transmit FIFO free list objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.57FPM PE IRRQ Base - GLHMC_PEQ1BASE[n] (0x000C5200 + 0x4*n, n=0...15; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only.

Field	Bit(s)	Init.	Type	Description
FPMPEQ1BASE	23:0	0x0	RW	**Function Private Memory PE Q1 Base** Reports the FPM space base address for the PE inbound RDMA read queue (Q1) objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.58FPM PE IRRQ Object Count - GLHMC_PEQ1CNT[n] (0x000C5300 + 0x4*n, n=0...15; RO)

The associated base register (Section 38.39.2.13.57) is updated after commit FPM values CQP operation is performed to indicate to hardware that all of the FPM size registers have been set properly and the FPM map should be recomputed.

Field	Bit(s)	Init.	Type	Description
FPMPEQ1CNT	28:0	0x0	RW	**Function Private Memory PE Q1 Count** Used to set the FPM space size for the PE inbound RDMA read queue (Q1) objects.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.59 FPM PE IRRQ Free List Base - GLHMC_PEQ1FLBASE[n] (0x000C5400 + 0x4*n, n=0...15; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only.

Field	Bit(s)	Init.	Type	Description
FPMPEQ1FLBASE	23:0	0x0	RW	**Function Private Memory PE Q1 Free List Base** Reports the FPM space base address for the PE inbound RDMA read queue (Q1) free list objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.60 FPM FSI Address Vector Base - GLHMC_FSIAVBASE[n] (0x000C5600 + 0x4*n, n=0...15; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only.

Field	Bit(s)	Init.	Type	Description
FPMFSIAVBASE	23:0	0x0	RW	**Function Private Memory FSI Address Vector Base** Reports the FPM space base address for the FSI address vector objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.61 FPM FSI Address Vector Object Count - GLHMC_FSIAVCNT[n] (0x000C5700 + 0x4*n, n=0...15; RO)

The associated base register (Section 38.39.2.13.60) is updated after commit FPM values CQP operation is performed to indicate to hardware that all of the FPM size registers have been set properly and the FPM map should be recomputed.

Field	Bit(s)	Init.	Type	Description
FPMFSIAVCNT	28:0	0x0	RW	**Function Private Memory FSI Address Vector Count** Used to set the FPM space size for the FSI address vector objects.
RESERVED	31:29	000b	RW	Reserved.

38.39.2.13.62 FPM PE Physical Buffer List Base - GLHMC_PEPBLBASE[n] (0x000C5800 + 0x4*n, n=0...15; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only.

Field	Bit(s)	Init.	Type	Description
FPMPEPBLBASE	23:0	0x0	RW	**Function Private Memory PE Physical Buffer List Base** Reports the FPM space base address for the PE physical buffer list objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.63 FPM PE PBL Object Count - GLHMC_PEPBLCNT[n] (0x000C5900 + 0x4*n, n=0...15; RO)

The associated base register (Section 38.39.2.13.62) is updated after commit FPM values CQP operation is performed to indicate to hardware that all of the FPM size registers have been set properly and the FPM map should be recomputed.

Field	Bit(s)	Init.	Type	Description
FPMPEPBLCNT	28:0	0x0	RW	**Function Private Memory PE Physical Buffer List Count** Used to set the FPM space size for the PE physical buffer list objects.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.64 FPM PE Timer Base - GLHMC_PETIMERBASE[n] (0x000C5A00 + 0x4*n, n=0...15; RO)

The value of this registers must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only.

Note: 16 instances of this register are implemented for this product. The remaining instances are reserved for future expansion.

Field	Bit(s)	Init.	Type	Description
FPMPETIMERBASE	23:0	0x0	RW	**Function Private Memory PE Timer Base** Reports the FPM space base address for the PE timer objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.65 FPM PE Timer Object Count - GLHMC_PETIMERCNT[n] (0x000C5B00 + 0x4*n, n=0...15; RO)

The associated base register (Section 38.39.2.13.64) is updated after commit FPM values CQP operation is performed to indicate to hardware that all of the FPM size registers have been set properly and the FPM map should be recomputed.

Note: 16 instances of this register are implemented for this product. The remaining instances are reserved for future expansion.

Field	Bit(s)	Init.	Type	Description
FPMPETIMERCNT	28:0	0x0	RW	**Function Private Memory PE Timer Count** Used to set the FPM space size for the PE timer objects.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.66 FPM FSI Multicast Group Base - GLHMC_FSIMCBASE[n] (0x000C6000 + 0x4*n, n=0...15; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only.

Field	Bit(s)	Init.	Type	Description
FPMFSIMCBASE	23:0	0x0	RW	**Function Private Memory FSI Multicast Base** Reports the FPM space base address for the FSI multicast group objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.67 FPM FSI Multicast Group Object Count - GLHMC_FSIMCCNT[n] (0x000C6100 + 0x4*n, n=0...15; RO)

The associated base register (Section 38.39.2.13.66) is updated after commit FPM values CQP operation is performed to indicate to hardware that all of the FPM size registers have been set properly and the FPM map should be recomputed.

Field	Bit(s)	Init.	Type	Description
FPMFSIMCSZ	28:0	0x0	RW	**Function Private Memory FSI Multicast Size** Used to set the FPM space size for the FSI multicast group objects.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.68 FPM LAN Tx Queue Base - GLHMC_LANTXBASE[n] (0x000C6200 + 0x4*n, n=0...15; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space.

Field	Bit(s)	Init.	Type	Description
FPMLANTXBASE	23:0	0x0	RW	**Function Private Memory LAN Transmit Base** Reports the FPM space base address for the LAN transmit queue objects in 512-byte increments.
RESERVED	31:24	0x0	RW	Reserved.

38.39.2.13.69 FPM LAN Tx Queue Object Count - GLHMC_LANTXCNT[n] (0x000C6300 + 0x4*n, n=0...15; RO)

Field	Bit(s)	Init.	Type	Description
FPMLANTXCNT	10:0	0x0	RW	**Function Private Memory LAN Transmit Count** Used to set the FPM space size for the LAN Tx queue objects.
RESERVED	31:11	0x0	RSV	Reserved.

38.39.2.13.70 FPM LAN Rx Queue Base - GLHMC_LANRXBASE[n] (0x000C6400 + 0x4*n, n=0...15; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space.

Field	Bit(s)	Init.	Type	Description
FPMLANRXBASE	23:0	0x0	RW	**Function Private Memory LAN Receive Base** Reports the FPM space base address for the LAN receive queue objects in 512 - byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.71 FPM LAN Rx Queue Object Count - GLHMC_LANRXCNT[n] (0x000C6500 + 0x4*n, n=0...15; RO)

Field	Bit(s)	Init.	Type	Description
FPMLANRXCNT	10:0	0x0	RW	**Function Private Memory LAN Receive Count** Used to set the FPM space size for the LAN Rx queue objects.
RESERVED	31:11	0x0	RSV	Reserved.

38.39.2.13.72 Private Memory Space Page Descriptor Invalidate - GLHMC_VFPDINV[n] (0x000C8300 + 0x4*n, n=0...31; RO)

This register is used to invalidate cached HMC page descriptors that have been set to the invalid state by software. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
PMSDIDX	11:0	0x0	RW	**Private Memory Space Descriptor Index** Relative index of the HMC segment descriptor associated with the HMC page descriptor that is to be invalidated. For PFs, the actual index to be used to access the segment table is (*PMSDBASE* + *PMSDIDX*), where *PMSDBASE* is from the *PMSDBASE* field of the GLHMC_SDPART or GLHMC_PFPESDPART register associated with this function (Section 38.39.2.13.10 and Section 38.39.2.13.11, respectively). If (*PMSDIDX* ≥ *PMSDSIZE*), the invalidate request is dropped, where *PMSDSIZE* is from the *PMSDSIZE* field of the GLHMC_SDPART or GLHMC_PFPESDPART register associated with this function. The *PMSDPARTSEL* bit of this register determines which SD partition register is selected for the absolute SD index calculation and for the check that *PMSDIDX* is within the range allowed by this function. For PE enabled VFs, the actual index to be used to invalidate a Segment Table entry is (*PMSDBASE* + *PMSDIDX*), where *PMSDBASE* is from the GLHMC_VFSDPART register associated with this function (Section 38.39.2.13.73). For VFs, if (*PMSDIDX* ≥ GLHMC_VFSDPART.*PMSDSIZE*), the invalidate request is dropped. The *PMSDPARTSEL* bit is ignored in the GLHMC_VFPDINV registers.
RESERVED	14:12	000b	RSV	Reserved.
PMSDPARTSEL	15	0b	RW	**Private Memory Space Descriptor Partition Select** 0b = The GLHMC_SDPART register is used for calculating the absolute SD index to invalidate within the segment table, and for checking whether or not *PMSDIDX* is within this function's allocated range. 1b = The GLHMC_PFPESDPART register is used for calculating the absolute SD index to invalidate within the segment table, and for checking whether or not *PMSDIDX* is within this function's allocated range. This field is ignored in the GLHMC_VFPDINV versions of this register.
PMPDIDX	24:16	0x0	RW	**Private Memory Page Descriptor Index** Index of the page descriptor within the page descriptor page indicated by *PMSDIDX*.
RESERVED	31:25	0x0	RSV	Reserved.

38.39.2.13.73 Private Memory Segment Table Partitioning Registers - GLHMC_VFSDPART[n] (0x000C8800 + 0x4*n, n=0...31; RO)

This register is used to partition the shared HMC segment table associated with PE enabled VFs. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
PMSDBASE	11:0	0x0	RW	**Private Memory SD Base** Base segment table index for the function n.
RESERVED	15:12	0x0	RSV	Reserved.
PMSDSIZE	28:16	0x0	RW	**Private Memory SD Size** Number of valid segment table entries for the function n.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.74 FPM PE QP Base - GLHMC_VFPEQPBASE[n] (0x000CC000 + 0x4*n, n=0...31; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only and is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
FPMPEQPBASE	23:0	0x0	RW	**Function Private Memory PE Queue Pair Base** Reports the FPM space base address for the PE queue pair objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.75 FPM PE QP Object Count - GLHMC_VFPEQPCNT[n] (0x000CC100 + 0x4*n, n=0...31; RO)

The associated base register (Section 38.39.2.13.74) is updated after commit FPM values CQP operation is performed to indicate to hardware that all of the FPM size registers have been set properly and the FPM map should be recomputed. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
FPMPEQPCNT	28:0	0x0	RW	**Function Private Memory PE Queue Pair Count** Used to set the FPM space size for the PE queue pair objects.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.76 FPM PE CQ Base - GLHMC_VFPECQBASE[n] (0x000CC200 + 0x4*n, n=0…31; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only and is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
FPMPECQBASE	23:0	0x0	RW	**Function Private Memory PE Completion Queue Base** Reports the FPM space base address for the PE completion queue objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.77 FPM PE CQ Object Count - GLHMC_VFPECQCNT[n] (0x000CC300 + 0x4*n, n=0…31; RO)

The associated base register (Section 38.39.2.13.76) is updated after commit FPM values CQP operation is performed to indicate to hardware that all of the FPM size registers have been set properly and the FPM map should be recomputed. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
FPMPECQCNT	28:0	0x0	RW	**Function Private Memory PE Completion Queue Count** Used to set the FPM space size for the PE completion queue objects.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.78 FPM PE Shared RQ Base - GLHMC_VFPESRQBASE[n] (0x000CC400 + 0x4*n, n=0…31; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only and is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
FPMPESRQBASE	23:0	0x0	RW	**Function Private Memory PE Shared Receive Queue Base** Reports the FPM space base address for the PE shared receive queue objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.79 FPM PE Shared RQ Size - GLHMC_VFPESRQCNT[n] (0x000CC500 + 0x4*n, n=0...31; RO)

The associated base register (Section 38.39.2.13.78) is updated after commit FPM values CQP operation is performed to inform hardware that all of the FPM size registers have been set properly and the FPM map should be recomputed. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
FPMPESRQCNT	28:0	0x0	RW	**Function Private Memory PE Shared Receive Queue Count** Used to set the FPM space size for the PE shared receive queue objects.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.80 FPM PE Hash Table Entry Base - GLHMC_VFPEHTEBASE[n] (0x000CC600 + 0x4*n, n=0...31; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only and is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
FPMPEHTEBASE	23:0	0x0	RW	**Function Private Memory PE Hash Table Entry Base** Reports the FPM space base address for the PE hash table entry objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.81 FPM PE Hash Table Object Count - GLHMC_VFPEHTCNT[n] (0x000CC700 + 0x4*n, n=0...31; RO)

The associated base register (Section 38.39.2.13.80) is updated after commit FPM values CQP operation is performed to indicate to hardware that all of the FPM size registers have been set properly and the FPM map should be recomputed. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
FPMPEHTCNT	28:0	0x0	RW	**Function Private Memory PE Hash Table Count** Used to set the FPM space size for the PE hash table objects.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.82 FPM PE ARP Table Base - GLHMC_VFPEARPBASE[n] (0x000CC800 + 0x4*n, n=0...31; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only and is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
FPMPEARPBASE	23:0	0x0	RW	**Function Private Memory PE ARP Base** Reports the FPM space base address for the PE ARP table objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.83 FPM PE ARP Table Object Count - GLHMC_VFPEARPCNT[n] (0x000CC900 + 0x4*n, n=0...31; RO)

The associated base register (Section 38.39.2.13.82) is updated after commit FPM values CQP operation is performed to indicate to hardware that all of the FPM size registers have been set properly and the FPM map should be recomputed. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
FPMPEARPCNT	28:0	0x0	RW	**Function Private Memory PE ARP Count** Used to set the FPM space size for the PE ARP table objects.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.84 FPM PE APBVT In-Use Base - GLHMC_VFAPBVTINUSEBASE[n] (0x000CCA00 + 0x4*n, n=0...31; RO)

There is a single APBVT in-use object for each PCI function and it is a fixed 8 KB in size. Since there is only a single object and it is a fixed size, there is not a corresponding object size or region size register. This register is updated by hardware when commit FPM values CQP operation is performed. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
FPMAPBINUSEBASE	23:0	0x0	RW	**Function Private Memory Accelerated Port Bit In-Use Base** Reports the base FPM space address of the accelerated port bit in-use object in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.85 FPM PE MRT Base - GLHMC_VFPEMRBASE[n] (0x000CCC00 + 0x4*n, n=0...31; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only and is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
FPMPEMRBASE	23:0	0x0	RW	**Function Private Memory PE Memory Region Base** Reports the FPM space base address for the PE memory region table objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.86 FPM PE Memory Region Table Object Count - GLHMC_VFPEMRCNT[n] (0x000CCD00 + 0x4*n, n=0...31; RO)

The associated base register (Section 38.39.2.13.85) is updated after commit FPM values CQP operation is performed to indicate to hardware that all of the FPM size registers have been set properly and the FPM map should be recomputed. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
FPMPEMRSZ	28:0	0x0	RW	**Function Private Memory PE Memory Region Size** Used to set the FPM space size for the PE memory region table objects.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.87 FPM PE Transmit FIFO Base - GLHMC_VFPEXFBASE[n] (0x000CCE00 + 0x4*n, n=0...31; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only and is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
FPMPEXFBASE	23:0	0x0	RW	**Function Private Memory PE Transmit FIFO Base** Reports the FPM space base address for the PE transmit FIFO objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.88 FPM PE Transmit FIFO Object Count - GLHMC_VFPEXFCNT[n] (0x000CCF00 + 0x4*n, n=0...31; RO)

The associated base register (Section 38.39.2.13.87) is updated after commit FPM values CQP operation is performed to indicate to hardware that all of the FPM size registers have been set properly and the FPM map should be recomputed. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
FPMPEXFCNT	28:0	0x0	RW	**Function Private Memory PE Transmit FIFO Count** Used to set the FPM space size for the PE transmit FIFO objects.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.89 FPM PE Transmit FIFO Free List Base - GLHMC_VFPEXFFLBASE[n] (0x000CD000 + 0x4*n, n=0...31; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only and is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
FPMPEXFFLBASE	23:0	0x0	RW	**Function Private Memory PE Transmit FIFO Free List Base** Reports the FPM space base address for the PE transmit FIFO free list objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.90 FPM PE IRRQ Base - GLHMC_VFPEQ1BASE[n] (0x000CD200 + 0x4*n, n=0...31; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only and is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
FPMPEQ1BASE	23:0	0x0	RW	**Function Private Memory PE Q1 Base** Reports the FPM space base address for the PE inbound RDMA read queue (Q1) objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.91 FPM PE IRRQ Object Count - GLHMC_VFPEQ1CNT[n] (0x000CD300 + 0x4*n, n=0...31; RO)

The associated base register (Section 38.39.2.13.90) is updated after commit FPM values CQP operation is performed to indicate to hardware that all of the FPM size registers have been set properly and the FPM map should be recomputed. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
FPMPEQ1CNT	28:0	0x0	RW	**Function Private Memory PE Q1 Count** Used to set the FPM space size for the PE inbound RDMA read queue (Q1) objects.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.92 FPM PE IRRQ Free List Base - GLHMC_VFPEQ1FLBASE[n] (0x000CD400 + 0x4*n, n=0...31; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only and is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
FPMPEQ1FLBASE	23:0	0x0	RW	**Function Private Memory PE Q1 Free List Base** Reports the FPM space base address for the PE inbound RDMA read queue (Q1) free list objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.93 FPM FSI Address Vector Base - GLHMC_VFFSIAVBASE[n] (0x000CD600 + 0x4*n, n=0...31; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only and is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
FPMFSIAVBASE	23:0	0x0	RW	**Function Private Memory FSI Address Vector Base** Reports the FPM space base address for the FSI address vector objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.94 FPM FSI Address Vector Object Count - GLHMC_VFFSIAVCNT[n] (0x000CD700 + 0x4*n, n=0...31; RO)

The associated base register (Section 38.39.2.13.93) is updated after commit FPM values CQP operation is performed to indicate to hardware that all of the FPM size registers have been set properly and the FPM map should be recomputed. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
FPMFSIAVCNT	28:0	0x0	RW	**Function Private Memory FSI Address Vector Count** Used to set the FPM space size for the FSI address vector objects.
RESERVED	31:29	000b	RW	Reserved.

38.39.2.13.95 FPM PE Physical Buffer List Base - GLHMC_VFPEPBLBASE[n] (0x000CD800 + 0x4*n, n=0...31; RO)

The value in this register must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only and is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
FPMPEPBLBASE	23:0	0x0	RW	**Function Private Memory PE Physical Buffer List Base** Reports the FPM space base address for the PE physical buffer list objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.96 FPM PE PBL Object Count - GLHMC_VFPEPBLCNT[n] (0x000CD900 + 0x4*n, n=0...31; RO)

The associated base register (Section 38.39.2.13.95) is updated after commit FPM values CQP operation is performed to indicate to hardware that all of the FPM size registers have been set properly and the FPM map should be recomputed. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
FPMPEPBLCNT	28:0	0x0	RW	**Function Private Memory PE Physical Buffer List Count** Used to set the FPM space size for the PE physical buffer list objects.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.97FPM VF PE Timer Base - GLHMC_VFPETIMERBASE[n] (0x000CDA00 + 0x4*n, n=0...31; RO)

The value of this registers must be multiplied by 512 to get the actual address out of the 8 GB FPM address space. This register is updated by hardware when commit FPM values CQP operation is performed. Other than for debug purposes, this register should be treated as read only.

Note: 16 instances of this register are implemented for this product. The remaining instances are reserved for future expansion.

Field	Bit(s)	Init.	Type	Description
FPMPETIMERBASE	23:0	0x0	RW	**Function Private Memory PE Timer Base** Reports the FPM space base address for the PE timer objects in 512-byte increments.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.98FPM VF PE Timer Object Count - GLHMC_VFPETIMERCNT[n] (0x000CDB00 + 0x4*n, n=0...31; RO)

The associated base register (Section 38.39.2.13.97) is updated after commit FPM values CQP operation is performed to indicate to hardware that all of the FPM size registers have been set properly and the FPM map should be recomputed.

Note: 16 instances of this register are implemented for this product. The remaining instances are reserved for future expansion.

Field	Bit(s)	Init.	Type	Description
FPMPETIMERCNT	28:0	0x0	RW	**Function Private Memory PE Timer Count** Used to set the FPM space size for the PE timer objects.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.13.99PDOC Cache Attributes - GLPDOC_CACHESIZE (0x000D0088; RO)

Field	Bit(s)	Init.	Type	Description
WORD_SIZE	7:0	0x40	RO	**Word Size** The cache line size in bytes.
SETS	19:8	0x40	RO	**Sets** The number of cache sets.
WAYS	23:20	0x8	RO	**Ways** The number of cache ways.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.13.100 Private Memory CQ Doorbell Partition Registers - GLHMC_DBCQPART[n] (0x00131240 + 0x4*n, n=0...15; RO)

These registers are used to partition the shared doorbell array for PE completion queues PCIe PFs.

Field	Bit(s)	Init.	Type	Description
PMDBCQBASE	13:0	0x0	RW	**Private Memory Doorbell Completion Queue Base** Base CQ doorbell array index for the HMC PCI function n in multiples of eight CQs.
RESERVED	15:14	00b	RSV	Reserved.
PMDBCQSIZE	30:16	0x0	RW	**Private Memory Doorbell Completion Queue Size** Number of valid doorbell array elements for the HMC PCI function n in increments of eight CQs.
RESERVED	31	0b	RSV	Reserved.

38.39.2.13.101 Private Memory CEQ Partitioning Registers - GLHMC_CEQPART[n] (0x001312C0 + 0x4*n, n=0...15; RO)

This register is used to partition the shared pool of PE completion event queues for PCIe PFs.

Field	Bit(s)	Init.	Type	Description
PMCEQBASE	7:0	0x0	RW	**Private Memory Completion Event Queue Base** Base CEQ index for the function n.
RESERVED	15:8	0x0	RSV	Reserved.
PMCEQSIZE	24:16	0x0	RW	**Private Memory Completion Event Queue Size** Number of valid CEQs index for the function n.
RESERVED	31:25	0x0	RSV	Reserved.

38.39.2.13.102 Private Memory CQ Doorbell Partition Registers - GLHMC_VFDBCQPART[n] (0x00132140 + 0x4*n, n=0...31; RO)

These registers are used to partition the shared doorbell array for PE completion queues associated with PE enabled VFs. Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
PMDBCQBASE	13:0	0x0	RW	**Private Memory Doorbell Completion Queue Base** Base CQ doorbell array index for the HMC PCI function n in multiples of eight CQs.
RESERVED	15:14	00b	RSV	Reserved.
PMDBCQSIZE	30:16	0x0	RW	**Private Memory Doorbell Completion Queue Size** Number of valid doorbell array elements for the HMC PCI function n in increments of eight CQs.
RESERVED	31	0b	RSV	Reserved.

38.39.2.13.103Private Memory CEQ Partitioning Registers - GLHMC_VFCEQPART[n] (0x00132240 + 0x4*n, n=0...31; RO)

This register is used to partition the shared pool of PE completion event queues associated with PE enabled VFs. Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
PMCEQBASE	7:0	0x0	RW	**Private Memory Completion Event Queue Base** Base CEQ index for the function n.
RESERVED	15:8	0x0	RSV	Reserved.
PMCEQSIZE	24:16	0x0	RW	**Private Memory Completion Event Queue Size** Number of valid CEQs index for the function n.
RESERVED	31:25	0x0	RSV	Reserved.

38.39.2.13.104Private Memory QP Doorbell Partition Registers - GLHMC_DBQPPART[n] (0x00138D80 + 0x4*n, n=0...15; RO)

These registers are used to partition the shared doorbell array for PE queue pairs associated with PCIe PFs.

Field	Bit(s)	Init.	Type	Description
PMDBQPBASE	13:0	0x0	RW	**Private Memory Doorbell Queue Pair Base** Base QP doorbell array index for the HMC PCI function n in multiples of 16 QPs.
RESERVED	15:14	00b	RSV	Reserved.
PMDBQPSIZE	30:16	0x0	RW	**Private Memory Doorbell Queue Pair Size** Number of valid doorbell array elements for the HMC PCI function n in increments of 16 QPs.
RESERVED	31	0b	RSV	Reserved.

38.39.2.13.105Private Memory VF QP Doorbell Partition Registers - GLHMC_VFDBQPPART[n] (0x00138E00 + 0x4*n, n=0...31; RO)

These registers are used to partition the shared doorbell array for PE queue pairs associated with PE enabled VFs. Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
PMDBQPBASE	13:0	0x0	RW	**Private Memory Doorbell Queue Pair Base** Base QP doorbell array index for the HMC PCI function n in multiples of 16 QPs.
RESERVED	15:14	00b	RSV	Reserved.
PMDBQPSIZE	30:16	0x0	RW	**Private Memory Doorbell Queue Pair Size** Number of valid doorbell array elements for the HMC PCI function n in increments of 16 QPs.
RESERVED	31	0b	RSV	Reserved.

38.39.2.14 PF - Context Manager Registers

38.39.2.14.1 CMLAN Context Data Registers - PFCM_LANCTXDATA[n] (0x0010C100 + 0x80*n, n=0...3; RW)

This register is indexed by PF index (n) and by context word index. Word index 0 is the least significant word, and line 3 is the most significant word.

Field	Bit(s)	Init.	Type	Description
DATA	31:0	0x0	RW	**Data** 32-bit portion of the 128-bit context sub-line data.

38.39.2.14.2 CMLAN Context Control Register - PFCM_LANCTXCTL (0x0010C300; RW)

This register provides an interface into the context cache for pre-boot context initialization.

Field	Bit(s)	Init.	Type	Description
QUEUE_NUM	11:0	0x0	RW	**Queue Number** Specifies the LAN queue index to be loaded or invalidated. This number is an absolute queue number.
SUB_LINE	14:12	000b	RW	**Sub-line** Specifies the 16-byte sub-line to be accessed for write operations.
QUEUE_TYPE	16:15	00b	RW	**Queue Type** Specifies the queue type: 00b = LAN Rx context. 01b = LAN Tx context. 10b = Reserved. 11b = Reserved.
OP_CODE	18:17	00b	RW	**Op Code** Specifies the operation type: 00b = Read. 01b = Write. 10b = Invalidate. 11b = Reserved.
RESERVED	31:19	0x0	RSV	Reserved.

38.39.2.14.3 CMLAN Context Status Register - PFCM_LANCTXSTAT (0x0010C380; RO)

This register provides completion status for operations initiated via the CMLAN context control register.

Field	Bit(s)	Init.	Type	Description
CTX_DONE	0	0b	RO	**Context Operation Done** 0b = A previous LAN context operation that was initiated by a write to the PFCM_LANCTXCTL register is in progress, or no LAN context operation has been issued. 1b = A previous LAN context operation that was initiated by a write to the PFCM_LANCTXCTL register has completed.
CTX_MISS	1	0b	RO	**Context Queue Number Not Present** Reports if the queue was in the context cache before the operation specified in the PFCM_LANCTXCTL.*OP_CODE*. 0b = The requested queue number was resident in the context cache. 1b = The requested queue number was not resident in the context cache.
RESERVED	31:2	0x0	RSV	Reserved.

38.39.2.14.4 CMLAN Cache Attributes - GLCM_LAN_CACHESIZE (0x0010C4D8; RO)

Field	Bit(s)	Init.	Type	Description
WORD_SIZE	11:0	0x80	RO	**Word Size** The cache line size in bytes.
SETS	15:12	0x1	RO	**Sets** The number of cache sets.
WAYS	25:16	0x200	RO	**Ways** The number of cache ways.
RESERVED	31:26	0x0	RSV	Reserved.

38.39.2.14.5 CMPE Cache Attributes - GLCM_PE_CACHESIZE (0x00138FE4; RO)

Field	Bit(s)	Init.	Type	Description
WORD_SIZE	11:0	0x200	RO	**Word Size** The cache line size in bytes.
SETS	15:12	0x1	RO	**Sets** The number of cache sets.
WAYS	24:16	0x100	RO	**Ways** The number of cache ways.
RESERVED	31:25	0x0	RSV	Reserved.

38.39.2.15 PF - Admin Queue

38.39.2.15.1 PF Admin Transmit Queue Base Address Low - PF_ATQBAL (0x00080000; RW)

Field	Bit(s)	Init.	Type	Description
ATQBAL	31:0	0x0	RW	**Admin Transmit Queue Base Address Low** Transmit descriptor base address low. Must be 64-byte aligned. Contains the lower bits of the 64-bit descriptor base address.

38.39.2.15.2 Global Admin Transmit Queue Base Address Low - GL_ATQBAL (0x00080040; RW)

Field	Bit(s)	Init.	Type	Description
ATQBAL	31:0	0x0	RW	**Admin Transmit Queue Base Address Low** Transmit descriptor base address low. Must be 64-byte aligned. Contains the lower bits of the 64-bit descriptor base address.

38.39.2.15.3 PF Admin Receive Queue Base Address Low - PF_ARQBAL (0x00080080; RW)

Field	Bit(s)	Init.	Type	Description
ARQBAL	31:0	0x0	RW	**Admin Receive Queue Base Address Low** Receive descriptor base address low. Must be 64-byte aligned. Contains the lower bits of the 64-bit descriptor base address.

38.39.2.15.4 Global Admin Receive Queue Base Address Low - GL_ARQBAL (0x000800C0; RW)

Field	Bit(s)	Init.	Type	Description
ARQBAL	31:0	0x0	RW	**Admin Receive Queue Base Address Low** Receive descriptor base address low. Must be 64-byte aligned. Contains the lower bits of the 64-bit descriptor base address.

38.39.2.15.5 PF Admin Transmit Queue Base Address High - PF_ATQBAH (0x00080100; RW)

Field	Bit(s)	Init.	Type	Description
ATQBAH	31:0	0x0	RW	**Admin Transmit Queue Base Address High** Transmit descriptor base address high. Contains the upper bits of the 64-bit descriptor base address.

38.39.2.15.6 Global Admin Transmit Queue Base Address High - GL_ATQBAH (0x00080140; RW)

Field	Bit(s)	Init.	Type	Description
ATQBAH	31:0	0x0	RW	**Admin Transmit Queue Base Address High** Transmit descriptor base address high. Contains the upper bits of the 64-bit descriptor base address.

38.39.2.15.7 PF Admin Receive Queue Base Address High - PF_ARQBAH (0x00080180; RW)

Field	Bit(s)	Init.	Type	Description
ARQBAH	31:0	0x0	RW	**Admin Receive Queue Base Address High** Receive descriptor base address high. Contains the upper bits of the 64-bit descriptor base address.

38.39.2.15.8 Global Admin Receive Queue Base Address High - GL_ARQBAH (0x000801C0; RW)

Field	Bit(s)	Init.	Type	Description
ARQBAH	31:0	0x0	RW	**Admin Receive Queue Base Address High** Receive descriptor base address high. Contains the upper bits of the 64-bit descriptor base address.

38.39.2.15.9 PF Admin Transmit Queue Length - PF_ATQLEN (0x00080200; RW)

Field	Bit(s)	Init.	Type	Description
ATQLEN	9:0	0x0	RW	**Admin Transmit Queue Length** Descriptor ring length. Max size is 1024.
RESERVED	27:10	0x0	RSV	Reserved.
ATQVFE	28	0b	RW	**Admin Transmit Queue VF Error** Set by firmware on a PF queue when one of its VFs has an admin queue error.
ATQOVFL	29	0b	RW	**Admin Transmit Queue Overflow Error** Set by firmware when a message is lost because there is no room in the queue.
ATQCRIT	30	0b	RW	**Admin Transmit Queue Critical Error** Set by firmware when a critical error is detected on this queue.

Field	Bit(s)	Init.	Type	Description
ATQENABLE	31	0b	RW	**Admin Transmit Queue Enable** Set by driver to indicate that the queue is active. When setting the enable bit, software should initialize all other fields. This flag is cleared by PFR.

38.39.2.15.10Global Admin Transmit Queue Length - GL_ATQLEN (0x00080240; RW)

Field	Bit(s)	Init.	Type	Description
ATQLEN	9:0	0x0	RW	**Admin Transmit Queue Length** Descriptor ring length. Max size is 1024.
RESERVED	27:10	0x0	RSV	Reserved.
ATQVFE	28	0b	RW	**Admin Transmit Queue VF Error** Set by firmware on a PF queue when one of its VFs has an admin queue error.
ATQOVFL	29	0b	RW	**Admin Transmit Queue Overflow Error** Set by firmware when a message is lost because there is no room in the queue.
ATQCRIT	30	0b	RW	**Admin Transmit Queue Critical Error** Set by firmware when a critical error is detected on this queue.
ATQENABLE	31	0b	RW	**Admin Transmit Queue Enable** Set by the software device driver to indicate that the queue is active. When setting the enable bit, software should initialize all other fields. This flag is cleared by CORER.

38.39.2.15.11PF Admin Receive Queue Length - PF_ARQLEN (0x00080280; RW)

Field	Bit(s)	Init.	Type	Description
ARQLEN	9:0	0x0	RW	**Admin Receive Queue Length** Descriptor ring length. Max size is 1024.
RESERVED	27:10	0x0	RSV	Reserved.
ARQVFE	28	0b	RW	**Admin Receive Queue VF Error** Set by firmware on a PF queue when one of its VFs has an admin queue error.
ARQOVFL	29	0b	RW	**Admin Receive Queue Overflow Error** Set by firmware when a message is lost because there is no room in the queue.
ARQCRIT	30	0b	RW	**Admin Receive Queue Critical Error** Set by firmware when a critical error is detected on this queue.
ARQENABLE	31	0b	RW	**Admin Receive Queue Enable** Set by the software device driver to indicate that the queue is active. When setting the enable bit, software should initialize all other fields. This flag is cleared by PFR.

38.39.2.15.12PF Admin Transmit Head - PF_ATQH (0x00080300; RW)

Field	Bit(s)	Init.	Type	Description
ATQH	9:0	0x0	RW	**Admin Transmit Queue Head** Transmit queue head pointer. At queue initialization, software clears the head pointer. During normal operation, firmware increments the head following command execution.
RESERVED	31:10	0x0	RSV	Reserved.

38.39.2.15.13Global Admin Transmit Head - GL_ATQH (0x00080340; RW)

Field	Bit(s)	Init.	Type	Description
ATQH	9:0	0x0	RW	**Admin Transmit Queue Head** Transmit queue head pointer. At queue initialization, software clears the head pointer. During normal operation, firmware increments the head following command execution.
RESERVED	31:10	0x0	RSV	Reserved.

38.39.2.15.14PF Admin Receive Queue Head - PF_ARQH (0x00080380; RW)

Field	Bit(s)	Init.	Type	Description
ARQH	9:0	0x0	RW	**Admin Receive Queue Head** Receive queue head pointer. At queue initialization, software clears the head pointer. During normal operation, firmware increments the head following command execution.
RESERVED	31:10	0x0	RSV	Reserved.

38.39.2.15.15Global Admin Receive Queue Head - GL_ARQH (0x000803C0; RW)

Field	Bit(s)	Init.	Type	Description
ARQH	9:0	0x0	RW	**Admin Receive Queue Head** Receive queue head pointer. At queue initialization, software clears the head pointer. During normal operation, firmware increments the head following command execution.
RESERVED	31:10	0x0	RSV	Reserved.

38.39.2.15.16PF Admin Transmit Tail - PF_ATQT (0x00080400; RW)

Field	Bit(s)	Init.	Type	Description
ATQT	9:0	0x0	RW	**Admin Transmit Queue Tail** Transmit queue tail pointer. Incremented to indicate that there are new valid descriptors on the ring. Software can only write to this register once both transmit and receive queues are properly initialized, and clears to zero at queue initialization.
RESERVED	31:10	0x0	RSV	Reserved.

38.39.2.15.17Global Admin Transmit Tail - GL_ATQT (0x00080440; RW)

Field	Bit(s)	Init.	Type	Description
ATQT	9:0	0x0	RW	**Admin Transmit Queue Tail** Transmit queue tail pointer. Incremented to indicate that there are new valid descriptors on the ring. Software can only write to this register once both transmit and receive queues are properly initialized, and clears to zero at queue initialization.
RESERVED	31:10	0x0	RSV	Reserved.

38.39.2.15.18PF Admin Receive Queue Tail - PF_ARQT (0x00080480; RW)

Field	Bit(s)	Init.	Type	Description
ARQT	9:0	0x0	RW	**Admin Receive Queue Tail** Receive queue tail pointer. Incremented to indicate that there are new valid descriptors on the ring. Software can only write to this register once the queue is fully configured, and clears to zero at queue initialization.
RESERVED	31:10	0x0	RSV	Reserved.

38.39.2.15.19Global Admin Receive Queue Tail - GL_ARQT (0x000804C0; RW)

Field	Bit(s)	Init.	Type	Description
ARQT	9:0	0x0	RW	**Admin Receive Queue Tail** Receive queue tail pointer. Incremented to indicate that there are new valid descriptors on the ring. Software can only write to this register once the queue is fully configured, and clears to zero at queue initialization.
RESERVED	31:10	0x0	RSV	Reserved.

38.39.2.15.20VF Admin Transmit Queue Base Address Low - VF_ATQBAL[VF] (0x00080800 + 0x4*VF, VF=0...127; RW)

Field	Bit(s)	Init.	Type	Description
ATQBAL	31:0	0x0	RW	**Admin Transmit Queue Base Address Low** Transmit descriptor base address low. Must be 64-byte aligned. Contains the lower bits of the 64-bit descriptor base address.

38.39.2.15.21VF Admin Receive Queue Base Address Low - VF_ARQBAL[VF] (0x00080C00 + 0x4*VF, VF=0...127; RW)

Field	Bit(s)	Init.	Type	Description
ARQBAL	31:0	0x0	RW	**Admin Receive Queue Base Address Low** Receive descriptor base address low. Must be 64-byte aligned. Contains the lower bits of the 64-bit descriptor base address.

38.39.2.15.22VF Admin Transmit Queue Base Address High - VF_ATQBAH[VF] (0x00081000 + 0x4*VF, VF=0...127; RW)

Field	Bit(s)	Init.	Type	Description
ATQBAH	31:0	0x0	RW	**Admin Transmit Queue Base Address High** Transmit descriptor base address high. Contains the upper bits of the 64-bit descriptor base address.

38.39.2.15.23VF admin receive queue base address high - VF_ARQBAH[VF] (0x00081400 + 0x4*VF, VF=0...127; RW)

Field	Bit(s)	Init.	Type	Description
ARQBAH	31:0	0x0	RW	**Admin Receive Queue Base Address High** Receive descriptor base address high. Contains the upper bits of the 64-bit descriptor base address.

38.39.2.15.24VF Admin Transmit Queue Length - VF_ATQLEN[VF] (0x00081800 + 0x4*VF, VF=0...127; RW)

Field	Bit(s)	Init.	Type	Description
ATQLEN	9:0	0x0	RW	**Admin Transmit Queue Length** Descriptor ring length. Max size is 1024.
RESERVED	27:10	0x0	RSV	Reserved.
ATQVFE	28	0b	RW	**Admin Transmit Queue VF Error** Set by firmware on a PF queue when one of its VFs has an admin queue error.
ATQOVFL	29	0b	RW	**Admin Transmit Queue Overflow Error** Set by firmware when a message is lost because there is no room in the queue.

Field	Bit(s)	Init.	Type	Description
ATQCRIT	30	0b	RW	**Admin Transmit Queue Critical Error** Set by firmware when a critical error is detected on this queue.
ATQENABLE	31	0b	RW	**Admin Transmit Queue Enable** Set by the software device driver to indicate that the queue is active. When setting the enable bit, software should initialize all other fields. This flag is cleared by VFR.

38.39.2.15.25VF Admin Receive Queue Length - VF_ARQLEN[VF] (0x00081C00 + 0x4*VF, VF=0...127; RW)

Field	Bit(s)	Init.	Type	Description
ARQLEN	9:0	0x0	RW	**Admin Receive Queue Length** Descriptor ring length. Max size is 1024.
RESERVED	27:10	0x0	RSV	Reserved.
ARQVFE	28	0b	RW	**Admin Receive Queue VF Error** Set by firmware on a PF queue when one of its VFs has an admin queue error.
ARQOVFL	29	0b	RW	**Admin Receive Queue Overflow Error** Set by firmware when a message is lost because there is no room in the queue.
ARQCRIT	30	0b	RW	**Admin Receive Queue Critical Error** Set by firmware when a critical error is detected on this queue.
ARQENABLE	31	0b	RW	**Admin Receive Queue Enable** Set by the software device driver to indicate that the queue is active. When setting the enable bit, software should initialize all other fields. This flag is cleared by PFR.

38.39.2.15.26VF Admin Transmit Head - VF_ATQH[VF] (0x00082000 + 0x4*VF, VF=0...127; RW)

Field	Bit(s)	Init.	Type	Description
ATQH	9:0	0x0	RW	**Admin Transmit Queue Head** Transmit queue head pointer. At queue initialization, software clears the head pointer. During normal operation, firmware increments the head following command execution.
RESERVED	31:10	0x0	RSV	Reserved.

38.39.2.15.27VF Admin Receive Queue Head - VF_ARQH[VF] (0x00082400 + 0x4*VF, VF=0...127; RW)

Field	Bit(s)	Init.	Type	Description
ARQH	9:0	0x0	RW	**Admin Receive Queue Head** Transmit queue head pointer. At queue initialization, software clears the head pointer. During normal operation, firmware increments the head following command execution.
RESERVED	31:10	0x0	RSV	Reserved.

38.39.2.15.28 VF Admin Transmit Tail - VF_ATQT[VF] (0x00082800 + 0x4*VF, VF=0...127; RW)

Field	Bit(s)	Init.	Type	Description
ATQT	9:0	0x0	RW	**Admin Transmit Queue Tail** Transmit queue tail pointer. Incremented to indicate that there are new valid descriptors on the ring. Software can only write to this register once both transmit and receive queues are properly initialized, and clears to zero at queue initialization.
RESERVED	31:10	0x0	RSV	Reserved.

38.39.2.15.29 VF Admin Receive Queue Tail - VF_ARQT[VF] (0x00082C00 + 0x4*VF, VF=0...127; RW)

Field	Bit(s)	Init.	Type	Description
ARQT	9:0	0x0	RW	**Admin Receive Queue Tail** Receive queue tail pointer. Incremented to indicate that there are new valid descriptors on the ring. Software can only write to this register once the queue is fully configured, and clears to zero at queue initialization.
RESERVED	31:10	0x0	RSV	Reserved.

38.39.2.16 PF - Statistics Registers

Statistics counters. Refer to the Statistics section for more details.

38.39.2.16.1 Port Good Octets Received Count Low - GLPRT_GORCL[n] (0x00300000 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
GORCL	31:0	0x0	RW1C	**Good Octets Received Count Low** Counts number of bytes received by this port. Lower 32 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.2 Port good octets received count high - GLPRT_GORCH[n] (0x00300004 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
GORCH	15:0	0x0	RW1C	**Good Octets Received Count High** Counts number of bytes received by this port. Upper 16 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.3 Port MAC local fault count - GLPRT_MLFC[n] (0x00300020 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
MLFC	31:0	0x0	RW1C	**MAC Local Fault Count** Number of faults in the local MAC.

38.39.2.16.4 Port MAC Remote Fault Count - GLPRT_MRFC[n] (0x00300040 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
MRFC	31:0	0x0	RW1C	**MAC Remote Fault Count** Number of faults in the remote MAC.

38.39.2.16.5 Port CRC Error Count - GLPRT_CRCERRS[n] (0x00300080 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
CRCERRS	31:0	0x0	RW1C	**CRC Error Count** Counts the number of receive packets with CRC errors. This includes packets that are also counted by other error registers.

38.39.2.16.6 Receive Length Error Count - GLPRT_RLEC[n] (0x003000A0 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
RLEC	31:0	0x0	RW1C	**Receive Length Error Count** Number of packets with receive length errors. A length error occurs if an incoming packet length field in the MAC header does not match the packet length.

38.39.2.16.7 Port Illegal Byte Error Count - GLPRT_ILLERRC[n] (0x003000E0 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
ILLERRC	31:0	0x0	RW1C	**Illegal Byte Error Packet Count** Counts the number of receive packets with illegal bytes errors. For example, there is an illegal symbol in the packet.

38.39.2.16.8 Receive Undersize Count - GLPRT_RUC[n] (0x00300100 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
RUC	31:0	0x0	RW1C	**Receive Undersize Count** Counts the number of received frames that are shorter than minimum size (64 bytes from <Destination Address> through <CRC>, inclusively), and had a valid CRC.

38.39.2.16.9 Receive Oversize Count - GLPRT_ROC[n] (0x00300120 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
ROC	31:0	0x0	RW1C	**Receive Oversize Count** Counts the number of received frames that are longer than maximum size as defined by the Set MAC config command (from <Destination Address> through <CRC>, inclusively), and have valid CRC.

38.39.2.16.10 Port Link XON Received Count - GLPRT_LXONRXC[n] (0x00300140 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
LXONRXCNT	31:0	0x0	RW1C	**Link XON Received Count** Number of XON packets received.

38.39.2.16.11 Port Link XOFF Received Count - GLPRT_LXOFFRXC[n] (0x00300160 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
LXOFFRXCNT	31:0	0x0	RW1C	**Link XOFF Received Count** Number of XOFF packets received.

38.39.2.16.12 Priority XON Received Count - GLPRT_PXONRXC[n,m] (0x00300180 + 0x8*n + 0x20*m, n=0...3, m=0...7; RW1C)

Port priority FC XON received count (array of eight per port).

Field	Bit(s)	Init.	Type	Description
PRPXONRXCNT	31:0	0x0	RW1C	**Priority XON Received Count** Number of XON packets received.

38.39.2.16.13 Priority XOFF Received Count - GLPRT_PXOFFRXC[n,m] (0x00300280 + 0x8*n + 0x20*m, n=0...3, m=0...7; RW1C)

Port priority FC XOFF received count (array of eight per port).

Field	Bit(s)	Init.	Type	Description
PRPXOFFRXCNT	31:0	0x0	RW1C	**Priority XOFF Received Count** Number of XOFF packets received.

38.39.2.16.14 Priority XON to XOFF Count - GLPRT_RXON2OFFCNT[n,m] (0x00300380 + 0x8*n + 0x20*m, n=0...3, m=0...7; RW1C)

Counts the number of times the priority egress was paused (array of eight per port).

Field	Bit(s)	Init.	Type	Description
PRRXON2OFFCNT	31:0	0x0	RW1C	**Priority XON to XOFF Count** Number of times transmitter transitioned from XON to XOFF.

38.39.2.16.15 Packets Received [64 Bytes] Count Low - GLPRT_PRC64L[n] (0x00300480 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PRC64H (Section 38.39.2.16.16) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PRC64L	31:0	0x0	RW1C	**Packets Received Count [64 Bytes] Low** Number of good packets received that are 64 bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.16 Packets Received [64 Bytes] Count High - GLPRT_PRC64H[n] (0x00300484 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PRC64L (Section 38.39.2.16.15) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PRC64H	15:0	0x0	RW1C	**Packets Received Count [64 Bytes] High** Number of good packets received that are 64 bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.17 Packets Received [65-127 Bytes] Count Low - GLPRT_PRC127L[n] (0x003004A0 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PRC127H (Section 38.39.2.16.18) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PRC127L	31:0	0x0	RW1C	**Packets Received Count [65-127 Bytes] Low** Number of packets received that are 65-127 bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.18 Packets Received [65-127 Bytes] Count High - GLPRT_PRC127H[n] (0x003004A4 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PRC127L (Section 38.39.2.16.17) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PRC127H	15:0	0x0	RW1C	**Packets Received Count [65-127 Bytes] High** Number of packets received that are 65-127 bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.19 Packets Received [128-255 Bytes] Count Low - GLPRT_PRC255L[n] (0x003004C0 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PRC255H (Section 38.39.2.16.20) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PRC255L	31:0	0x0	RW1C	**Packets Received Count [128-255 Bytes] Low** Number of packets received that are 128-255 bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.20 Packets Received [128-255 Bytes] Count High - GLPRT_PRC255H[n] (0x003004C4 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PRC255L (Section 38.39.2.16.19) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PRTPRC255H	15:0	0x0	RW1C	**Packets Received Count [128-255 Bytes] High** Number of packets received that are 128-255 bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.21 Packets Received [256-511 Bytes] Count Low - GLPRT_PRC511L[n] (0x003004E0 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PRC511H (Section 38.39.2.16.22) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PRC511L	31:0	0x0	RW1C	**Packets Received Count [256-511 Bytes] Low** Number of packets received that are 256-511 bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.22 Packets Received [256-511 Bytes] Count High - GLPRT_PRC511H[n] (0x003004E4 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PRC511L (Section 38.39.2.16.21) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PRC511H	15:0	0x0	RW1C	**Packets Received Count [256-511 Bytes] High** Number of packets received that are 256-511 bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.23 Packets Received [512-1023 Bytes] Count Low - GLPRT_PRC1023L[n] (0x00300500 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PRC1023H (Section 38.39.2.16.24) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PRC1023L	31:0	0x0	RW1C	**Packets Received Count [512-1023 Bytes] Low** Number of packets received that are 512-1023 bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.24 Packets Received [512-1023 Bytes] Count High - GLPRT_PRC1023H[n] (0x00300504 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PRC1023L (Section 38.39.2.16.23) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PRC1023H	15:0	0x0	RW1C	**Packets Received Count [512-1023 Bytes] High** Number of packets received that are 512-1023 bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.25 Packets Received [1024-1522 bytes] Count Low - GLPRT_PRC1522L[n] (0x00300520 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PRC1522H (Section 38.39.2.16.26) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PRC1522L	31:0	0x0	RW1C	**Packets Received Count [1024-1522 Bytes] Low** Number of packets received that are 1024-max bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.26 Packets Received [1024-1522 bytes] Count High - GLPRT_PRC1522H[n] (0x00300524 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PRC1522L (Section 38.39.2.16.25) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PRC1522H	15:0	0x0	RW1C	**Packets Received Count [1024-1522 Bytes] High** Number of packets received that are 1024-max bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.27 Packets Received [1523-9522 Bytes] Count Low - GLPRT_PRC9522L[n] (0x00300540 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PRC9522H (Section 38.39.2.16.28) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PRC1522L	31:0	0x0	RW1C	**Packets Received Count [1523-9522 Bytes] Low** Number of packets received that are 1024-max bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.28 Packets Received [1523-9522 Bytes] Count High - GLPRT_PRC9522H[n] (0x00300544 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PRC9522L (Section 38.39.2.16.27) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PRC1522H	15:0	0x0	RW1C	**Packets Received Count [1523-9522 Bytes] High** Number of packets received that are 1024-max bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.29 Receive Fragment Count - GLPRT_RFC[n] (0x00300560 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
RFC	31:0	0x0	RW1C	**Receive Fragments Count** Counts the number of received frames that are shorter than minimum size (64 bytes from <Destination Address> through <CRC>, inclusively), and had an invalid CRC.

38.39.2.16.30 Receive Jabber Count - GLPRT_RJC[n] (0x00300580 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
RJC	31:0	0x0	RW1C	**Receive Jabber Count** Counts the number of received packets that passed address filtering, and are greater than maximum size and have bad CRC (this is slightly different from the Receive Oversize Count register). The packet length is counted from <Destination Address> through <CRC>, inclusively.

38.39.2.16.31 Port Unicast Packets Received Count Low - GLPRT_UPRCL[n] (0x003005A0 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
UPRCL	31:0	0x0	RW1C	**Unicast Packets Received Count Low** Counts number of unicast packets received by this VSI. Lower 32 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.32 Port Unicast Packets Received Count High - GLPRT_UPRCH[n] (0x003005A4 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
UPRCH	15:0	0x0	RW1C	**Unicast Packets Received Count High** Counts number of unicast packets received by this VSI. Upper 16 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.33 Port Multicast Packets Received Count Low - GLPRT_MPRCL[n] (0x003005C0 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
MPRCL	31:0	0x0	RW1C	**Multicast Packets Received Count Low** Counts number of multicast packets received by this VSI. Lower 32 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.34 Port Multicast Packets Received Count High - GLPRT_MPRCH[n] (0x003005C4 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
MPRCH	15:0	0x0	RW1C	**Multicast Packets Received Count High** Counts number of multicast packets received by this VSI. Upper 16 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.35 Port Broadcast Packets Received Count Low - GLPRT_BPRCL[n] (0x003005E0 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
BPRCL	31:0	0x0	RW1C	**Broadcast Packets Received Count Low** Counts number of broadcast packets received by this port. Lower 32 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.36 Port Broadcast Packets Received Count High - GLPRT_BPRCH[n] (0x003005E4 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
BPRCH	15:0	0x0	RW1C	**Broadcast Packets Received Count High** Counts number of broadcast packets received by this port. Upper 16 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.37 Port Receive Packets Discarded Count - GLPRT_RDPC[n] (0x00300600 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
RDPC	31:0	0x0	RW1C	**Receive Discarded Packets Count**

38.39.2.16.38 VM-VM Loopback Packets Discarded Count - GLPRT_LDPC[n] (0x00300620 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
LDPC	31:0	0x0	RW1C	**Loopback Discarded Packets Count**

38.39.2.16.39 Port Received with No Destination - GLPRT_RUPP[n] (0x00300660 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
RUPP	31:0	0x0	RW1C	**Received Unknown Packet Protocol count** Receive packets dropped for this port because they did not match any STag. (No forwarding rule).

38.39.2.16.40 Port Good Octets Transmit Count Low - GLPRT_GOTCL[n] (0x00300680 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
GOTCL	31:0	0x0	RW1C	**Good Octets Transmit Count Low** Counts number of bytes transmitted by this VSI. Lower 32 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.41 Port Good Octets Transmit Count High - GLPRT_GOTCH[n] (0x00300684 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
GOTCH	15:0	0x0	RW1C	**Good Octets Transmit Count High** Counts number of bytes transmitted by this port. Upper 16 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.42 Packets Transmitted [64 Bytes] Count Low - GLPRT_PTC64L[n] (0x003006A0 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PTC64H (Section 38.39.2.16.43) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PTC64L	31:0	0x0	RW1C	**Packets Transmitted Count [64 Bytes] Low** Number of packets transmitted that are 64 bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.43 Packets Transmitted [64 Bytes] Count High - GLPRT_PTC64H[n] (0x003006A4 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PTC64L (Section 38.39.2.16.42) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PTC64H	15:0	0x0	RW1C	**Packets Transmitted Count [64 Bytes] High** Number of packets transmitted that are 64 bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.44 Packets Transmitted [65-127 Bytes] Count Low - GLPRT_PTC127L[n] (0x003006C0 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PTC127H (Section 38.39.2.16.45) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PTC127L	31:0	0x0	RW1C	**Packets Transmitted Count [65-127 Bytes] Low** Number of packets transmitted that are 65-127 bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.45 Packets Transmitted [65-127 Bytes] Count High - GLPRT_PTC127H[n] (0x003006C4 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PTC127L (Section 38.39.2.16.44) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PTC127H	15:0	0x0	RW1C	**Packets Transmitted Count [65-127 Bytes] High** Number of packets transmitted that are 65-127 bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.46 Packets Transmitted [128-255 Bytes] Count Low - GLPRT_PTC255L[n] (0x003006E0 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PTC255H (Section 38.39.2.16.47) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PTC255L	31:0	0x0	RW1C	**Packets Transmitted Count [128-255 Bytes] Low** Number of packets transmitted that are 128-255 bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.47 Packets Transmitted [128-255 Bytes] Count High - GLPRT_PTC255H[n] (0x003006E4 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PTC255L (Section 38.39.2.16.46) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PTC255H	15:0	0x0	RW1C	**Packets Transmitted Count [128-255 Bytes] High** Number of packets transmitted that are 128-255 bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.48 Packets Transmitted [256-511 Bytes] Count Low - GLPRT_PTC511L[n] (0x00300700 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PTC511H (Section 38.39.2.16.49) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PTC511L	31:0	0x0	RW1C	**Packets Transmitted Count [256-511 Bytes] Low** Number of packets transmitted that are 256-511 bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.49 Packets Transmitted [256-511 Bytes] Count High - GLPRT_PTC511H[n] (0x00300704 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PTC511L (Section 38.39.2.16.48) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PTC511H	15:0	0x0	RW1C	**Packets Transmitted Count [256-511 Bytes] High** Number of packets transmitted that are 256-511 bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.50 Packets Transmitted [512-1023 Bytes] Count Low - GLPRT_PTC1023L[n] (0x00300720 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PTC1023H (Section 38.39.2.16.51) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PTC1023L	31:0	0x0	RW1C	**Packets Transmitted Count [512-1023 Bytes] Low** Number of packets transmitted that are 512-1023 bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.51 Packets Transmitted [512-1023 Bytes] Count High - GLPRT_PTC1023H[n] (0x00300724 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PTC1023L (Section 38.39.2.16.50) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PTC1023H	15:0	0x0	RW1C	**Packets Transmitted Count [512-1023 Bytes] High** Number of packets transmitted that are 512-1023 bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.52 Packets Transmitted [1024-1522 Bytes] Count Low - GLPRT_PTC1522L[n] (0x00300740 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PTC1522H (Section 38.39.2.16.53) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PTC1522L	31:0	0x0	RW1C	**Packets Transmitted Count [1024-1522 Bytes] Low** Number of packets transmitted that are 1024 or more bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.53 Packets Transmitted [1024-1522 Bytes] Count High - GLPRT_PTC1522H[n] (0x00300744 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PTC1522L (Section 38.39.2.16.52) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PTC1522H	15:0	0x0	RW1C	**Packets Transmitted Count [1024-1522 Bytes] High** Number of packets transmitted that are 1024 or more bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.54 Packets Transmitted [1523-9522 Bytes] Count Low - GLPRT_PTC9522L[n] (0x00300760 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PTC9522H (Section 38.39.2.16.55) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PTC9522L	31:0	0x0	RW1C	**Packets Transmitted Count [1523-9522 Bytes] Low** Number of packets transmitted that are 1523 to 9522 bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.55 Packets Transmitted [1523-9522 Bytes] Count High - GLPRT_PTC9522H[n] (0x00300764 + 0x8*n, n=0...3; RW1C)

Combined with GLPRT_PTC9522L (Section 38.39.2.16.54) to form a 64-bit register.

Field	Bit(s)	Init.	Type	Description
PTC9522H	15:0	0x0	RW1C	**Packets Transmitted Count [1523-9522 Bytes] High** Number of packets transmitted that are 1523 to 9522 bytes in length (from <Destination Address> through <CRC>, inclusively). The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.56 Priority XON Transmitted Count - GLPRT_PXONTXC[n,m] (0x00300780 + 0x8*n + 0x20*m, n=0...3, m=0...7; RW1C)

Port priority FC XON transmitted count (array of 8 per port).

Field	Bit(s)	Init.	Type	Description
PRPXONTXC	31:0	0x0	RW1C	**Priority XON Transmitted Count** Number of XON packets transmitted.

38.39.2.16.57 Priority XOFF Transmitted Count - GLPRT_PXOFFTXC[n,m] (0x00300880 + 0x8*n + 0x20*m, n=0...3, m=0...7; RW1C)

Port priority FC XOFF transmitted count (array of 8 per port).

Field	Bit(s)	Init.	Type	Description
PRPXOFFTXCNT	31:0	0x0	RW1C	**Priority XOFF Transmitted Count** Number of XOFF packets transmitted.

38.39.2.16.58 Port Link XON Transmitted Count - GLPRT_LXONTXC[n] (0x00300980 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
LXONTXC	31:0	0x0	RW1C	**Link XON Transmitted Count** Number of XON packets transmitted.

38.39.2.16.59 Port Link XOFF Transmitted Count - GLPRT_LXOFFTXC[n] (0x003009A0 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
LXOFFTXC	31:0	0x0	RW1C	**Link XOFF Transmitted Count** Number of XOFF packets transmitted.

38.39.2.16.60 Port Unicast Packets Transmit Count Low - GLPRT_UPTCL[n] (0x003009C0 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
UPTCL	31:0	0x0	RW1C	**Unicast Packets Transmit Count Low** Counts number of unicast packets transmitted by this VSI. Lower 32 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.61 Port Unicast Packets Transmit Count High - GLPRT_UPTCH[n] (0x003009C4 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
UPTCH	15:0	0x0	RW1C	**Unicast Packets Transmit Count High** Counts number of unicast packets transmitted by this VSI. Upper 16 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.62 Port Multicast Packets Transmit Count Low - GLPRT_MPTCL[n] (0x003009E0 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
MPTCL	31:0	0x0	RW1C	**Multicast Packets Transmit Count Low** Counts number of multicast packets transmitted by this VSI. Lower 32 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.63 Port Multicast Packets Transmit Count High - GLPRT_MPTCH[n] (0x003009E4 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
MPTCH	15:0	0x0	RW1C	**Multicast Packets Transmit Count High** Counts number of multicast packets transmitted by this VSI. Upper 16 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.64 Port Broadcast Packets Transmit Count Low - GLPRT_BPTCL[n] (0x00300A00 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
BPTCL	31:0	0x0	RW1C	**Broadcast Packets Transmit Count Low** Counts number of broadcast packets transmitted by this port. Lower 32-bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.65 Port Broadcast Packets Transmit Count High - GLPRT_BPTCH[n] (0x00300A04 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
BPTCH	15:0	0x0	RW1C	**Broadcast Packets Transmit Count High** Counts number of broadcast packets transmitted by this port. Upper 16 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.66 Transmit Discard on Link Down - GLPRT_TDOLD[n] (0x00300A20 + 0x8*n, n=0...3; RW1C)

Field	Bit(s)	Init.	Type	Description
GLPRT_TDOLD	31:0	0x0	RW1C	**Transmit Discard On Link Down** Packets discarded at the port because the link was down.

38.39.2.16.67VSI Received Discard Packet Count - GLV_RDPC[n] (0x00310000 + 0x8*n, n=0...383; RW1C)

Field	Bit(s)	Init.	Type	Description
RDPC	31:0	0x0	RW1C	**Received Discard Packet Count** Counts (per VSI) packets that were dropped due to no descriptors in host queue. For EMP VSIs it counts dropped packets due to no EMP buffer space.

38.39.2.16.68Receive Error Counter 1 Low - GL_RXERR1_L[n] (0x00318000 + 0x8*n, n=0...143; RW1C)

Field	Bit(s)	Init.	Type	Description
RESERVED	31:0	0x0	RW1C	**Reserved.**

38.39.2.16.69VSI Good Octets Transmit Count Low - GLV_GOTCL[n] (0x00328000 + 0x8*n, n=0...383; RW1C)

Field	Bit(s)	Init.	Type	Description
GOTCL	31:0	0x0	RW1C	**Good Octets Transmit Count Low** Counts number of bytes transmitted by this VSI. Lower 32 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.70VSI Good Octets Transmit Count High - GLV_GOTCH[n] (0x00328004 + 0x8*n, n=0...383; RW1C)

Field	Bit(s)	Init.	Type	Description
GOTCH	15:0	0x0	RW1C	**Good Octets Transmit Count High** Counts number of bytes transmitted by this VSI. Upper 16 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.71Switch Good Octets Transmit Count Low - GLSW_GOTCL[n] (0x0032C000 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
GOTCL	31:0	0x0	RW1C	**Good Octets Transmit Count Low** Counts number of bytes transmitted. Lower 32 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.72Switch Good Octets Transmit Count High - GLSW_GOTCH[n] (0x0032C004 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
GOTCH	15:0	0x0	RW1C	**Good Octets Transmit Count High** Counts number of bytes transmitted. Upper 16 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.73VEB VLAN Transmit Byte Count Low - GLVEBVL_GOTCL[n] (0x00330000 + 0x8*n, n=0...127; RW1C)

Field	Bit(s)	Init.	Type	Description
VLBCL	31:0	0x0	RW1C	**VEB per VLAN Byte Count Low** The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.74VEB VLAN Transmit Byte Count High - GLVEBVL_GOTCH[n] (0x00330004 + 0x8*n, n=0...127; RW1C)

Field	Bit(s)	Init.	Type	Description
VLBCH	15:0	0x0	RW1C	**VEB per VLAN Byte Count High** The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.75VEB TC Transmit Byte Count Low - GLVEBTC_TBCL[n,m] (0x00334000 + 0x8*n + 0x40*m, n=0...7, m=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
TCBCL	31:0	0x0	RW1C	**VEB per TC Byte Count Low** The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.76VEB TC Transmit Byte Count High - GLVEBTC_TBCH[n,m] (0x00334004 + 0x8*n + 0x40*m, n=0...7, m=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
TCBCH	15:0	0x0	RW1C	**VEB per TC Byte Count High** The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.77VEB TC Transmit Packet Count Low - GLVEBTC_TPCL[n,m] (0x00338000 + 0x8*n + 0x40*m, n=0...7, m=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPCL	31:0	0x0	RW1C	**VEB per TC Packet Count Low** The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.78VEB TC Transmit Packet Count High - GLVEBTC_TPCH[n,m] (0x00338004 + 0x8*n + 0x40*m, n=0...7, m=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPCH	15:0	0x0	RW1C	**VEB per TC Packet Count High** The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.79VSI Unicast Packets Transmit Count Low - GLV_UPTCL[n] (0x0033C000 + 0x8*n, n=0...383; RW1C)

Field	Bit(s)	Init.	Type	Description
UPTCL	31:0	0x0	RW1C	**Unicast Packets Transmit Count Low** Counts number of unicast packets transmitted by this VSI. Lower 32 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.80VSI Unicast Packets Transmit Count High - GLV_UPTCH[n] (0x0033C004 + 0x8*n, n=0...383; RW1C)

Field	Bit(s)	Init.	Type	Description
UPTCH	15:0	0x0	RW1C	**Unicast Packets Transmit Count High** Counts number of unicast packets transmitted by this VSI. Upper 16 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.81VSI Multicast Packets Transmit Count Low - GLV_MPTCL[n] (0x0033CC00 + 0x8*n, n=0...383; RW1C)

Field	Bit(s)	Init.	Type	Description
MPTCL	31:0	0x0	RW1C	**Multicast Packets Transmit Count Low** Counts number of multicast packets transmitted by this VSI. Lower 32 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.82VSI Multicast Packets Transmit Count High - GLV_MPTCH[n] (0x0033CC04 + 0x8*n, n=0...383; RW1C)

Field	Bit(s)	Init.	Type	Description
MPTCH	15:0	0x0	RW1C	**Multicast Packets Transmit Count High** Counts number of multicast packets transmitted by this VSI. Upper 16 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.83VSI Broadcast Packets Transmit Count Low - GLV_BPTCL[n] (0x0033D800 + 0x8*n, n=0...383; RW1C)

Field	Bit(s)	Init.	Type	Description
BPTCL	31:0	0x0	RW1C	**Broadcast Packets Transmit Count Low** Counts number of broadcast packets transmitted by this VSI. Lower 32 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.84VSI Broadcast Packets Transmit Count High - GLV_BPTCH[n] (0x0033D804 + 0x8*n, n=0...383; RW1C)

Field	Bit(s)	Init.	Type	Description
BPTCH	15:0	0x0	RW1C	**Broadcast Packets Transmit Count High** Counts number of broadcast packets transmitted by this VSI. Upper 16 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.85Switch Unicast Packets Transmit Count Low - GLSW_UPTCL[n] (0x00340000 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
UPTCL	31:0	0x0	RW1C	**Unicast Packets Transmit Count Low** Counts number of unicast packets transmitted. Lower 32 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.86Switch Unicast Packets Transmit Count High - GLSW_UPTCH[n] (0x00340004 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
UPTCH	15:0	0x0	RW1C	**Unicast Packets Transmit Count High** Counts number of unicast packets transmitted. Upper 16 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.87Switch Multicast Packets Transmit Count Low - GLSW_MPTCL[n] (0x00340080 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
MPTCL	31:0	0x0	RW1C	**Multicast Packets Transmit Count Low** Counts number of multicast packets transmitted. Lower 32 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.88Switch Multicast Packets Transmit Count High - GLSW_MPTCH[n] (0x00340084 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
MPTCH	15:0	0x0	RW1C	**Multicast Packets Transmit Count High** Counts number of multicast packets transmitted. Upper 16 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.89Switch Broadcast Packets Transmit Count Low - GLSW_BPTCL[n] (0x00340100 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
BPTCL	31:0	0x0	RW1C	**Broadcast Packets Transmit Count Low** Counts number of broadcast packets transmitted. Lower 32 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.90Switch Broadcast Packets Transmit Count High - GLSW_BPTCH[n] (0x00340104 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
BPTCH	15:0	0x0	RW1C	**Broadcast Packets Transmit Count High** Counts number of broadcast packets transmitted. Upper 16 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.91VSI Transmit Error Packet Count - GLV_TEPC[VSI] (0x00344000 + 0x4*VSI, VSI=0...383; RW1C)

Field	Bit(s)	Init.	Type	Description
TEPC	31:0	0x0	RW1C	**Transmit Error Packet Count**

38.39.2.16.92Switch Transmit Packets Discarded Count - GLSW_TDPC[n] (0x00348000 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
TDPC	31:0	0x0	RW1C	**Transmit Discard Packet Count**

38.39.2.16.93VSI Good Octets Received Count Low - GLV_GORCL[n] (0x00358000 + 0x8*n, n=0...383; RW1C)

Field	Bit(s)	Init.	Type	Description
GORCL	31:0	0x0	RW1C	**Good Octets Received Count Low** Counts number of bytes received by this VSI. Lower 32 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.94VSI Good Octets Received Count High - GLV_GORCH[n] (0x00358004 + 0x8*n, n=0...383; RW1C)

Field	Bit(s)	Init.	Type	Description
GORCH	15:0	0x0	RW1C	**Good Octets Received Count High** Counts number of bytes received by this VSI. Upper 16 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.95Switch Good Octets Received Count Low - GLSW_GORCL[n] (0x0035C000 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
GORCL	31:0	0x0	RW1C	**Good Octets Received Count Low** Counts number of bytes received. Lower 32 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.96Switch Good Octets Received Count High - GLSW_GORCH[n] (0x0035C004 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
GORCH	15:0	0x0	RW1C	**Good Octets Received Count High** Counts number of bytes received. Upper 16 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.97VEB VLAN Receive Byte Count Low - GLVEBVL_GORCL[n] (0x00360000 + 0x8*n, n=0...127; RW1C)

Field	Bit(s)	Init.	Type	Description
VLBCL	31:0	0x0	RW1C	**VEB per VLAN Byte Count Low** The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.98VEB VLAN Receive Byte Count High - GLVEBVL_GORCH[n] (0x00360004 + 0x8*n, n=0...127; RW1C)

Field	Bit(s)	Init.	Type	Description
VLBCH	15:0	0x0	RW1C	**VEB per VLAN Byte Count High** The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.99VEB TC Receive Byte Count Low - GLVEBTC_RBCL[n,m] (0x00364000 + 0x8*n + 0x40*m, n=0...7, m=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
TCBCL	31:0	0x0	RW1C	**VEB per TC Byte Count Low** The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.100VEB TC Receive Byte Count High - GLVEBTC_RBCH[n,m] (0x00364004 + 0x8*n + 0x40*m, n=0...7, m=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
TCBCH	15:0	0x0	RW1C	**VEB per TC Byte Count High** The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.101VEB TC Receive Packet Count Low - GLVEBTC_RPCL[n,m] (0x00368000 + 0x8*n + 0x40*m, n=0...7, m=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPCL	31:0	0x0	RW1C	**VEB per TC Packet Count Low** The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.102VEB TC Receive Packet Count High - GLVEBTC_RPCH[n,m] (0x00368004 + 0x8*n + 0x40*m, n=0...7, m=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPCH	15:0	0x0	RW1C	**VEB per TC Packet Count High** The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.103VSI Unicast Packets Received Count Low - GLV_UPRCL[n] (0x0036C000 + 0x8*n, n=0...383; RW1C)

Field	Bit(s)	Init.	Type	Description
UPRCL	31:0	0x0	RW1C	**Unicast Packets Received Count Low** Counts number of unicast packets received by this VSI. Lower 32 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.104VSI Unicast Packets Received Count High - GLV_UPRCH[n] (0x0036C004 + 0x8*n, n=0...383; RW1C)

Field	Bit(s)	Init.	Type	Description
UPRCH	15:0	0x0	RW1C	**Unicast Packets Received Count High** Counts number of unicast packets received by this VSI. Upper 16 bits. It is implemented internally breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.105VSI Multicast Packets Received Count Low - GLV_MPRCL[n] (0x0036CC00 + 0x8*n, n=0...383; RW1C)

Field	Bit(s)	Init.	Type	Description
MPRCL	31:0	0x0	RW1C	**Multicast Packets Received Count Low** Counts number of multicast packets received by this VSI. Lower 32 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.106VSI Multicast Packets Received Count High - GLV_MPRCH[n] (0x0036CC04 + 0x8*n, n=0...383; RW1C)

Field	Bit(s)	Init.	Type	Description
MPRCH	15:0	0x0	RW1C	**Multicast Packets Received Count High** Counts number of multicast packets received by this VSI. Upper 16 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.107VSI Broadcast Packets Received Count Low - GLV_BPRCL[n] (0x0036D800 + 0x8*n, n=0...383; RW1C)

Field	Bit(s)	Init.	Type	Description
BPRCL	31:0	0x0	RW1C	**Broadcast Packets Received Count Low** Counts number of broadcast packets received by this VSI. Lower 32 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.108VSI Broadcast Packets Received Count High - GLV_BPRCH[n] (0x0036D804 + 0x8*n, n=0...383; RW1C)

Field	Bit(s)	Init.	Type	Description
BPRCH	15:0	0x0	RW1C	**Broadcast Packets Received Count High** Counts number of broadcast packets received by this VSI. Upper 16 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.109VSI Received Unknown Packet Protocol Count - GLV_RUPP[n] (0x0036E400 + 0x8*n, n=0...383; RW1C)

Field	Bit(s)	Init.	Type	Description
RUPP	31:0	0x0	RW1C	**Received Unknown Packet Protocol count** Receive Packets dropped because of an a unknown protocol or no forward destination.

38.39.2.16.110Switch Unicast Packets Received Count Low - GLSW_UPRCL[n] (0x00370000 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
UPRCL	31:0	0x0	RW1C	**Unicast Packets Received Count Low** Counts number of unicast packets received. Lower 32 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.111Switch Unicast Packets Received Count High - GLSW_UPRCH[n] (0x00370004 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
UPRCH	15:0	0x0	RW1C	**Unicast Packets Received Count High** Counts number of unicast packets received. Upper 16 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.112Switch Multicast Packets Received Count Low - GLSW_MPRCL[n] (0x00370080 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
MPRCL	31:0	0x0	RW1C	**Multicast Packets Received Count Low** Counts number of multicast packets received. Lower 32 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.113Switch Multicast Packets Received Count High - GLSW_MPRCH[n] (0x00370084 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
MPRCH	15:0	0x0	RW1C	**Multicast Packets Received Count High** Counts number of multicast packets received. Upper 16 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.114Switch Broadcast Packets Received Count Low - GLSW_BPRCL[n] (0x00370100 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
BPRCL	31:0	0x0	RW1C	**Broadcast Packets Received Count Low** Counts number of broadcast packets received. Lower 32 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.115Switch Broadcast Packets Received Count High - GLSW_BPRCH[n] (0x00370104 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
BPRCH	15:0	0x0	RW1C	**Broadcast Packets Received Count High** Counts number of broadcast packets received. Upper 16 bits. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.116Switch Received Unknown Packet Protocol Count - GLSW_RUPP[n] (0x00370180 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
RUPP	31:0	0x0	RW1C	**Received Unknown Packet Protocol count** Receive Packets dropped because of an a unknown protocol or no forward destination.

38.39.2.16.117VEB VLAN Unicast Packet Count Low - GLVEBVL_UPCL[n] (0x00374000 + 0x8*n, n=0...127; RW1C)

Field	Bit(s)	Init.	Type	Description
VLUPCL	31:0	0x0	RW1C	**VEB per VLAN Unicast Packet Count Low** The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.118VEB VLAN Unicast Packet Count High - GLVEBVL_UPCH[n] (0x00374004 + 0x8*n, n=0...127; RW1C)

Field	Bit(s)	Init.	Type	Description
VLUPCH	15:0	0x0	RW1C	**VEB per VLAN Unicast Packet Count High** The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.119VEB VLAN Multicast Packet Count Low - GLVEBVL_MPCL[n] (0x00374400 + 0x8*n, n=0...127; RW1C)

Field	Bit(s)	Init.	Type	Description
VLMPCL	31:0	0x0	RW1C	**VEB per VLAN Multicast Packet Count Low** The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.120VEB VLAN Multicast Packet Count High - GLVEBVL_MPCH[n] (0x00374404 + 0x8*n, n=0...127; RW1C)

Field	Bit(s)	Init.	Type	Description
VLMPCH	15:0	0x0	RW1C	**VEB per VLAN Multicast Packet Count High** The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.16.121VEB VLAN Broadcast Packet Count Low - GLVEBVL_BPCL[n] (0x00374800 + 0x8*n, n=0...127; RW1C)

Field	Bit(s)	Init.	Type	Description
VLBPCL	31:0	0x0	RW1C	**VEB per VLAN Broadcast Packet Count Low** The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.

38.39.2.16.122VEB VLAN Broadcast Packet Count High - GLVEBVL_BPCH[n] (0x00374804 + 0x8*n, n=0...127; RW1C)

Field	Bit(s)	Init.	Type	Description
VLBPCH	15:0	0x0	RW1C	**VEB per VLAN Broadcast Packet Count High** The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17 PF - Protocol Engine Statistics Registers

38.39.2.17.1 Protocol engine Statistics Received VLAN_ID Errors - GLPES_PFRXVLANERR[n] (0x00010000 + 0x4*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
RXVLANERR	23:0	0x0	RW1C	**Received VLAN Errors** Counts the number of packets received by the PE with incorrect VLAN_ID.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.17.2 Protocol engine Statistics IPv4 Received Octets Low - GLPES_PFIP4RXOCTSLO[n] (0x00010200 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP4RXOCTSLO	31:0	0x0	RW1C	**IPv4 Received Octets Low** Counts the number of IPv4 octets received by the PE. This is the lower 32 bits of the 48-bit counter. This counter is not incremented for received IPv4 multicast packets. Software must add the value of this counter to the IPv4 multicast octet counter to calculate a total number of octets received. The low and high registers are part of a 64 bit register, and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.3 Protocol Engine Statistics IPv4 Received Octets High - GLPES_PFIP4RXOCTSHI[n] (0x00010204 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP4RXOCTSHI	15:0	0x0	RW1C	**IPv4 Received Octets High** Counts the number of IPv4 octets received by the PE. This is the upper 16 bits of the 48-bit counter. This counter is not incremented for received IPv4 multicast packets. Software must add the value of this counter to the IPv4 multicast octet counter to calculate a total number of octets received. The low and high registers are part of a 64 bit register, and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.4 Protocol Engine Statistics IPv4 Received Packets Low - GLPES_PFIP4RXPKTSLO[n] (0x00010400 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP4RXPKTSLO	31:0	0x0	RW1C	**IPv4 Received Packets Low** Counts the number of IPv4 packets received by the PE. This is the lower 32 bits of the 48-bit counter. This counter is not incremented for received IPv4 multicast packets. Software must add the value of this counter to the IPv4 multicast octet packet to calculate a total number of octets received. The low and high registers are part of a 64 bit register, and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.5 Protocol Engine Statistics IPv4 Received Packets High - GLPES_PFIP4RXPKTSHI[n] (0x00010404 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP4RXPKTSHI	15:0	0x0	RW1C	**IPv4 Received Packets High** Counts the number of IPv4 packets received by the PE. This is the upper 16 bits of the 48-bit counter. This counter is not incremented for received IPv4 multicast packets. Software must add the value of this counter to the IPv4 multicast octet packet to calculate a total number of octets received. The low and high registers are part of a 64 bit register, and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.6 Protocol Engine Statistics IPv4 Discards - GLPES_PFIP4RXDISCARD[n] (0x00010600 + 0x4*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP4RXDISCARD	31:0	0x0	RW1C	**IPv4 Received Discards** Counts the number of IPv4 packets received by the PE without errors and are discarded.

38.39.2.17.7 Protocol Engine Statistics IPv4 Truncated Packets - GLPES_PFIP4RXTRUNC[n] (0x00010700 + 0x4*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP4RXTRUNC	31:0	0x0	RW1C	**IPv4 Received Truncated** Counts the number of IPv4 packets received by the PE and truncated due to insufficient payload or header buffering space in RQ descriptors.

38.39.2.17.8 Protocol Engine Statistics IPv4 Received Fragments Low - GLPES_PFIP4RXFRAGSLO[n] (0x00010800 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP4RXFRAGSLO	31:0	0x0	RW1C	**IPv4 Received Fragments Low** Counts the number of IPv4 fragments received by the PE. This is the lower 32 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.9 Protocol Engine Statistics IPv4 Received Fragments High - GLPES_PFIP4RXFRAGSHI[n] (0x00010804 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP4RXFRAGSHI	15:0	0x0	RW1C	**IPv4 Received Fragments High** Counts the number of IPv4 fragments received by the PE. This is the upper 16 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.10 Protocol Engine Statistics IPv4 Received Multicast Octets Low - GLPES_PFIP4RXMCOCTSLO[n] (0x00010A00 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP4RXMCOCTSLO	31:0	0x0	RW1C	**IPv4 Received Multicast Octets Low** Counts the number of IPv4 multicast octets received by the PE. This is the lower 32 bits of the 48-bit counter. This register does not count number of octets of the multicast packets replicated inside the PE. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.11 Protocol Engine Statistics IPv4 Received Multicast Octets High - GLPES_PFIP4RXMCOCTSHI[n] (0x00010A04 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP4RXMCOCTSHI	15:0	0x0	RW1C	**IPv4 Received Multicast Octets High** Counts the number of IPv4 multicast octets received by the PE. This is the upper 16 bits of the 48-bit counter. This register does not count number of octets of the multicast packets replicated inside the PE. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.12 Protocol Engine Statistics IPv4 Received Multicast Packets Low - GLPES_PFIP4RXMCPKTSLO[n] (0x00010C00 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP4RXMCPKTSLO	31:0	0x0	RW1C	**IPv4 Received Multicast Packets Low** Counts the number of IPv4 multicast packets received by the PE. This is the lower 32 bits of the 48-bit counter. This register does not count number of multicast packets replicated inside the PE. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.13 Protocol Engine Statistics IPv4 Received Multicast Packets High - GLPES_PFIP4RXMCPKTSHI[n] (0x00010C04 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP4RXMCPKTSHI	15:0	0x0	RW1C	**IPv4 Received Multicast Packets High** Counts the number of IPv4 multicast packets received by the PE. This is the upper 16 bits of the 48-bit counter. This register does not count number of multicast packets replicated inside the PE. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.14 Protocol Engine Statistics IPv6 Received Octets Low - GLPES_PFIP6RXOCTSLO[n] (0x00010E00 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP6RXOCTSLO	31:0	0x0	RW1C	**IPv6 Received Octets Low** Counts the number of IPv6 octets received by the PE. This is the lower 32 bits of the 48-bit counter. This counter is not incremented for received IPv6 multicast packets. Software must add the value of this counter to the IPv6 Multicast Octet counter to calculate a total number of octets received. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.15 Protocol Engine Statistics IPv6 Received Octets High - GLPES_PFIP6RXOCTSHI[n] (0x00010E04 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP6RXOCTSHI	15:0	0x0	RW1C	**IPv6 Received Octets High** Counts the number of IPv6 octets received by the PE. This is the upper 16 bits of the 48-bit counter. This counter is not incremented for received IPv6 multicast packets. Software must add the value of this counter to the IPv6 Multicast Octet counter to calculate a total number of octets received. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.16 Protocol Engine Statistics IPv6 Received Packets Low - GLPES_PFIP6RXPKTSLO[n] (0x00011000 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP6RXPKTSLO	31:0	0x0	RW1C	**IPv6 Received Packets Low** Counts the number of IPv6 packets received by the PE. This is the lower 32 bits of the 48-bit counter. This counter is not incremented for received IPv6 multicast packets. Software must add a value of this counter to the IPv6 Multicast Packet counter to calculate a total number of packets received. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.17 Protocol Engine Statistics IPv6 Received Packets High - GLPES_PFIP6RXPKTSHI[n] (0x00011004 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP6RXPKTSHI	15:0	0x0	RW1C	**IPv6 Received Packets High** Counts the number of IPv6 packets received by the PE. This is the upper 16 bits of the 48-bit counter. This counter is not incremented for received IPv6 multicast packets. Software must add a value of this counter to the IPv6 Multicast Packet counter to calculate a total number of packets received. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.18 Protocol Engine Statistics IPv6 Discards - GLPES_PFIP6RXDISCARD[n] (0x00011200 + 0x4*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP6RXDISCARD	31:0	0x0	RW1C	**IPv6 Received Discards** Counts the number of IPv6 packets received by the PE without errors and are discarded.

38.39.2.17.19 Protocol Engine Statistics IPv6 Truncated Packets - GLPES_PFIP6RXTRUNC[n] (0x00011300 + 0x4*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP6RXTRUNC	31:0	0x0	RW1C	**IPv6 Received Truncated** Counts the number of IPv6 packets received by the PE and truncated due to insufficient payload or header buffering space in RQ descriptors.

38.39.2.17.20 Protocol Engine Statistics IPv6 Received Fragments Low - GLPES_PFIP6RXFRAGSLO[n] (0x00011400 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP6RXFRAGSLO	31:0	0x0	RW1C	**IPv6 Received Fragments Low** Counts the number of IPv6 fragments received by the PE. This is the lower 32 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.21 Protocol Engine Statistics IPv6 Received Fragments High - GLPES_PFIP6RXFRAGSHI[n] (0x00011404 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP6RXFRAGSHI	15:0	0x0	RW1C	**IPv6 Received Fragments High** Counts the number of IPv6 fragments received by the PE. This is the upper 16 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.22 Protocol Engine Statistics IPv6 Received Multicast Octets Low - GLPES_PFIP6RXMCOCTSLO[n] (0x00011600 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP6RXMCOCTSLO	31:0	0x0	RW1C	**IPv6 Received Multicast Octets Low** Counts the number of IPv6 multicast octets received by the PE. This is the lower 32 bits of the 48-bit counter. This register does not count number of octets of the multicast packets replicated inside the PE. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.23 Protocol Engine Statistics IPv6 Received Multicast Octets High - GLPES_PFIP6RXMCOCTSHI[n] (0x00011604 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP6RXMCOCTSHI	15:0	0x0	RW1C	**IPv6 Received Multicast Octets High** Counts the number of IPv6 multicast octets received by the PE. This is the upper 16 bits of the 48-bit counter. This register does not count number of octets of the multicast packets replicated inside the PE. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.24 Protocol Engine Statistics IPv6 Received Multicast Packets Low - GLPES_PFIP6RXMCPKTSLO[n] (0x00011800 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP6RXMCPKTSLO	31:0	0x0	RW1C	**IPv6 Received Multicast Packets Low** Counts the number of IPv6 multicast packets received by the PE. This is the lower 32 bits of the 48-bit counter. This register does not count number of multicast packets replicated inside the PE. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.25 Protocol Engine Statistics IPv6 Received Multicast Packets High - GLPES_PFIP6RXMCPKTSHI[n] (0x00011804 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP6RXMCPKTSHI	15:0	0x0	RW1C	**IPv6 Received Multicast Packets High** Counts the number of IPv6 multicast packets received by the PE. This is the upper 16 bits of the 48-bit counter. This register does not count number of multicast packets replicated inside the PE. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.26 Protocol Engine Statistics IPv4 Transmitted Octets Low - GLPES_PFIP4TXOCTSLO[n] (0x00011A00 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP4TXOCTSLO	31:0	0x0	RW1C	**IPv4 Transmitted Octets Low** Counts the number of IPv4 octets supplied by the PE to the lower layers for transmission. This is the lower 32 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.27 Protocol Engine Statistics IPv4 Transmitted Octets High - GLPES_PFIP4TXOCTSHI[n] (0x00011A04 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP4TXOCTSHI	15:0	0x0	RW1C	**IPv4 Transmitted Octets High** Counts the number of IPv4 octets supplied by the PE to the lower layers for transmission. This is the upper 16 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.28 Protocol Engine Statistics IPv4 Transmitted Packets Low - GLPES_PFIP4TXPKTSLO[n] (0x00011C00 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP4TXPKTSLO	31:0	0x0	RW1C	**IPv4 Transmitted Packets Low** Counts the number of IPv4 packets supplied by the PE to the lower layers for transmission. This is the lower 32 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.29 Protocol Engine Statistics IPv4 Transmitted Packets High - GLPES_PFIP4TXPKTSHI[n] (0x00011C04 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP4TXPKTSHI	15:0	0x0	RW1C	**IPv4 Transmitted Packets High** Counts the number of IPv4 packets supplied by the PE to the lower layers for transmission. This is the upper 16 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.30 Protocol Engine Statistics IPv4 Transmitted Fragments Low - GLPES_PFIP4TXFRAGSLO[n] (0x00011E00 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP4TXFRAGSLO	31:0	0x0	RW1C	**IPv4 Transmitted Fragments Low** Counts the number of IPv4 fragments supplied by the PE to the lower layers for transmission. This is the lower 32 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.31 Protocol Engine Statistics IPv4 Transmitted Fragments High - GLPES_PFIP4TXFRAGSHI[n] (0x00011E04 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP4TXFRAGSHI	15:0	0x0	RW1C	**IPv4 Transmitted Fragments High** Counts the number of IPv4 fragments supplied by the PE to the lower layers for transmission. This is the upper 16 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.32 Protocol Engine Statistics IPv4 Transmitted Multicast Octets Low - GLPES_PFIP4TXMCOCTSLO[n] (0x00012000 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP4TXMCOCTSLO	31:0	0x0	RW1C	**IPv4 Transmitted Multicast Octets Low** Counts the number of IPv4 multicast octets supplied by the PE to the lower layers for transmission. This is the lower 32 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.33 Protocol Engine Statistics IPv4 Transmitted Multicast Octets High - GLPES_PFIP4TXMCOCTSHI[n] (0x00012004 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP4TXMCOCTSHI	15:0	0x0	RW1C	**IPv4 Transmitted Multicast Octets High** Counts the number of IPv4 multicast octets supplied by the PE to the lower layers for transmission. This is the upper 16 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.34 Protocol Engine Statistics IPv4 Transmitted Multicast Packets Low - GLPES_PFIP4TXMCPKTSLO[n] (0x00012200 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP4TXMCPKTSLO	31:0	0x0	RW1C	**IPv4 Transmitted Multicast Packets Low** Counts the number of IPv4 multicast packets supplied by the PE to the lower layers for transmission. This is the lower 32 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.35 Protocol Engine Statistics IPv4 Transmitted Multicast Packets High - GLPES_PFIP4TXMCPKTSHI[n] (0x00012204 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP4TXMCPKTSHI	15:0	0x0	RW1C	**IPv4 Transmitted Multicast Packets High** Counts the number of IPv4 multicast packets supplied by the PE to the lower layers for transmission. This is the upper 16 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.36 Protocol Engine Statistics IPv6 Transmitted Octets Low - GLPES_PFIP6TXOCTSLO[n] (0x00012400 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP6TXOCTSLO	31:0	0x0	RW1C	**IPv6 Transmitted Octets Low** Counts the number of IPv6 octets supplied by the PE to the lower layers for transmission. This is the lower 32 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.37 Protocol Engine Statistics IPv6 Transmitted Octets High - GLPES_PFIP6TXOCTSHI[n] (0x00012404 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP6TXOCTSHI	15:0	0x0	RW1C	**IPv6 Transmitted Octets High** Counts the number of IPv6 octets supplied by the PE to the lower layers for transmission. This is the upper 16 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.38 Protocol Engine Statistics IPv6 Transmitted Packets Low - GLPES_PFIP6TXPKTSLO[n] (0x00012600 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP6TXPKTSLO	31:0	0x0	RW1C	**IPv6 Transmitted Packets Low** Counts the number of IPv6 packets supplied by the PE to the lower layers for transmission. This is the lower 32 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.39 Protocol Engine Statistics IPv6 Transmitted Packets High - GLPES_PFIP6TXPKTSHI[n] (0x00012604 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP6TXPKTSHI	15:0	0x0	RW1C	**IPv6 Transmitted Packets High** Counts the number of IPv6 packets supplied by the PE to the lower layers for transmission. This is the upper 16 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.40 Protocol Engine Statistics IPv6 Transmitted Fragments Low - GLPES_PFIP6TXFRAGSLO[n] (0x00012800 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP6TXFRAGSLO	31:0	0x0	RW1C	**IPv6 Transmitted Fragments Low** Counts the number of IPv6 fragments supplied by the PE to the lower layers for transmission. This is the lower 32 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.41 Protocol Engine Statistics IPv6 Transmitted Fragments High - GLPES_PFIP6TXFRAGSHI[n] (0x00012804 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP6TXFRAGSHI	15:0	0x0	RW1C	**IPv6 Transmitted Fragments High** Counts the number of IPv6 fragments supplied by the PE to the lower layers for transmission. This is the upper 16 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.42 Protocol Engine Statistics IPv6 Transmitted Multicast Octets Low - GLPES_PFIP6TXMCOCTSLO[n] (0x00012A00 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP6TXMCOCTSLO	31:0	0x0	RW1C	**IPv6 Transmitted Multicast Octets Low** Counts the number of IPv6 multicast octets supplied by the PE to the lower layers for transmission. This is the lower 32 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.43 Protocol Engine Statistics IPv6 Transmitted Multicast Octets High - GLPES_PFIP6TXMCOCTSHI[n] (0x00012A04 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP6TXMCOCTSHI	15:0	0x0	RW1C	**IPv6 Transmitted Multicast Octets High** Counts the number of IPv6 multicast octets supplied by the PE to the lower layers for transmission. This is the upper 16 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.44 Protocol Engine Statistics IPv6 Transmitted Multicast Packets Low - GLPES_PFIP6TXMCPKTSLO[n] (0x00012C00 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP6TXMCPKTSLO	31:0	0x0	RW1C	**IPv6 Transmitted Multicast Packets Low** Counts the number of IPv6 multicast packets supplied by the PE to the lower layers for transmission. This is the lower 32 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.45 Protocol Engine Statistics IPv6 Transmitted Multicast Packets High - GLPES_PFIP6TXMCPKTSHI[n] (0x00012C04 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP6TXMCPKTSHI	15:0	0x0	RW1C	**IPv6 Transmitted Multicast Packets High** Counts the number of IPv6 multicast packets supplied by the PE to the lower layers for transmission. This is the upper 16 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.46 Protocol Engine Statistics IPv4 Discarded No Route Packets - GLPES_PFIP4TXNOROUTE[n] (0x00012E00 + 0x4*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP4TXNOROUTE	23:0	0x0	RW1C	**IPv4 Transmitted No Route** Counts the number of IPv4 packets discarded due to routing problem (no hit in ARP table).
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.17.47 Protocol engine Statistics IPv6 Discarded No Route Packets - GLPES_PFIP6TXNOROUTE[n] (0x00012F00 + 0x4*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
IP6TXNOROUTE	23:0	0x0	RW1C	**IPv6 Transmitted No Route** Counts the number of IPv6 packets discarded due to routing problem (no hit in ARP table).
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.17.48 Protocol Engine Statistics TCP Received Segments Low - GLPES_PFTCPRXSEGSLO[n] (0x00013000 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPRXSEGSLO	31:0	0x0	RW1C	**TCP Received Segments Low** Counts the number of TCP segments received by the PE. This is the lower 32 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.49 Protocol Engine Statistics TCP Received Segments High - GLPES_PFTCPRXSEGSHI[n] (0x00013004 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPRXSEGSHI	15:0	0x0	RW1C	**TCP Received Segments High** Counts the number of TCP segments received by the PE. This is the upper 16 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.50 Protocol Engine Statistics TCP Received Segments with Unsupported Options - GLPES_PFTCPRXOPTERR[n] (0x00013200 + 0x4*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPRXOPTERR	23:0	0x0	RW1C	**TCP Received Options Errors** Counts the number of TCP segments received by the PE with unsupported TCP options and TCP option length errors.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.17.51 Protocol Engine Statistics TCP Dropped Segments Due to Protocol Errors - GLPES_PFTCPRXPROTOERR[n] (0x00013300 + 0x4*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPRXPROTOERR	23:0	0x0	RW1C	**TCP Received Protocol Errors** Counts the number of TCP segments received and dropped by the PE due to TCP protocol errors.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.17.52 Protocol Engine Statistics TCP Transmitted Segments Low - GLPES_PFTCPTXSEGLO[n] (0x00013400 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPTXSEGLO	31:0	0x0	RW1C	**TCP Transmitted Segments Low** Counts the number of TCP segments supplied by the PE to the lower layers for transmission. This is the lower 32 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.53 Protocol Engine Statistics TCP Transmitted Segments High - GLPES_PFTCPTXSEGHI[n] (0x00013404 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPTXSEGHI	15:0	0x0	RW1C	**TCP Transmitted Segments High** Counts the number of TCP segments supplied by the PE to the lower layers for transmission. This is the upper 16 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.54 Protocol Engine Statistics TCP Retransmitted Segments - GLPES_PFTCPRTXSEG[n] (0x00013600 + 0x4*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPRTXSEG	31:0	0x0	RW1C	**TCP Retransmitted Segments** Counts the number of TCP segments retransmitted by the PE.

38.39.2.17.55 Protocol Engine Statistics UDP Received Packets Low - GLPES_PFUDPRXPKTSLO[n] (0x00013800 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
UDPRXPKTSLO	31:0	0x0	RW1C	**UDP Received Packets Low** Counts the number of UDP packets received by the PE without errors. This is the lower 32 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.56 Protocol Engine Statistics UDP Received Packets High - GLPES_PFUDPRXPKTSHI[n] (0x00013804 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
UDPRXPKTSHI	15:0	0x0	RW1C	**UDP Received Packets High** Counts the number of UDP packets received by the PE without errors. This is the upper 16 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.57 Protocol Engine Statistics UDP Transmitted Packets Low - GLPES_PFUDPTXPKTSLO[n] (0x00013A00 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
UDPTXPKTSLO	31:0	0x0	RW1C	**UDP Transmitted Packets Low** Counts the number of UDP packets submitted by the PE to the lower layers for transmission. This is the lower 32 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.58 Protocol Engine Statistics UDP Transmitted Packets High - GLPES_PFUDPTXPKTSHI[n] (0x00013A04 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
UDPTXPKTSHI	15:0	0x0	RW1C	**UDP Transmitted Packets High** Counts the number of UDP packets submitted by the PE to the lower layers for transmission. This is the upper 16 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.59 Protocol Engine Statistics RDMA Received Write Messages Low - GLPES_PFRDMARXWRSLO[n] (0x00013C00 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
RDMARXWRSLO	31:0	0x0	RW1C	**RDMA Received Writes Low** Counts the number of RDMA write messages received by the PE. This is the lower 32 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.60 Protocol Engine Statistics RDMA Received Write Messages High - GLPES_PFRDMARXWRSHI[n] (0x00013C04 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
RDMARXWRSHI	15:0	0x0	RW1C	**RDMA Received Writes High** Counts the number of RDMA write messages received by the PE. This is the upper 16 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.61 Protocol Engine Statistics RDMA Received Read Request Messages Low - GLPES_PFRDMARXRDSLO[n] (0x00013E00 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
RDMARXRDSLO	31:0	0x0	RW1C	**RDMA Received Reads Low** Counts the number of RDMA read request messages received by the PE. This is the lower 32 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.62 Protocol Engine Statistics RDMA Received Read Request Messages High - GLPES_PFRDMARXRDSHI[n] (0x00013E04 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
RDMARXRDSHI	15:0	0x0	RW1C	**RDMA Received Reads High** Counts the number of RDMA read request messages received by the PE. This is the upper 16 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.63 Protocol Engine Statistics RDMA Received Send Messages Low - GLPES_PFRDMARXSNDSLO[n] (0x00014000 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
RDMARXSNDSLO	31:0	0x0	RW1C	**RDMA Received Sends Low** Counts the number of RDMA send messages received by the PE. This is the lower 32 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.64 Protocol Engine Statistics RDMA Received Send Messages High - GLPES_PFRDMARXSNDSHI[n] (0x00014004 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
RDMARXSNDSHI	15:0	0x0	RW1C	**RDMA Received Sends High** Counts the number of RDMA send messages received by the PE. This is the upper 16 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.65 Protocol Engine Statistics RDMA Transmitted Write Messages Low - GLPES_PFRDMATXWRSLO[n] (0x00014200 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
RDMATXWRSLO	31:0	0x0	RW1C	**RDMA Transmitted Writes Low** Counts the number of RDMA write messages transmitted by the PE. This is the lower 32 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.66 Protocol Engine Statistics RDMA Transmitted Write Messages High - GLPES_PFRDMATXWRSHI[n] (0x00014204 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
RDMATXWRSHI	15:0	0x0	RW1C	**RDMA Transmitted Writes High** Counts the number of RDMA write messages transmitted by the PE. This is the upper 16 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.67 Protocol Engine Statistics RDMA Transmitted Read Request Messages Low - GLPES_PFRDMATXRDSLO[n] (0x00014400 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
RDMATXRDSLO	31:0	0x0	RW1C	**RDMA Transmitted Reads Low** Counts the number of RDMA read request messages transmitted by the PE. This is the lower 32 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.68 Protocol Engine Statistics RDMA Transmitted Read Request Messages High - GLPES_PFRDMATXRDSHI[n] (0x00014404 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
RDMATXRDSHI	15:0	0x0	RW1C	**RDMA Transmitted Reads High** Counts the number of RDMA read request messages transmitted by the PE. This is the upper 16 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.69 Protocol Engine Statistics RDMA Transmitted Send Messages Low - GLPES_PFRDMATXSNDSLO[n] (0x00014600 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
RDMATXSNDSLO	31:0	0x0	RW1C	**RDMA Transmitted Sends Low** Counts the number of RDMA send messages transmitted by the PE. This is the lower 32 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.70 Protocol Engine Statistics RDMA Transmitted Send Messages High - GLPES_PFRDMATXSNDSHI[n] (0x00014604 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
RDMATXSNDSHI	15:0	0x0	RW1C	**RDMA Transmitted Sends High** Counts the number of RDMA send messages transmitted by the PE. This is the upper 16 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.71 Protocol Engine Statistics RDMA Verbs Bind Operations Low - GLPES_PFRDMAVBNDLO[n] (0x00014800 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
RDMAVBNDLO	31:0	0x0	RW1C	**RDMA Verbs Bind Low** Counts the total number of RDMA verb bind operations carried out by the PE. This is the lower 32 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.72 Protocol Engine Statistics RDMA Verbs Bind Operations High - GLPES_PFRDMAVBNDHI[n] (0x00014804 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
RDMAVBNDHI	15:0	0x0	RW1C	**RDMA Verbs Bind High** Counts the total number of RDMA verb bind operations carried out by the PE. This is the upper 16 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.73 Protocol Engine Statistics RDMA Verbs Invalidate Operations Low - GLPES_PFRDMAVINVLO[n] (0x00014A00 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
RDMAVINVLO	31:0	0x0	RW1C	**RDMA Verbs Invalidate Low** Counts the total number of RDMA verb invalidate operations carried out by the PE. This is the lower 32 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.

38.39.2.17.74 Protocol Engine Statistics RDMA Verbs Invalidate Operations High - GLPES_PFRDMAVINVHI[n] (0x00014A04 + 0x8*n, n=0...15; RW1C)

Field	Bit(s)	Init.	Type	Description
RDMAVINVHI	15:0	0x0	RW1C	**RDMA Verbs Invalidate High** Counts the total number of RDMA verb invalidate operations carried out by the PE. This is the upper 16 bits of the 48-bit counter. The low and high registers are part of a 64-bit register and are read using 64-bit read accesses only. It is implemented internally, breaking the read request into two 32-bit reads. Reading the low 32 bits latches the high 32 bits into a shadow register. Reading the high 32 bits returns the value in the shadow register.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.75 Protocol Engine Statistics Received VLAN_ID Errors - GLPES_VFRXVLANERR[n] (0x00018000 + 0x4*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
RXVLANERR	23:0	0x0	RW1C	**Received VLAN Errors** Counts the number of packets received by the PE with incorrect VLAN_ID.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.17.76 Protocol Engine Statistics IPv4 Received Octets Low - GLPES_VFIP4RXOCTSLO[n] (0x00018200 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP4RXOCTSLO	31:0	0x0	RW1C	**IPv4 Received Octets Low** Counts the number of IPv4 octets received by the PE. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.77 Protocol Engine Statistics IPv4 Received Octets High - GLPES_VFIP4RXOCTSHI[n] (0x00018204 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP4RXOCTSHI	15:0	0x0	RW1C	**IPv4 Received Octets High** Counts the number of IPv4 octets received by the PE. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.78 Protocol Engine Statistics IPv4 Received Packets Low - GLPES_VFIP4RXPKTSLO[n] (0x00018400 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP4RXPKTSLO	31:0	0x0	RW1C	**IPv4 Received Packets Low** Counts the number of IPv4 packets received by the PE. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.79 Protocol Engine Statistics IPv4 Received Packets High - GLPES_VFIP4RXPKTSHI[n] (0x00018404 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP4RXPKTSHI	15:0	0x0	RW1C	**IPv4 Received Packets High** Counts the number of IPv4 packets received by the PE. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.80 Protocol Engine Statistics IPv4 Discards - GLPES_VFIP4RXDISCARD[n] (0x00018600 + 0x4*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP4RXDISCARD	31:0	0x0	RW1C	**IPv4 Received Discards** Counts the number of IPv4 packets received by the PE without errors and discarded.

38.39.2.17.81 Protocol Engine Statistics IPv4 Truncated Packets - GLPES_VFIP4RXTRUNC[n] (0x00018700 + 0x4*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP4RXTRUNC	31:0	0x0	RW1C	**IPv4 Received Truncated** Counts the number of IPv4 packets received by the PE and truncated due to insufficient payload or header buffering space in RQ descriptors.

38.39.2.17.82 Protocol Engine Statistics IPv4 Received Fragments Low - GLPES_VFIP4RXFRAGSLO[n] (0x00018800 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP4RXFRAGSLO	31:0	0x0	RW1C	**IPv4 Received Fragments Low** Counts the number of IPv4 fragments received by the PE. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.83 Protocol Engine Statistics IPv4 Received Fragments High - GLPES_VFIP4RXFRAGSHI[n] (0x00018804 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP4RXFRAGSHI	15:0	0x0	RW1C	**IPv4 Received Fragments High** Counts the number of IPv4 fragments received by the PE. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.84 Protocol Engine Statistics IPv4 Received Multicast Octets Low - GLPES_VFIP4RXMCOCTSLO[n] (0x00018A00 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP4RXMCOCTSLO	31:0	0x0	RW1C	**IPv4 Received Multicast Octets Low** Counts the number of IPv4 multicast octets received by the PE. This is the lower 32 bits of the 48-bit counter. This register does not count number of octets of the multicast packets replicated inside the PE.

38.39.2.17.85 Protocol Engine Statistics IPv4 Received Multicast Octets High - GLPES_VFIP4RXMCOCTSHI[n] (0x00018A04 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP4RXMCOCTSHI	15:0	0x0	RW1C	**IPv4 Received Multicast Octets High** Counts the number of IPv4 multicast octets received by the PE. This is the upper 16 bits of the 48-bit counter. This register does not count number of octets of the multicast packets replicated inside the PE.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.86 Protocol Engine Statistics IPv4 Received Multicast Packets Low - GLPES_VFIP4RXMCPKTSLO[n] (0x00018C00 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP4RXMCPKTSLO	31:0	0x0	RW1C	**IPv4 Received Multicast Packets Low** Counts the number of IPv4 multicast packets received by the PE. This is the lower 32 bits of the 48-bit counter. This register does not count number of multicast packets replicated inside the PE.

38.39.2.17.87 Protocol Engine Statistics IPv4 Received Multicast Packets High - GLPES_VFIP4RXMCPKTSHI[n] (0x00018C04 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP4RXMCPKTSHI	15:0	0x0	RW1C	**IPv4 Received Multicast Packets High** Counts the number of IPv4 multicast packets received by the PE. This is the upper 16 bits of the 48-bit counter. This register does not count number of multicast packets replicated inside the PE.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.88 Protocol Engine Statistics IPv6 Received Octets Low - GLPES_VFIP6RXOCTSLO[n] (0x00018E00 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP6RXOCTSLO	31:0	0x0	RW1C	**IPv6 Received Octets Low** Counts the number of IPv6 octets received by the PE. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.89 Protocol Engine Statistics IPv6 Received Octets High - GLPES_VFIP6RXOCTSHI[n] (0x00018E04 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP6RXOCTSHI	15:0	0x0	RW1C	**IPv6 Received Octets High** Counts the number of IPv6 octets received by the PE. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.90 Protocol Engine Statistics IPv6 Received Packets Low - GLPES_VFIP6RXPKTSLO[n] (0x00019000 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP6RXPKTSLO	31:0	0x0	RW1C	**IPv6 Received Packets Low** Counts the number of IPv6 packets received by the PE. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.91 Protocol Engine Statistics IPv6 Received Packets High - GLPES_VFIP6RXPKTSHI[n] (0x00019004 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP6RXPKTSHI	15:0	0x0	RW1C	**IPv6 Received Packets High** Counts the number of IPv6 packets received by the PE. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.92 Protocol Engine Statistics IPv6 Discards - GLPES_VFIP6RXDISCARD[n] (0x00019200 + 0x4*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP6RXDISCARD	31:0	0x0	RW1C	**IPv6 Received Discards** Counts the number of IPv6 packets received by the PE without errors and discarded.

38.39.2.17.93 Protocol Engine Statistics IPv6 Truncated Packets - GLPES_VFIP6RXTRUNC[n] (0x00019300 + 0x4*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP6RXTRUNC	31:0	0x0	RW1C	**IPv6 Received Truncated** Counts the number of IPv6 packets received by the PE and truncated due to insufficient payload or header buffering space in RQ descriptors.

38.39.2.17.94 Protocol Engine Statistics IPv6 Received Fragments Low - GLPES_VFIP6RXFRAGSLO[n] (0x00019400 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP6RXFRAGSLO	31:0	0x0	RW1C	**IPv6 Received Fragments Low** Counts the number of IPv6 fragments received by the PE. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.95 Protocol Engine Statistics IPv6 Received Fragments High - GLPES_VFIP6RXFRAGSHI[n] (0x00019404 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP6RXFRAGSHI	15:0	0x0	RW1C	**IPv6 Received Fragments High** Counts the number of IPv6 fragments received by the PE. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.96 Protocol Engine Statistics IPv6 Received Multicast Octets Low - GLPES_VFIP6RXMCOCTSLO[n] (0x00019600 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP6RXMCOCTSLO	31:0	0x0	RW1C	**IPv6 Received Multicast Octets Low** Counts the number of IPv6 multicast octets received by the PE. This is the lower 32 bits of the 48-bit counter. This register does not count number of octets of the multicast packets replicated inside the PE.

38.39.2.17.97 Protocol Engine Statistics IPv6 Received Multicast Octets High - GLPES_VFIP6RXMCOCTSHI[n] (0x00019604 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP6RXMCOCTSHI	15:0	0x0	RW1C	**IPv6 Received Multicast Octets High** Counts the number of IPv6 multicast octets received by the PE. This is the upper 16 bits of the 48-bit counter. This register does not count number of octets of the multicast packets replicated inside the PE.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.98 Protocol Engine Statistics IPv6 Received Multicast Packets Low - GLPES_VFIP6RXMCPKTSLO[n] (0x00019800 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP6RXMCPKTSLO	31:0	0x0	RW1C	**IPv6 Received Multicast Packets Low** Counts the number of IPv6 multicast packets received by the PE. This is the lower 32 bits of the 48-bit counter. This register does not count number of multicast packets replicated inside the PE.

38.39.2.17.99 Protocol Engine Statistics IPv6 Received Multicast Packets High - GLPES_VFIP6RXMCPKTSHI[n] (0x00019804 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP6RXMCPKTSHI	15:0	0x0	RW1C	**IPv6 Received Multicast Packets High** Counts the number of IPv6 multicast packets received by the PE. This is the upper 16 bits of the 48-bit counter. This register does not count number of multicast packets replicated inside the PE.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.100 Protocol Engine Statistics IPv4 Transmitted Octets Low - GLPES_VFIP4TXOCTSLO[n] (0x00019A00 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP4TXOCTSLO	31:0	0x0	RW1C	**IPv4 Transmitted Octets Low** Counts the number of IPv4 octets supplied by the PE to the lower layers for transmission. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.101 Protocol Engine Statistics IPv4 Transmitted Octets High - GLPES_VFIP4TXOCTSHI[n] (0x00019A04 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP4TXOCTSHI	15:0	0x0	RW1C	**IPv4 Transmitted Octets High** Counts the number of IPv4 octets supplied by the PE to the lower layers for transmission. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.102 Protocol Engine Statistics IPv4 Transmitted Packets Low - GLPES_VFIP4TXPKTSLO[n] (0x00019C00 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP4TXPKTSLO	31:0	0x0	RW1C	**IPv4 Transmitted Packets Low** Counts the number of IPv4 packets supplied by the PE to the lower layers for transmission. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.103 Protocol Engine Statistics IPv4 Transmitted Packets High - GLPES_VFIP4TXPKTSHI[n] (0x00019C04 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP4TXPKTSHI	15:0	0x0	RW1C	**IPv4 Transmitted Packets High** Counts the number of IPv4 packets supplied by the PE to the lower layers for transmission. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.104 Protocol Engine Statistics IPv4 Transmitted Fragments Low - GLPES_VFIP4TXFRAGSLO[n] (0x00019E00 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP4TXFRAGSLO	31:0	0x0	RW1C	**IPv4 Transmitted Fragments Low** Counts the number of IPv4 fragments supplied by the PE to the lower layers for transmission. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.105 Protocol Engine Statistics IPv4 Transmitted Fragments High - GLPES_VFIP4TXFRAGSHI[n] (0x00019E04 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP4TXFRAGSHI	15:0	0x0	RW1C	**IPv4 Transmitted Fragments High** Counts the number of IPv4 fragments supplied by the PE to the lower layers for transmission. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.106 Protocol Engine Statistics IPv4 Transmitted Multicast Octets Low - GLPES_VFIP4TXMCOCTSLO[n] (0x0001A000 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP4TXMCOCTSLO	31:0	0x0	RW1C	**IPv4 Transmitted Multicast Octets Low** Counts the number of IPv4 multicast octets supplied by the PE to the lower layers for transmission. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.107 Protocol Engine Statistics IPv4 Transmitted Multicast Octets High - GLPES_VFIP4TXMCOCTSHI[n] (0x0001A004 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP4TXMCOCTSHI	15:0	0x0	RW1C	**IPv4 Transmitted Multicast Octets High** Counts the number of IPv4 multicast octets supplied by the PE to the lower layers for transmission. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.108 Protocol Engine Statistics IPv4 Transmitted Multicast Packets Low - GLPES_VFIP4TXMCPKTSLO[n] (0x0001A200 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP4TXMCPKTSLO	31:0	0x0	RW1C	**IPv4 Transmitted Multicast Packets Low** Counts the number of IPv4 multicast packets supplied by the PE to the lower layers for transmission. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.109 Protocol Engine Statistics IPv4 Transmitted Multicast Packets High - GLPES_VFIP4TXMCPKTSHI[n] (0x0001A204 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP4TXMCPKTSHI	15:0	0x0	RW1C	**IPv4 Transmitted Multicast Packets High** Counts the number of IPv4 multicast packets supplied by the PE to the lower layers for transmission. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.110 Protocol Engine Statistics IPv6 Transmitted Octets Low - GLPES_VFIP6TXOCTSLO[n] (0x0001A400 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP6TXOCTSLO	31:0	0x0	RW1C	**IPv6 Transmitted Octets Low** Counts the number of IPv6 octets supplied by the PE to the lower layers for transmission. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.111 Protocol Engine Statistics IPv6 Transmitted Octets High - GLPES_VFIP6TXOCTSHI[n] (0x0001A404 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP6TXOCTSHI	15:0	0x0	RW1C	**IPv6 Transmitted Octets High** Counts the number of IPv6 octets supplied by the PE to the lower layers for transmission. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.112 Protocol Engine Statistics IPv6 Transmitted Packets Low - GLPES_VFIP6TXPKTSLO[n] (0x0001A600 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP6TXPKTSLO	31:0	0x0	RW1C	**IPv6 Transmitted Packets Low** Counts the number of IPv6 packets supplied by the PE to the lower layers for transmission. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.113 Protocol Engine Statistics IPv6 Transmitted Packets High - GLPES_VFIP6TXPKTSHI[n] (0x0001A604 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP6TXPKTSHI	15:0	0x0	RW1C	**IPv6 Transmitted Packets High** Counts the number of IPv6 packets supplied by the PE to the lower layers for transmission. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.114 Protocol Engine Statistics IPv6 Transmitted Fragments Low - GLPES_VFIP6TXFRAGSLO[n] (0x0001A800 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP6TXFRAGSLO	31:0	0x0	RW1C	**IPv6 Transmitted Fragments Low** Counts the number of IPv6 fragments supplied by the PE to the lower layers for transmission. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.115 Protocol Engine Statistics IPv6 Transmitted Fragments High - GLPES_VFIP6TXFRAGSHI[n] (0x0001A804 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP6TXFRAGSHI	15:0	0x0	RW1C	**IPv6 Transmitted Fragments High** Counts the number of IPv6 fragments supplied by the PE to the lower layers for transmission. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.116 Protocol Engine Statistics IPv6 Transmitted Multicast Octets Low - GLPES_VFIP6TXMCOCTSLO[n] (0x0001AA00 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP6TXMCOCTSLO	31:0	0x0	RW1C	**IPv6 Transmitted Multicast Octets Low** Counts the number of IPv6 multicast octets supplied by the PE to the lower layers for transmission. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.117 Protocol Engine Statistics IPv6 Transmitted Multicast Octets High - GLPES_VFIP6TXMCOCTSHI[n] (0x0001AA04 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP6TXMCOCTSHI	15:0	0x0	RW1C	**IPv6 Transmitted Multicast Octets High** Counts the number of IPv6 multicast octets supplied by the PE to the lower layers for transmission. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.118 Protocol Engine Statistics IPv6 Transmitted Multicast Packets Low - GLPES_VFIP6TXMCPKTSLO[n] (0x0001AC00 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP6TXMCPKTSLO	31:0	0x0	RW1C	**IPv6 Transmitted Multicast Packets Low** Counts the number of IPv6 multicast packets supplied by the PE to the lower layers for transmission. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.119 Protocol Engine Statistics IPv6 Transmitted Multicast Packets High - GLPES_VFIP6TXMCPKTSHI[n] (0x0001AC04 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP6TXMCPKTSHI	15:0	0x0	RW1C	**IPv6 Transmitted Multicast Packets High** Counts the number of IPv6 multicast packets supplied by the PE to the lower layers for transmission. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.120 Protocol Engine Statistics IPv4 Discarded No Route Packets - GLPES_VFIP4TXNOROUTE[n] (0x0001AE00 + 0x4*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP4TXNOROUTE	23:0	0x0	RW1C	**IPv4 Transmitted No Route** Counts the number of IPv4 packets discarded due to routing problem (no hit in ARP table).
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.17.121 Protocol Engine Statistics IPv6 Discarded No Route Packets - GLPES_VFIP6TXNOROUTE[n] (0x0001AF00 + 0x4*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IP6TXNOROUTE	23:0	0x0	RW1C	**IPv6 Transmitted No Route** Counts the number of IPv6 packets discarded due to routing problem (no hit in ARP table).
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.17.122 Protocol Engine Statistics TCP Received Segments Low - GLPES_VFTCPRXSEGSLO[n] (0x0001B000 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
TCPRXSEGSLO	31:0	0x0	RW1C	**TCP Received Segments Low** Counts the number of TCP segments received by the PE. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.123 Protocol Engine Statistics TCP Received Segments High - GLPES_VFTCPRXSEGSHI[n] (0x0001B004 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
TCPRXSEGSHI	15:0	0x0	RW1C	**TCP Received Segments High** Counts the number of TCP segments received by the PE. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.124 Protocol Engine Statistics TCP Received Segments with Unsupported Options - GLPES_VFTCPRXOPTERR[n] (0x0001B200 + 0x4*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
TCPRXOPTERR	23:0	0x0	RW1C	**TCP Received Options Error** Counts the number of TCP segments received by the PE with unsupported TCP options and TCP option length errors.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.17.125 Protocol Engine Statistics TCP Dropped Segments Due to Protocol Errors - GLPES_VFTCPRXPROTOERR[n] (0x0001B300 + 0x4*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
TCPRXPROTOERR	23:0	0x0	RW1C	**TCP Received Protocol Error** Counts the number of TCP segments received and dropped by the PE due to TCP protocol errors.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.17.126 Protocol Engine Statistics TCP Transmitted Segments Low - GLPES_VFTCPTXSEGLO[n] (0x0001B400 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
TCPTXSEGLO	31:0	0x0	RW1C	**TCP Transmitted Segments Low** Counts the number of TCP segments supplied by the PE to the lower layers for transmission. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.127 Protocol Engine Statistics TCP Transmitted Segments High - GLPES_VFTCPTXSEGHI[n] (0x0001B404 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
TCPTXSEGHI	15:0	0x0	RW1C	**TCP Transmitted Segments High** Counts the number of TCP segments supplied by the PE to the lower layers for transmission. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.128 Protocol Engine Statistics TCP Retransmitted Segments - GLPES_VFTCPRTXSEG[n] (0x0001B600 + 0x4*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
TCPRTXSEG	31:0	0x0	RW1C	**TCP Retransmitted Segments** Counts the number of TCP segments retransmitted by the PE.

38.39.2.17.129 Protocol Engine Statistics UDP Received Packets Low - GLPES_VFUDPRXPKTSLO[n] (0x0001B800 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
UDPRXPKTSLO	31:0	0x0	RW1C	**UDP Received Packets Low** Counts the number of UDP packets received by the PE without errors. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.130 Protocol Engine Statistics UDP Received Packets High - GLPES_VFUDPRXPKTSHI[n] (0x0001B804 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
UDPRXPKTSHI	15:0	0x0	RW1C	**UDP Received Packets High** Counts the number of UDP packets received by the PE without errors. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.131 Protocol Engine Statistics UDP Transmitted Packets Low - GLPES_VFUDPTXPKTSLO[n] (0x0001BA00 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
UDPTXPKTSLO	31:0	0x0	RW1C	**UDP Transmitted Packets Low** Counts the number of UDP packets submitted by the PE to the lower layers for transmission. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.132 Protocol Engine Statistics UDP Transmitted Packets High - GLPES_VFUDPTXPKTSHI[n] (0x0001BA04 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
UDPTXPKTSHI	15:0	0x0	RW1C	**UDP Transmitted Packets High** Counts the number of UDP packets submitted by the PE to the lower layers for transmission. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.133 Protocol Engine Statistics RDMA Received Write Messages Low - GLPES_VFRDMARXWRSLO[n] (0x0001BC00 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
RDMARXWRSLO	31:0	0x0	RW1C	**RDMA Received Writes Low** Counts the number of RDMA write messages received by the PE. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.134 Protocol Engine Statistics RDMA Received Write Messages High - GLPES_VFRDMARXWRSHI[n] (0x0001BC04 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
RDMARXWRSHI	15:0	0x0	RW1C	**RDMA Received Writes High** Counts the number of RDMA write messages received by the PE. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.135 Protocol Engine Statistics RDMA Received Read Request Messages Low - GLPES_VFRDMARXRDSLO[n] (0x0001BE00 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
RDMARXRDSLO	31:0	0x0	RW1C	**RDMA Received Reads Low** Counts the number of RDMA read request messages received by the PE. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.136 Protocol Engine Statistics RDMA Received Read Request Messages High - GLPES_VFRDMARXRDSHI[n] (0x0001BE04 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
RDMARXRDSHI	15:0	0x0	RW1C	**RDMA Received Reads High** Counts the number of RDMA read request messages received by the PE. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.137 Protocol Engine Statistics RDMA Received Send Messages Low - GLPES_VFRDMARXSNDSLO[n] (0x0001C000 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
RDMARXSNDSLO	31:0	0x0	RW1C	**RDMA Received Sends Low** Counts the number of RDMA send messages received by the PE. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.138 Protocol Engine Statistics RDMA Received Send Messages High - GLPES_VFRDMARXSNDSHI[n] (0x0001C004 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
RDMARXSNDSHI	15:0	0x0	RW1C	**RDMA Received Sends High** Counts the number of RDMA send messages received by the PE. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.139 Protocol Engine Statistics RDMA Transmitted Write Messages Low - GLPES_VFRDMATXWRSLO[n] (0x0001C200 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
RDMATXWRSLO	31:0	0x0	RW1C	**RDMA Transmitted Writes Low** Counts the number of RDMA write messages transmitted by the PE. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.140 Protocol Engine Statistics RDMA Transmitted Write Messages High - GLPES_VFRDMATXWRSHI[n] (0x0001C204 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
RDMATXWRSHI	15:0	0x0	RW1C	**RDMA Transmitted Writes High** Counts the number of RDMA write messages transmitted by the PE. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.141 Protocol Engine Statistics RDMA Transmitted Read Request Messages Low - GLPES_VFRDMATXRDSLO[n] (0x0001C400 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
RDMATXRDSLO	31:0	0x0	RW1C	**RDMA Transmitted Reads Low** Counts the number of RDMA read request messages transmitted by the PE. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.142 Protocol Engine Statistics RDMA Transmitted Read Request Messages High - GLPES_VFRDMATXRDSHI[n] (0x0001C404 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
RDMATXRDSHI	15:0	0x0	RW1C	**RDMA Transmitted Reads High** Counts the number of RDMA read request messages transmitted by the PE. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.143 Protocol Engine Statistics RDMA Transmitted Send Messages Low - GLPES_VFRDMATXSNDSLO[n] (0x0001C600 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
RDMATXSNDSLO	31:0	0x0	RW1C	**RDMA Transmitted Sends Low** Counts the number of RDMA send messages transmitted by the PE. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.144 Protocol Engine Statistics RDMA Transmitted Send Messages High - GLPES_VFRDMATXSNDSHI[n] (0x0001C604 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
RDMATXSNDSHI	15:0	0x0	RW1C	**RDMA Transmitted Sends High** Counts the number of RDMA send messages transmitted by the PE. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.145 Protocol Engine Statistics RDMA Verbs Bind Operations Low - GLPES_VFRDMAVBNDLO[n] (0x0001C800 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
RDMAVBNDLO	31:0	0x0	RW1C	**RDMA Verbs Bind Low** Counts the total number of RDMA verb bind operations carried out by the PE. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.146 Protocol Engine Statistics RDMA Verbs Bind Operations High - GLPES_VFRDMAVBNDHI[n] (0x0001C804 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
RDMAVBNDHI	15:0	0x0	RW1C	**RDMA Verbs Bind High** Counts the total number of RDMA verb bind operations carried out by the PE. This is the upper 16 bits of the 48-bit counter.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.17.147 Protocol Engine Statistics RDMA Verbs Invalidate Operations Low - GLPES_VFRDMAVINVLO[n] (0x0001CA00 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
RDMAVINVLO	31:0	0x0	RW1C	**RDMA Verbs Invalidate Low** Counts the total number of RDMA verb invalidate operations carried out by the PE. This is the lower 32 bits of the 48-bit counter.

38.39.2.17.148 Protocol Engine Statistics RDMA Verbs Invalidate Operations High - GLPES_VFRDMAVINVHI[n] (0x0001CA04 + 0x8*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
RDMAVINVHI	31:0	0x0	RW1C	**RDMA Verbs Invalidate High** Counts the total number of RDMA verb invalidate operations carried out by the PE. This is the upper 16 bits of the 48-bit counter.

38.39.2.17.149 Protocol Engine Statistics RDMA Received Unaligned FPDUs - GLPES_RDMARXUNALIGN (0x0001E000; RW1C)

Field	Bit(s)	Init.	Type	Description
RDMRXUNALIGN	31:0	0x0	RW1C	**RDMA Received Unaligned** Counts the number of TCP segments received by the PE that probably carried unaligned FPDUs.

38.39.2.17.150 Protocol Engine Statistics RDMA Received Out of Order No Markers FPDUs - GLPES_RDMARXOOONOMARK (0x0001E004; RW1C)

Field	Bit(s)	Init.	Type	Description
RDMAOOONOMARK	31:0	0x0	RW1C	**RDMA Out of Order No Markers** This register counts RDMA FPDUs received by the PE out of order and not carrying markers.

38.39.2.17.151 Protocol Engine Statistics RDMA Received Multiple FPDUs Low - GLPES_RDMARXMULTFPDUSLO (0x0001E010; RW1C)

Field	Bit(s)	Init.	Type	Description
RDMARXMULTFPDUSLO	31:0	0x0	RW1C	**RDMA Received Multiple FPDUs Low** Counts the number of TCP segments received by the PE that probably have multiple FPDUs. This is the lower 32 bits of the 56-bit counter.

38.39.2.17.152 Protocol Engine Statistics RDMA Received Multiple FPDUs High - GLPES_RDMARXMULTFPDUSHI (0x0001E014; RW1C)

Field	Bit(s)	Init.	Type	Description
RDMARXMULTFPDUSHI	23:0	0x0	RW1C	**RDMA Received Multiple FPDUs High** Counts the number of TCP segments received by the PE that probably have multiple FPDUs. This is the upper 24 bits of the 56-bit counter.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.17.153 Protocol Engine Statistics RDMA Out of Order Placed DDP Segments Low - GLPES_RDMARXOOODDPLO (0x0001E018; RW1C)

Field	Bit(s)	Init.	Type	Description
RDMARXOOODDPLO	31:0	0x0	RW1C	**RDMA Received Out of Order DDP Low** Counts the number of the DDP segments received by the PE and out order placed. This is the lower 32 bits of the 56-bit counter.

38.39.2.17.154 Protocol Engine Statistics RDMA Out of Order Placed DDP Segments High - GLPES_RDMARXOOODDPHI (0x0001E01C; RW1C)

Field	Bit(s)	Init.	Type	Description
RDMARXOOODDPHI	23:0	0x0	RW1C	**RDMA Received Out of Order DDP High** Counts the number of the DDP segments received by the PE and out order placed. This is the upper 24 bits of the 56-bit counter.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.17.155 Protocol Engine Statistics TCP Received Pure Acks Low - GLPES_TCPRXPUREACKSLO (0x0001E020; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPRXPUREACKLO	31:0	0x0	RW1C	**TCP Received Pure ACKs Low** Counts the number of TCP ACKs received by the PE carrying no data. This is the lower 32 bits of the 56-bit counter.

38.39.2.17.156 Protocol Engine Statistics TCP Received Pure Acks High - GLPES_TCPRXPUREACKHI (0x0001E024; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPRXPUREACKSHI	23:0	0x0	RW1C	**TCP Received Pure ACKs High** Counts the number of TCP ACKs received by the PE carrying no data. This is the upper 24 bits of the 56-bit counter.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.17.157 Protocol Engine Statistics TCP Receive First Hole Low - GLPES_TCPRXONEHOLELO (0x0001E028; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPRXONEHOLELO	31:0	0x0	RW1C	**TCP Receive One (First) Hole Low** Counts the number of TCP segments received by the PE and opened a first TCP hole in TCP sequence space. This is the lower 32 bits of the 56-bit counter.

38.39.2.17.158 Protocol Engine Statistics TCP Received First Hole High - GLPES_TCPRXONEHOLEHI (0x0001E02C; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPRXONEHOLEHI	23:0	0x0	RW1C	**TCP Receive One (First) Hole High** Counts the number of TCP segments received by the PE and opened a first TCP hole in TCP sequence space. This is the upper 24 bits of the 56-bit counter.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.17.159 Protocol Engine Statistics TCP Receive Second Hole Low - GLPES_TCPRXTWOHOLELO (0x0001E030; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPRXTWOHOLELO	31:0	0x0	RW1C	**TCP Receive Two (Second) Hole Low** Counts the number of TCP segments received by the PE and opened a second TCP hole in TCP sequence space. This is the lower 32 bits of the 56-bit counter.

38.39.2.17.160 Protocol Engine Statistics TCP Received Second Hole High - GLPES_TCPRXTWOHOLEHI (0x0001E034; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPRXTWOHOLEHI	23:0	0x0	RW1C	**TCP Receive Two (Second) Hole High** Counts the number of TCP segments received by the PE and opened a second TCP hole in TCP sequence space. This is the upper 24 bits of the 56-bit counter.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.17.161 Protocol Engine Statistics TCP Receive Third Hole Low - GLPES_TCPRXTHREEHOLELO (0x0001E038; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPRXTHREEHOLELO	31:0	0x0	RW1C	**TCP Receive Three (Third) Hole Low** Counts the number of TCP segments received by the PE and opened a third TCP hole in TCP sequence space. This is the lower 32 bits of the 56-bit counter.

38.39.2.17.162 Protocol Engine Statistics TCP Received Third Hole High - GLPES_TCPRXTHREEHOLEHI (0x0001E03C; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPRXTHREEHOLEHI	23:0	0x0	RW1C	**TCP Receive Three (third) Hole High** Counts the number of TCP segments received by the PE and opened a third TCP hole in TCP sequence space. This is the upper 24 bits of the 56-bit counter.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.17.163 Protocol Engine Statistics TCP Receive Fourth Hole Low - GLPES_TCPRXFOURHOLELO (0x0001E040; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPRXFOURHOLELO	31:0	0x0	RW1C	**TCP Receive Four (Fourth) Hole Low** Counts the number of TCP segments received by the PE and opened a fourth TCP hole in TCP sequence space. This is the lower 32 bits of the 56-bit counter.

38.39.2.17.164 Protocol Engine Statistics TCP Receive Fourth Hole High - GLPES_TCPRXFOURHOLEHI (0x0001E044; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPRXFOURHOLEHI	23:0	0x0	RW1C	**TCP Receive Four (Fourth) Hole High** Counts the number of TCP segments received by the PE and opened a fourth TCP hole in TCP sequence space. This is the upper 24 bits of the 56-bit counter.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.17.165 Protocol Engine Statistics TCP Fast Re-transmissions Low - GLPES_TCPTXRETRANSFASTLO (0x0001E048; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPTXRETRANSFASTLO	31:0	0x0	RW1C	**TCP Re-transmissions Fast Low** Counts the number of TCP fast re-transmissions by the PE. This is the lower 32 bits of the 56-bit counter.

38.39.2.17.166 Protocol Engine Statistics TCP Fast Re-transmissions High - GLPES_TCPTXRETRANSFASTHI (0x0001E04C; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPTXRETRANSFASTHI	23:0	0x0	RW1C	**TCP Re-transmissions Fast High** Counts the number of TCP fast re-transmissions by the PE. This is the upper 24 bits of the 56-bit counter.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.17.167 Protocol Engine Statistics TCP Fast Re-transmissions Timeouts Low - GLPES_TCPTXTOUTSFASTLO (0x0001E050; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPTXTOUTSFASTLO	31:0	0x0	RW1C	**TCP Re-transmission Timeouts Fast Low** Counts the number of TCP re-transmission timeouts by the PE on the connections attempting fast re-transmit. This is the lower 32 bits of the 56-bit counter.

38.39.2.17.168 Protocol Engine Statistics TCP Fast Re-Transmissions Timeouts High - GLPES_TCPTXTOUTSFASTHI (0x0001E054; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPTXTOUTSFASTHI	23:0	0x0	RW1C	**TCP Re-transmission Timeouts Fast High** Counts the number of TCP re-transmission timeouts by the PE on the connections attempting fast re-transmit. This is the upper 24 bits of the 56-bit counter.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.17.169 Protocol Engine Statistics TCP Re-Transmissions Timeouts Low - GLPES_TCPTXTOUTSLO (0x0001E058; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPTXTOUTSLO	31:0	0x0	RW1C	**TCP Re-transmission Timeouts Low** Counts the number of TCP re-transmission timeouts by the PE on the connections that are not currently attempting fast re-transmit. This is the lower 32 bits of the 56-bit counter.

38.39.2.17.170 Protocol Engine Statistics TCP Re-Transmissions Timeouts High - GLPES_TCPTXTOUTSHI (0x0001E05C; RW1C)

Field	Bit(s)	Init.	Type	Description
TCPTXTOUTSHI	23:0	0x0	RW1C	**TCP Re-transmission Timeouts High** Counts the number of TCP re-transmission timeouts by the PE on the connections that are not currently attempting fast re-transmit. This is the upper 24 bits of the 56-bit counter.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.18 PF - LAN Transmit Receive Registers

38.39.2.18.1 Global TSO TCP Mask First - GLLAN_TSOMSK_F (0x000442D8; RW)

Field	Bit(s)	Init.	Type	Description
TCPMSKF	11:0	0x0	RW	**TCP Mask First** TCP flags mask for the first segment in the TSO. Any bit set to one in the *TCPMSK* field clears the respective TCP flag in the TSO. Bit 0 relates to *FIN* flag, bit 1 relates to the *SYN* flag and so on.
RESERVED	31:12	0x0	RSV	Reserved.

38.39.2.18.2 Global TSO TCP Mask Middle - GLLAN_TSOMSK_M (0x000442DC; RW)

Field	Bit(s)	Init.	Type	Description
TCPMSKM	11:0	0x0	RW	**TCP Mask Middle** TCP flags mask for the middle segments in the TSO. See *TCPMSKF* in Section 38.39.2.18.1 for the impact of each bit in the field.
RESERVED	31:12	0x0	RSV	Reserved.

38.39.2.18.3 Global TSO TCP Mask Last - GLLAN_TSOMSK_L (0x000442E0; RW)

Field	Bit(s)	Init.	Type	Description
TCPMSKL	11:0	0x0	RW	**TCP Mask Last** TCP flags mask for the middle segments in the TSO. See *TCPMSKF* in Section 38.39.2.18.1 for the impact of each bit in the field.
RESERVED	31:12	0x0	RSV	Reserved.

38.39.2.18.4 Receive Processing Block Control - GL_RDPU_CNTRL (0x00051060; RW)

Field	Bit(s)	Init.	Type	Description
RX_PAD_EN	0	1b	RW	**Receive Pad Enable** Pad Rx packets with zeros that do not include CRC or the CRC is stripped to be at least 60 bytes long. ***Note:*** Padding and CRC stripping for EMP packets is always done regardless of the setting of this flag.
ECO	31:1	0xFFFF	RW	**ECO** Reserved for future ECO.

38.39.2.18.5 VF PF Queue Mapping Table - VPLAN_QTABLE[n,VF] (0x00070000 + 0x400*n + 0x4*VF, n=0...15, VF=0...127; RW)

This register affects the VF, but is exposed only to the parent PF.

Field	Bit(s)	Init.	Type	Description
QINDEX	10:0	0x7FF	RW	**Queue Index** Defines the index of VF queue 'n' in the PF queues space, where 'n' is the register index. Setting the *QINDEX* to 0x7FF means that the queue is not valid for the VF. Relevant only if VPLAN_QBASE.*VFQTABLE_ENA* is set to 1b (see Section 38.39.2.18.7).
RESERVED	31:11	0x0	RSV	Reserved.

38.39.2.18.6 VF LAN Enablement - VPLAN_MAPENA[VF] (0x00074000 + 0x4*VF, VF=0...127; RW)

Field	Bit(s)	Init.	Type	Description
TXRX_ENA	0	0b	RW	**Transmit/Receive Enable** The VFLAN_QTABLE for the VF is enabled only when the *TXRX_ENA* flag is set.
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.18.7 VF PF Queue Range - VPLAN_QBASE[VF] (0x00074800 + 0x4*VF, VF=0...127; RW)

Field	Bit(s)	Init.	Type	Description
VFFIRSTQ	10:0	0x0	RW	**VF First Queue** Defines the base index of VF 'n' within the range of the PF queues, where 'n' is the VF register index. The *VFFIRSTQ* is meaningful for this VSI only if the *VFQTABLE_ENA* is cleared.
VFNUMQ	18:11	0x0	RW	**VF Number of Queues** Defines the number of queues allocated to VF 'n' within the range of the PF queues, where 'n' is the VF register index. The value should be set to the number of queues -1 (a value of 0 means 1 queue and a value of 255 means 256 queues). The *VFNUMQ* is meaningful for this VSI only if the *VFQTABLE_ENA* is cleared.
RESERVED	30:19	0x0	RSV	Reserved.
VFQTABLE_ENA	31	1b	RW	**VF Queue Table Enable** Selects between contiguous range of queues for this VSI versus a scattered range. 0b = The VF is assigned a contiguous range starting at VFBASE 1b = The VF is assigned a scattered range defined by the VFLAN_QTABLE

38.39.2.18.8 Global Transmit Queue Head - QTX_HEAD[Q] (0x000E4000 + 0x4*Q, Q=0...1535; RW)

Field	Bit(s)	Init.	Type	Description
RESERVED	31:0	0x0	RSV	Reserved.

38.39.2.18.9 Global Transmit Pre Queue Disable - GLLAN_TXPRE_QDIS[n] (0x000E6500 + 0x4*n, n=0...11; RW)

Field	Bit(s)	Init.	Type	Description
QINDX	10:0	0x0	RW	**Queue Index** Absolute index in the device space. Each register 'n' covers 128*n-128*n-1 queues. Software is expected to use the matched register when accessing a specific queue.
RESERVED	29:11	0x0	RSV	Reserved.
SET_QDIS	30	0b	RW	**Set Queue Disable** Setting this flag to 1b sets an internal *QDIS* flag of the transmit queue (that is indicated by *QINDX* field in this register). As a result, any accumulated quanta of the queue are invalidated, which is a needed step before the queue is disabled. Setting the *SET_QDIS* flag is mutually exclusive with the *CLEAR_QDIS* flag.
CLEAR_QDIS	31	0b	RW	**Clear Queue Disable** Setting this flag to 1b clears an internal QDIS flag of the transmit queue (that is indicated by *QINDX* field in this register). This step should be made before the queue is enabled. Setting the *CLEAR_QDIS* flag is mutually exclusive with the *SET_QDIS* flag.

38.39.2.18.10 Global Transmit Queue Enable - QTX_ENA[Q] (0x00100000 + 0x4*Q, Q=0...1535; RW)

Field	Bit(s)	Init.	Type	Description
QENA_REQ	0	0b	RW	**Transmit Queue Enable Request** If this bit is set, the software should poll the *QENA_STAT* flag (in this register) before using the queue. After clearing this flag, the software should poll the *QENA_STAT* flag before releasing the memory structures. Once software changes the state of the *QENA_REQ* flag, it must poll the *QENA_STAT* before it is permitted to revert the state of the *QENA_REQ* once again.
FAST_QDIS	1	0b	RW	**Fast Queue Disable** See the Fast Transmit Queue Disable Flow section for the usage of this flag.
QENA_STAT	2	0b	RO	**Transmit Queue Enable Status Indication** 0b = Indicates that the queue is inactive. 1b = Indicates that the queue is active.
RESERVED	31:3	0x0	RSV	Reserved.

38.39.2.18.11 Global Transmit Queue Control - QTX_CTL[Q] (0x00104000 + 0x4*Q, Q=0...1535; RW)

Field	Bit(s)	Init.	Type	Description
PFVF_Q	1:0	00b	RW	**PF/VF Queue** Queue association to PF or VF as follows: 00b = VF queue. 01b = VM queue. 10b = PF queue. 11b = Reserved.
PF_INDX	5:2	0x0	RW	**PF Index** Index between 0 and 15.
RESERVED	6	0b	RSV	Reserved.
VFVM_INDX	15:7	0x0	RW	**VF/VM Index** Should be programmed per PFVF_Q setting as follows: **PFVF_Q Setting** — **Matched VF/VM Index Programming** 00b = VF queue — The absolute VF index (between 0-127). 01b = VM queue — The absolute VSI index (between 0-383). Else = PF or EMP queue — Must be set to zero.
RESERVED	31:16	0x0	RSV	Reserved

38.39.2.18.12 Global Transmit Queue Tail - QTX_TAIL[Q] (0x00108000 + 0x4*Q, Q=0...1535; RW)

Field	Bit(s)	Init.	Type	Description
TAIL	12:0	0x0	RW	**Transmit Tail** Defines the first descriptor that the software prepares for the hardware (it is the last valid descriptor plus one). The tail is a relative descriptor index to the beginning of the transmit descriptor ring.
RESERVED	31:13	0x0	RSV	Reserved.

38.39.2.18.13Global Receive Queue Enable - QRX_ENA[Q] (0x00120000 + 0x4*Q, Q=0...1535; RW)

Field	Bit(s)	Init.	Type	Description
QENA_REQ	0	0b	RW	**Receive Queue Enable Request** If this bit is set, the software should poll the *QENA_STAT* flag (in this register) before using the queue. After clearing this flag, the software should poll the *QENA_STAT* flag before releasing the memory structures. Once software changes the state of the *QENA_REQ* flag it must poll the *QENA_STAT* before it is permitted to revert the state of the *QENA_REQ* once again.
FAST_QDIS	1	0b	RW1C	**Fast Queue Disable** See the Fast Receive Queue Disable Flow section for the usage of this flag. This flag is auto-cleared by hardware.
QENA_STAT	2	0b	RO	**Receive Queue Enable Status Indication** 0b = Indicates that the queue is inactive. 1b = Indicates that the queue is active.
RESERVED	31:3	0x0	RSV	Reserved.

38.39.2.18.14Global Receive Queue Tail - QRX_TAIL[Q] (0x00128000 + 0x4*Q, Q=0...1535; RW)

Field	Bit(s)	Init.	Type	Description
TAIL	12:0	0x0	RW	**Receive Tail** Defines the first descriptor that software hands to hardware (it is the last valid descriptor plus one). The tail is a relative descriptor index to the beginning of the receive descriptor ring.
RESERVED	31:13	0x0	RSV	Reserved.

38.39.2.18.15Global RLAN Control 0 - GLLAN_RCTL_0 (0x0012A500; RW1C)

Field	Bit(s)	Init.	Type	Description
PXE_MODE	0	1b	RW1C	**PXE Mode** When this flag is set, the device fetches and writes back a single descriptor at a time. During normal performance operation, (non-PXE mode) this flag must be cleared.
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.18.16PF Queue Allocation - PFLAN_QALLOC (0x001C0400; RO)

These registers define the LAN queue pairs allocation to the PFs.

Field	Bit(s)	Init.	Type	Description
FIRSTQ	10:0	0x0	RW	**First Queue** The first LAN queue pair allocated to this PF. Valid only if the *VALID* flag is set. Valid values are 0-1535.
RESERVED	15:11	0x0	RSV	Reserved.
LASTQ	26:16	0x0	RW	**Last Queue** The last LAN queue pair allocated to this PF. Valid only if the *VALID* flag is set. Valid values are 0-1535.
RESERVED	30:27	0x0	RSV	Reserved.
VALID	31	0b	RW	**Valid** Indicates that queues are allocated to this PF. For any active PF this flag must be set.

38.39.2.18.17 VSI Receive Queue Mapping Table - VSILAN_QTABLE[n,VSI] (0x00200000 + 0x800*n + 0x4*VSI, n=0...7, VSI=0...383; RO)

Field	Bit(s)	Init.	Type	Description
QINDEX_0	10:0	0x0	RW	**Queue Index 0** Defines the index of the VSI queue '2*n' in the PF queues space, where 'n' is the register index. The absolute queue index in the device space equals *QINDEX* plus PFLAN_QALLOC.*FIRSTQ* of the parent PF. Setting the *QINDEX* to 0x7FF means the queue is not valid.
RESERVED	15:11	0x0	RSV	Reserved.
QINDEX_1	26:16	0x0	RW	**Queue Index 1** Defines the index of the VSI queue '2*n+1' in the PF queues space, where 'n' is the register index. The absolute queue index in the device space equals to *QINDEX* plus PFLAN_QALLOC.*FIRSTQ* of the parent PF. Setting the *QINDEX* to 0x7FF means the queue is not valid.
RESERVED	31:27	0x0	RSV	Reserved.

38.39.2.18.18 VSI Queue Control - VSILAN_QBASE[VSI] (0x0020C800 + 0x4*VSI, VSI=0...383; RO)

Field	Bit(s)	Init.	Type	Description
VSIBASE	10:0	0x0	RW	**VSI Base** Defines the base index of VSI 'n' within the range of the PF queues, where 'n' is the VSI register index. The *VSIBASE* is meaningful for this VSI only if the *VSIQTABLE_ENA* is cleared.
VSIQTABLE_ENA	11	0b	RW	**VSI Queue Table Enable** Selects between contiguous range of queues for this VSI vs. scattered range: 0b = The VSI is assigned a contiguous range starting at *VSIBASE*. 1b = The VSI is assigned a scattered range defined by the *VSILAN_QTABLE*.
RESERVED	31:12	0x0	RSV	Reserved.

38.39.2.19 PF - Rx Filters Registers

38.39.2.19.1 VF Queue Filter Control - VPQF_CTL[VF] (0x001C0000 + 0x4*VF, VF=0...127; RO)

This register affects the VF, but is exposed only to the parent PF.

Field	Bit(s)	Init.	Type	Description
PEHSIZE	4:0	0x0	RW	**PE Hash Size** Defines the number of buckets of the PE quad hash filter table for the VF. It is defined in power of 2 equal to 1K x 2 ** *PEHSIZE*. *PEHSIZE* can have any value between 0 and 10. *PEHSIZE* = 0, 1,... 10 is equivalent to 1 K, 2 K, 4 K... 1 M buckets.
PEDSIZE	9:5	0x0	RW	**PEDSIZE** Defines the number of PE quad hash contexts for the VF. It is defined in power of 2 equal to 0.5K x 2 ** *PEDSIZE*. *PEDSIZE* can have any value between 0 and 9. *PEHSIZE* = 0, 1,... 9 is equivalent to 0.5 K, 1 K, 2 K, 4 K... 256 K contexts.
RESERVED	31:10	0x0	RSV	Reserved.

38.39.2.19.2 PF Queue Filter Control 0 - PFQF_CTL_0 (0x001C0AC0; RW)

Settings in this register allow the hardware to auto-clear internal table pointers, as well as the internal PE Quad Hash context counter and table pointers of the function.

Field	Bit(s)	Init.	Type	Description
RESERVED	9:0	0x0	RSV	Reserved.
PFFCHSIZE	15:10	0x0	RW	Reserved.
HASHLUTSIZE	16	0b	RW	**Hash LUT Size** Defines the size of the Hash LUT as follows: 128 512
FD_ENA	17	0b	RW	**Flow Director Enable** Enable flow director filters for the PF and its VFs.
ETYPE_ENA	18	0b	RW	**EtherType Enable** Enable EtherType queue filters for the PF and its VFs.
MACVLAN_ENA	19	0b	RW	**MAC/LAN Enable** Enable MAC/VLAN queue filters for the PF and its VFs.
RESERVED	31:20	0x0	RSV	Reserved.

38.39.2.19.3 VSI Receive Traffic Class Queues - VSIQF_TCREGION[n,VSI] (0x00206000 + 0x800*n + 0x4*VSI, n=0...3, VSI=0...383; RO)

Field	Bit(s)	Init.	Type	Description
TC_OFFSET	8:0	0x0	RW	**Traffic Class Offset** Indicates the relative queue index to the beginning of the VSI that the TC '2*n' uses, where 'n' is the register index. Hardware does not check if the queue index exceeds the VSI range. It is the responsibility of PF software to ensure that *TC_SIZE* + *TC_OFFSET* does not exceed the VSI range.
TC_SIZE	11:9	000b	RW	**Traffic Class Size** Indicates the number of receive queues allocated to TC '2*n' for the VSI, where 'n' is the register index. The *TC_SIZE* field can have a value of 0 to 7, which means any of the following respective number of allocated queues: 000b = 1. 001b = 2. 010b = 4. 011b = 8. 100b = 16. 101b = 32. 110b = 64. 111b = 128.
RESERVED	15:12	0x0	RSV	Reserved.
TC_OFFSET2	24:16	0x0	RW	**Traffic Class Offset 2** Indicates the relative queue index to the beginning of the VSI that the TC '2*n+1' uses, where 'n' is the register index. Hardware does not check if the queue index exceeds the VSI range. It is the responsibility of PF software to ensure that *TC_SIZE2* + *TC_OFFSET2* does not exceeds the VSI range.

Field	Bit(s)	Init.	Type	Description
TC_SIZE2	27:25	000b	RW	**Traffic Class Size 2** Indicates the number of receive queues allocated to TC '2*n+1' for the VSI, where 'n' is the register index. The TC_SIZE2 field can have any value of 0 to 7, which means any of the following respective number of allocated queues: 000b = 1. 001b = 2. 010b = 4. 011b = 8. 100b = 16. 101b = 32. 110b = 64. 111b = 128.
RESERVED	31:28	0x0	RSV	Reserved.

38.39.2.19.4 VSI Queue Filter Control - VSIQF_CTL[VSI] (0x0020D800 + 0x4*VSI, VSI=0...383; RO)

Field	Bit(s)	Init.	Type	Description
RESERVED	0	0b	RW	Reserved.
PETCP_ENA	1	0b	RW	**TCP Enable** Enable TCP packets by the quad hash filter for the VSI.
RESERVED	7:2	0x0	RSV	Reserved.
RSS_LUT_TYPE	8	0b	RW	**RSS LUT Type** 0b = Use PF LUT (PFQF_LUT). 1b = Use VSI LUT (VSIQF_LUT).
RESERVED	31:9	0x0	RSV	Reserved.

38.39.2.19.5 VSI Queue Filter Hash LUT - VSIQF_HLUT[n,VSI] (0x00220000 + 0x800*n + 0x4*VSI, n=0...15, VSI=0...383; RW)

Field	Bit(s)	Init.	Type	Description
LUT0	3:0	0x0	RW	**LUT 0** Hash redirection LUT entry '4*n', where 'n' is the register index.
RESERVED	7:4	0x0	RSV	Reserved.
LUT1	11:8	0x0	RW	**LUT 1** Hash redirection LUT entry '4*n+1', where 'n' is the register index.
RESERVED	15:12	0x0	RSV	Reserved.
LUT2	19:16	0x0	RW	**LUT 2** Hash redirection LUT entry '4*n+2', where 'n' is the register index.
RESERVED	23:20	0x0	RSV	Reserved.
LUT3	27:24	0x0	RW	**LUT 3** Hash redirection LUT entry '4*n+3', where 'n' is the register index.
RESERVED	31:28	0x0	RSV	Reserved.

38.39.2.19.6 VF Queue Filter Hash Region of Queues - VFQF_HREGION[n,VF] (0x0022E000 + 0x400*n + 0x4*VF, n=0...7, VF=0...127; RW)

Field	Bit(s)	Init.	Type	Description
OVERRIDE_ENA_0	0	0b	RW	**Override Enable 0** Override TC region for packet type '8*n', where 'n' is the register index.
REGION_0	3:1	000b	RW	**Region 0** Receive queue region for packet type '8*n', where 'n' is the register index. This field is meaningful only if the *OVERRIDE_ENA_1* flag is set.
OVERRIDE_ENA_1	4	0b	RW	**Override Enable 1** Override TC region for packet type '8*n+1', where 'n' is the register index.
REGION_1	7:5	000b	RW	**Region 1** Receive queue region for packet type '8*n+1', where 'n' is the register index. This field is meaningful only if the *OVERRIDE_ENA_1* flag is set.
OVERRIDE_ENA_2	8	0b	RW	**Override Enable 2** Override TC region for packet type '8*n+2', where 'n' is the register index.
REGION_2	11:9	000b	RW	**Region 2** Receive queue region for packet type '8*n+2', where 'n' is the register index. This field is meaningful only if the *OVERRIDE_ENA_1* flag is set.
OVERRIDE_ENA_3	12	0b	RW	**Override Enable 3** Override TC region for packet type '8*n+3', where 'n' is the register index.
REGION_3	15:13	000b	RW	**Region 3** Receive queue region for packet type '8*n+3', where 'n' is the register index. This field is meaningful only if the *OVERRIDE_ENA_1* flag is set.
OVERRIDE_ENA_4	16	0b	RW	**Override Enable 4** Override TC region for packet type '8 x n + 4', where 'n' is the register index.
REGION_4	19:17	000b	RW	**Region 4** Receive queue region for packet type '8*n+4', where 'n' is the register index. This field is meaningful only if the *OVERRIDE_ENA_1* flag is set.
OVERRIDE_ENA_5	20	0b	RW	**Override Enable 5** Override TC region for packet type '8*n+5', where 'n' is the register index.
REGION_5	23:21	000b	RW	**Region 5** Receive queue region for packet type '8*n+5', where 'n' is the register index. This field is meaningful only if the *OVERRIDE_ENA_1* flag is set.
OVERRIDE_ENA_6	24	0b	RW	**Override Enable 6** Override TC region for packet type '8*n+6', where 'n' is the register index.
REGION_6	27:25	000b	RW	**Region 6** Receive queue region for packet type '8*n+6', where 'n' is the register index. This field is meaningful only if the *OVERRIDE_ENA_1* flag is set.
OVERRIDE_ENA_7	28	0b	RW	**Override Enable 7** Override TC region for packet type '8*n+7', where 'n' is the register index.
REGION_7	31:29	000b	RW	**Region 7** Receive queue region for packet type '8*n+7', where 'n' is the register index. This field is meaningful only if the *OVERRIDE_ENA_1* flag is set.

38.39.2.19.7 VF Queue Filter Hash Enabled Packet Type - VFQF_HENA[n,VF] (0x00230800 + 0x400*n + 0x4*VF, n=0...1, VF=0...127; RW)

Field	Bit(s)	Init.	Type	Description
PTYPE_ENA	31:0	0x0	RW	**Packet Type Enable** Packet type enablement of the hash filter for the function. Bit 'm' in register 'n' enables packet type '32*n+m' as defined in the Packet Types for the Classification Filters table.

38.39.2.19.8 PF Queue Filter Hash LUT - PFQF_HLUT[n] (0x00240000 + 0x80*n, n=0...127; RW)

Field	Bit(s)	Init.	Type	Description
LUT0	6:0	0x0	RW	**LUT 0** Hash redirection LUT entry '4*n', where 'n' is the register index.
RESERVED	7	0b	RSV	Reserved.
LUT1	14:8	0x0	RW	**LUT 1** Hash redirection LUT entry '4*n+1', where 'n' is the register index.
RESERVED	15	0b	RSV	Reserved.
LUT2	22:16	0x0	RW	**LUT 2** Hash redirection LUT entry '4*n+2', where 'n' is the register index.
RESERVED	23	0b	RSV	Reserved.
LUT3	30:24	0x0	RW	**LUT 3** Hash redirection LUT entry '4*n+3', where 'n' is the register index.
RESERVED	31	0b	RSV	Reserved.

38.39.2.19.9 PF Queue Filter Hash Region of Queues - PFQF_HREGION[n] (0x00245400 + 0x80*n, n=0...7; RW)

Field	Bit(s)	Init.	Type	Description
OVERRIDE_ENA_0	0	0b	RW	**Override Enable 0** Override TC region for packet type '8* n', where 'n' is the register index.
REGION_0	3:1	000b	RW	**Region 0** Receive queue region for packet type '8*n', where 'n' is the register index. This field is meaningful only if the OVERRIDE_ENA_1 flag is set.
OVERRIDE_ENA_1	4	0b	RW	**Override Enable 1** Override TC region for packet type '8*n+1', where 'n' is the register index.
REGION_1	7:5	000b	RW	**Region 1** Receive queue region for packet type '8*n+1', where 'n' is the register index. This field is meaningful only if the OVERRIDE_ENA_1 flag is set.
OVERRIDE_ENA_2	8	0b	RW	**Override Enable 2** Override TC region for packet type '8*n+2', where 'n' is the register index.
REGION_2	11:9	000b	RW	**Region 2** Receive queue region for packet type 8*n+2', where 'n' is the register index. This field is meaningful only if the OVERRIDE_ENA_1 flag is set.
OVERRIDE_ENA_3	12	0b	RW	**Override Enable 3** Override TC region for packet type '8*n+3', where 'n' is the register index.
REGION_3	15:13	000b	RW	**Region 3** Receive queue region for packet type '8*n+3', where 'n' is the register index. This field is meaningful only if the OVERRIDE_ENA_1 flag is set.
OVERRIDE_ENA_4	16	0b	RW	**Override Enable 4** Override TC region for packet type '8*n+4', where 'n' is the register index.
REGION_4	19:17	000b	RW	**Region 4** Receive queue region for packet type '8*n+4', where 'n' is the register index. This field is meaningful only if the OVERRIDE_ENA_1 flag is set.
OVERRIDE_ENA_5	20	0b	RW	**Override Enable 5** Override TC region for packet type '8*n+5', where 'n' is the register index.

Field	Bit(s)	Init.	Type	Description
REGION_5	23:21	000b	RW	**Region 5** Receive queue region for packet type '8*n+5', where 'n' is the register index. This field is meaningful only if the OVERRIDE_ENA_1 flag is set.
OVERRIDE_ENA_6	24	0b	RW	**Override Enable 6** Override TC region for packet type '8*n+6', where 'n' is the register index.
REGION_6	27:25	000b	RW	**Region 6** Receive queue region for packet type '8*n+6', where 'n' is the register index. This field is meaningful only if the OVERRIDE_ENA_1 flag is set.
OVERRIDE_ENA_7	28	0b	RW	**Override Enable 7** Override TC region for packet type '8*n+7', where 'n' is the register index.
REGION_7	31:29	000b	RW	**Region 7** Receive queue region for packet type '8*n+7', where 'n' is the register index. This field is meaningful only if the OVERRIDE_ENA_1 flag is set.

38.39.2.19.10PF Queue Filter Hash Enabled Packet Type - PFQF_HENA[n] (0x00245900 + 0x80*n, n=0...1; RW)

Field	Bit(s)	Init.	Type	Description
PTYPE_ENA	31:0	0x0	RW	**Packet Type Enable** Packet type enablement of the hash filter for the function. Bit 'm' in register 'n' enables packet type '32*n+m' as defined in the Packet Types for the Classification Filters table.

38.39.2.19.11PF Queue Filter Control 1 - PFQF_CTL_1 (0x00245D80; RW)

Field	Bit(s)	Init.	Type	Description
CLEARFDTABLE	0	0b	RW1C	**Clear Flow Director Table** If software sets this flag, hardware invalidates all entries of the PF in the FD table. Once all entries of the PF are invalidated, this flag is cleared as well.
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.19.12PF Queue Filter Flow Director Allocation - PFQF_FDALLOC (0x00246280; RW)

Field	Bit(s)	Init.	Type	Description
FDALLOC	7:0	0x0	RW	**Flow Director Allocation** Defines the number of guaranteed entries in the FD table. It is defined in granularity of 32 entries.
FDBEST	15:8	0x0	RW	**Flow Director Best Effort** Defines the maximum number of entries the PF can consume from the shared space. It is defined in granularity of 32 entries.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.19.13PF Queue Filter Flow Director Allocation Status - PFQF_FDSTAT (0x00246380; RO)

Field	Bit(s)	Init.	Type	Description
GUARANT_CNT	12:0	0x0	RO	**Guaranteed Count** Indicates the number of FD entries consumed by the PF out of its guaranteed space.
RESERVED	15:13	000b	RSV	Reserved.
BEST_CNT	28:16	0x0	RO	**Best Effort Count** Indicates the number of FD entries consumed by the PF out of its best effort space.
RESERVED	31:29	000b	RSV	Reserved.

38.39.2.19.14Port Queue Filter Flow Director Input Set - PRTQF_FD_FLXINSET[n] (0x00253800 + 0x20*n, n=0...63; RW)

Field	Bit(s)	Init.	Type	Description
INSET	7:0	0x0	RW	**Inset** Bit 'i' of *INSET* enables word 7 minus 'i' of the flexible payload in the field vector.
RESERVED	31:8	0x0	RSV	Reserved.

38.39.2.19.15Port Queue Filter - Flexible Parser Information Table - PRTQF_FLX_PIT[n] (0x00255200 + 0x20*n, n=0...8; RW)

These registers define the byte stream extracted to the field vector (unique setting option per LAN port).

Field	Bit(s)	Init.	Type	Description
SOURCE_OFF	4:0	0x0	RW	**Source Offset** Source word offset in the mapped protocol layer header starting from its beginning. Setting Source Offset rules: • *SOURCE_OFF* + *FSIZE* should not exceed byte 480 of the packet. • Must be programmed in ascending order: Current Offset >= previous offset + previous *FSIZE*. • The previous rule applies for all entries, including non-used ones.
FSIZE	9:5	0x0	RW	**Field Size** Defined in word units. A zero value means that this register has no impact. For non used registers, this field should be set to 1.
DEST_OFF	15:10	0x0	RW	**Destination Offset** Destination word offset in the Field Vector. The destination offset can be set to 50...57 matching offset 0...7 in the flexible field vector, respectively, where *DEST_OFF* + *FSIZE* must not be greater than 58. For non-used registers the *DEST_OFF* must be set to 0x63 (outside of active entries).
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.19.16Port Queue Filter Control 0 - PRTQF_CTL_0 (0x00256E60; RW)

Field	Bit(s)	Init.	Type	Description
HSYM_ENA	0	0b	RW	**Symmetric Hash Enable** Enable symmetric hash for this physical port.
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.19.17Global Queue Filter Accelerated Port Bit Vector - GLQF_APBVT[n] (0x00260000 + 0x4*n, n=0...2047; RO)

Field	Bit(s)	Init.	Type	Description
APBVT	31:0	0x0	RW	**Accelerated Port Bit Vector** Each bit 'i' in the APBVT or register 'n' enables port number 32 x 'n' + 'i'.

38.39.2.19.18Global Queue Filter Packet Counter - GLQF_PCNT[n] (0x00266800 + 0x4*n, n=0...511; RW1C)

Field	Bit(s)	Init.	Type	Description
PCNT	31:0	0x0	RW1C	**Packet Counter** Packet counter indicated by the FD filter(s). The counter wraps around after 0xFF...F.

38.39.2.19.19Global Queue Filter SWAP Fields - GLQF_SWAP[n,m] (0x00267E00 + 0x4*n + 0x8*m, n=0...1, m=0...63; RW)

Field	Bit(s)	Init.	Type	Description
OFF0_SRC0	5:0	0x0	RW	**Offset 0/Source 0** Offset of the first field of the first couple in the field vector to be swapped. The offset is defined in word units.
OFF0_SRC1	11:6	0x0	RW	**Offset 0/Source 1** Offset of the second field of the first couple in the field vector to be swapped. The offset is defined in word units.
FLEN0	15:12	0x0	RW	**Field Length 0** (in word units) When the *FLEN0* is set to zero, the fields defined by *OFF0* are not candidates for swapping.
OFF1_SRC0	21:16	0x0	RW	**Offset 1/Source 0** Offset of the first field of the second couple in the field vector to be swapped. The offset is defined in word units.
OFF1_SRC1	27:22	0x0	RW	**Offset 1/Source 1** Offset of the second field of the second couple in the field vector to be swapped. The offset is defined in word units.
FLEN1	31:28	0x0	RW	**Field Length 1** (in word units) When the *FLEN1* is set to zero, the fields defined by *OFF1* are not candidates for swapping.

38.39.2.19.20 Flow Director PCTYPE Translation Table - GLQF_FD_PCTYPES[n] (0x00268000 + 0x4*n, n=0...63; RW)

For each PCTYPE, this table defines the target PCTYPE to be used for flow director processing.

Field	Bit(s)	Init.	Type	Description
FD_PCTYPE	5:0	0x0	RW	**Flow Directory PCTYPE** Defines the PCTYPE to be used for FD processing if the incoming packet is mapped to PCTYPE #n. By default, an identity mapping is used (default value of filter #n is n).
RESERVED	31:6	0x0	RSV	Reserved.

38.39.2.19.21 Global Queue Filter Control - GLQF_CTL (0x00269BA4; RO)

Field	Bit(s)	Init.	Type	Description
RESERVED	0	0b	RSV	Reserved.
HTOEP	1	0b	RW	**Hash Toeplitz** Hash Toeplitz select for all packet types. 0b = The hash filters are based on a simple 32 bit XOR. 1b = The hash filters are based on the standard Toeplitz scheme, where the hash key is defined per function by the xxQFHKEY registers.
RESERVED	2	0b	RW	**Reserved.**
PCNT_ALLOC	5:3	000b	RW	**Packet Counters Allocation** Controls the GLQF_PCNT counters allocation to the PF as follows: 000b = All counters are exposed to all PFs. 100b = Each PF is allocated 1/2 of the GLQF_PCNT counters. 101b = Each PF is allocated 1/4 of the GLQF_PCNT counters. 110b = Each PF is allocated 1/8 of the GLQF_PCNT counters. 111b = Each PF is allocated 1/16 of the GLQF_PCNT counters. All other values are reserved.
FD_AUTO_PCTYPE	6	0b	RW	**Flow Director Auto PCTYPE** 0b = The *PCTYPE* of FD filter entries are defined by the *PCTYPE* field in the FD programming descriptor. 1b = The *PCTYPE* of FD filter entries are extracted from the programming packet, the same as it is done for received packets.
RESERVED	16:7	0b	RSV	Reserved.
FDBEST	24:17	0x0	RW	**Flow Director Best Effort** Defines the total number of entries in the FD table. It must not exceed the FD table size (equal to 8 KB) minus the sum of *FDALLOC* for all PFs. The global *FDBEST* is defined in granularity of 32 entries.
PROGPRIO	25	0b	RW	**Programming Priority** Priority ordering at filter programming between best effort space and guaranteed space. 0b = At filter programming, hardware tries the guaranteed space first. Only when it is exhausted, does the hardware use the best effort space. 1b = At filter programming, hardware tries the best effort space first. Only when it is exhausted or the PF exhausted its budget in the best effort space, does the hardware use the guaranteed space.

Field	Bit(s)	Init.	Type	Description
INVALPRIO	26	0b	RW	**Invalid Priority** Priority ordering at filter invalidation between best effort space and guaranteed space. 0b = At filter invalidation, hardware first tries to increment the best effort space. The guaranteed space is incremented only when the global best effort space is at it maximum value, or the best effort space of the PF is at its maximum value. 1b = At filter invalidation, hardware first tries to increment its guaranteed space. The best effort space is incremented only when it is already at its maximum value.
IGNORE_IP	27	0b	RW	**Ignore IP** PE quad hash filters ignore (bypass) the IP address table hit/miss indication.
RESERVED	31:28	0x0	RSV	Reserved.

38.39.2.19.22 Global Queue Filter Flow Director Status 0 - GLQF_FDCNT_0 (0x00269BAC; RO)

Field	Bit(s)	Init.	Type	Description
GUARANT_CNT	12:0	0x0	RO	**Guaranteed Count** Total number of FD entries in guaranteed spaces of all PFs.
BESTCNT	25:13	0x0	RO	**Best Effort Count** Total number of FD entries in the best effort space.
RESERVED	31:26	0x0	RSV	Reserved.

38.39.2.19.23 Global Queue Filter Symmetric Hash Enablement - GLQF_HSYM[n] (0x00269D00 + 0x4*n, n=0...63; RW)

Field	Bit(s)	Init.	Type	Description
SYMH_ENA	0	0b	RW	**Symmetric Hash Enable** Enables symmetric hash for PCTYPE ‘n’, where ‘n’ is the register index.
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.19.24 Global Queue Filter Hash Key - GLQF_HKEY[n] (0x00270140 + 0x4*n, n=0...12; RW)

Field	Bit(s)	Init.	Type	Description
KEY_0	7:0	0x0	RW	**Key 0** Toeplitz key byte ‘4*n+0’, where ‘n’ is the register index.
KEY_1	15:8	0x0	RW	**Key 1** Toeplitz key byte ‘4*n+1’, where ‘n’ is the register index.
KEY_2	23:16	0x0	RW	**Key 2** Toeplitz key byte ‘4*n+2’, where ‘n’ is the register index.
KEY_3	31:24	0x0	RW	**Key 3** Toeplitz key byte ‘4*n+3’, where ‘n’ is the register index.

38.39.2.19.25PF Queue Filter Control 2 - PFQF_CTL_2 (0x00270300; RO)

Settings in this register enables hardware to auto-clear internal table pointers, as well as the internal PE quad hash context counter and table pointers of the function.

Field	Bit(s)	Init.	Type	Description
PEHSIZE	4:0	0x0	RW	**PE Hash Size** Defines the number of buckets of the PE quad hash filter table for the VF. It is defined in power of 2 equal to 1 K x 2 ** *PEHSIZE*. The *PEHSIZE* can have any value between 0 and 10. *PEHSIZE* = 0, 1,... 10 is equivalent to 1 K, 2 K, 4 K... 1 M buckets.
PEDSIZE	9:5	0x0	RW	**PEDSIZE** Defines the number of PE quad hash contexts for the VF. It is defined in power of 2 equal to 0.5K x 2 ** *PEDSIZE*. The *PEDSIZE* can have any value between 0 and 9. *PEHSIZE* = 0, 1,... 9 is equivalent to 0.5 K, 1 K, 2 K, 4 K... 256 K contexts.
RESERVED	31:10	0x0	RSV	Reserved.

38.39.2.19.26Global Flow Director PCTYPE Evict Enable - GLQF_FDEVICTENA[n] (0x00270384 + 0x4*n, n=0...1; RW)

Defines the PCTYPES which are candidates for hardware eviction of flow director filters.

Field	Bit(s)	Init.	Type	Description
GLQF_FDEVICTENA	31:0	0x0	RW	**Flow Director Eviction Enable** Defines if a packet matching this PCTYPE is candidate to filter eviction. • For register 'n'=0, any bit 'k' relates to line item 'k' in the Protocol Layers Table. • For register 'n'=1, any bit 'k' relates to line item 32+ 'k' in the Protocol Layers Table.

38.39.2.19.27Global Flow Director Flag Evict Enable - GLQF_FDEVICTFLAG (0x00270280; RW)

Field	Bit(s)	Init.	Type	Description
TX_FLAGS	7:0	0x05	RW	**Transmit Flags** Defines the TCP flags that a Tx packet should match to cause an ATR filter removal: 0 = FIN. 1 = SYN. 2 = Reset. 3 = Push. 4 = ACK. 5 = Urgent flag. 7:6 = Reserved.
RX_FLAGS	15:8	0x05	RW	**Receive Flags** Defines the TCP flags that a Rx packet should match to cause an ATR filter removal: 0 = FIN. 1 = SYN. 2 = Reset. 3 = Push. 4 = ACK. 5 = Urgent flag. 7:6 = Reserved.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.19.28VSI Queue Filter Hash Key - VSIQF_HKEY[n,VSI] (0x002A0000 + 0x800*n + 0x4*VSI, n=0...12, VSI=0...383; RW)

Field	Bit(s)	Init.	Type	Description
KEY_0	7:0	0x0	RW	**Key 0** Toeplitz key byte '4*n+0', where 'n' is the register index.
KEY_1	15:8	0x0	RW	**Key 1** Toeplitz key byte '4*n+1', where 'n' is the register index.
KEY_2	23:16	0x0	RW	**Key 2** Toeplitz key byte '4*n+2', where 'n' is the register index.
KEY_3	31:24	0x0	RW	**Key 3** Toeplitz key byte '4*n+3', where 'n' is the register index.

38.39.2.20 PF - Timesync (IEEE 1588) Registers

38.39.2.20.1 Port Time Sync Control 1 - PRTTSYN_CTL1 (0x00085020; RW)

Field	Bit(s)	Init.	Type	Description
V1MESSTYPE0	7:0	0x01	RW	**V1 Message Type 0** PTP V1 message type 0 for sampled time stamp of received 1588 packets. Default setting is 0x01 for sync and Delay_Req packets. Setting this field to 0xFF is like a wild card option, enabling all PTP V1 packets.
V1MESSTYPE1	15:8	0x01	RW	**V1 Message Type 1** PTP V1 message type 1 for sampled time stamp of received 1588 packets. Default setting is 0x01 for Sync and Delay_Req packets. Setting this field to 0xFF is like a wild card option, enabling all PTP V1 packets.
V2MESSTYPE0	19:16	0x0	RW	**V2 Message Type 0** PTP V2 message type 0 for sampled timestamp of received 1588 packets. Default setting is 0x0 for sync packets. Other values are: 0x1 = Delay Req. 0x2 = Pdelay Req. 0x3 = Pdelay Resp. Setting this field to 0xF is like a wild card option, enabling all PTP V2 packets.
V2MESSTYPE1	23:20	0x1	RW	**V2 Message Type 1** PTP V2 message type 1 for sampled time stamp of received 1588 packets. Default setting is 0x1 for delay request.
TSYNTYPE	25:24	00b	RW	**Time Sync Type** Receive packets types sampled by the 1588 timer. 00b = L2 version 2 packets (message type is defined by the PRTTSYN_CTL1 register). 01b = UDP version 1 packets (UDP ports are enabled by *UDP_ENA* field in this register and *Message Type* field is defined by the PRTTSYN_CTL1 register). 10b = L2 and UDP version 2 packets (UDP ports are enabled by *UDP_ENA* field in this register and message type is defined by the PRTTSYN_CTL1 register). 11b = L2 and UDP version 2 event packets (UDP ports are enabled by *UDP_ENA* field in this register and message type < 8).
UDP_ENA	27:26	00b	RW	**UDP Enable** Enable the UDP ports recognized as 1588 packets. 00b = No UDP packet recognition. 01b = UDP port number equals to 0x013F. 10b = UDP port number equals to 0x0140. 11b = UDP port numbers equals to either 0x013F or 0x140.
RESERVED	30:28	000b	RSV	Reserved.

Field	Bit(s)	Init.	Type	Description
TSYNENA	31	0b	RW	**Time Sync Enable** Enable the 1588 logic. When the *TSYNENA* flag is cleared, the 1588 logic is not functional. This flag must be set the same as the *TSYNENA* flag in the PRTTSYN_CTL0 register (Section 38.39.2.20.15).

38.39.2.20.2 Port Time Sync Receive PTP Packet Time High - PRTTSYN_RXTIME_H[n] (0x00085040 + 0x20*n, n=0...3; RO)

Field	Bit(s)	Init.	Type	Description
RXTIME_H	31:0	0x0	RO	**Receive Time High** 32 MS bits of the receive PTP packet time matches the units of the TSYN_TIME_H register (Section 38.39.2.20.10).

38.39.2.20.3 Port Time Sync Receive PTP Packet Time Low - PRTTSYN_RXTIME_L[n] (0x000850C0 + 0x20*n, n=0...3; RO)

Field	Bit(s)	Init.	Type	Description
RXTIME_L	31:0	0x0	RO	**Receive Time Low** 32 LS bits of the receive PTP packet time matches the units of the TSYN_TIME_L register (Section 38.39.2.20.9).

38.39.2.20.4 Port Time Sync Status 1 - PRTTSYN_STAT_1 (0x00085140; RO)

Field	Bit(s)	Init.	Type	Description
RXT0	0	0b	RO	**Receive Timestamp 0** PRTTSYN_RXTIME[0] register contains valid timestamp. This bit is set by hardware when received packet reception time is captured in the PRTTSYN_RXTIME[0] register, and it is auto-cleared when software reads the PRTTSYN_RXTIME[0] register.
RXT1	1	0b	RO	**Receive Timestamp 1** PRTTSYN_RXTIME[1] register contains valid timestamp. This bit is set by hardware when received packet reception time is captured in the PRTTSYN_RXTIME[1] register, and it is auto-cleared when software reads the PRTTSYN_RXTIME[1] register.
RXT2	2	0b	RO	**Receive Timestamp 2** PRTTSYN_RXTIME[2] register contains valid timestamp. This bit is set by hardware when received packet reception time is captured in the PRTTSYN_RXTIME[2] register, and it is auto-cleared when software reads the PRTTSYN_RXTIME[2] register.
RXT3	3	0b	RO	**Receive Timestamp 3** PRTTSYN_RXTIME[3] register contains valid timestamp. This bit is set by hardware when received packet reception time is captured in the PRTTSYN_RXTIME[3] register. and it is auto-cleared when software reads the PRTTSYN_RXTIME[3] register.
RESERVED	31:4	0x0	RSV	Reserved.

38.39.2.20.5 Port Time Sync Increment Value Low - PRTTSYN_INC_L (0x001E4040; RW)

Field	Bit(s)	Init.	Type	Description
TSYNINC_L	31:0	0x0	RW	**Time Sync Increment Low** 32 LS bits of the increment value added to the 96-bit TSYNTIME registers for each MAC clock.

38.39.2.20.6 Port Time Sync Increment Value High - PRTTSYN_INC_H (0x001E4060; RW)

Field	Bit(s)	Init.	Type	Description
TSYNINC_H	5:0	0x0	RW	**Time Sync Increment High** 6 MS bits of the increment value added to the 96-bit TSYNTIME registers for each MAC clock.
RESERVED	31:6	0x0	RSV	Reserved.

38.39.2.20.7 Port Time Sync Event Time Low - PRTTSYN_EVNT_L[n] (0x001E4080 + 0x20*n, n=0...1; RO)

Field	Bit(s)	Init.	Type	Description
TSYNEVNT_L	31:0	0x0	RO	**Time Synch Event Time Low** 32 LS bit of the sampled event time. The sampled event is defined by *EVNTLVL* field in the PRTTSYN_AUX_0 register (Section 38.39.2.20.19).

38.39.2.20.8 Port Time Sync Event Time High - PRTTSYN_EVNT_H[n] (0x001E40C0 + 0x20*n, n=0...1; RO)

Field	Bit(s)	Init.	Type	Description
TSYNEVNT_H	31:0	0x0	RO	**Time Synch Event Time High** 32 MS bit of the sampled event time of a 1588 event defined by the PRTTSYN_AUX_0 register (Section 38.39.2.20.19).

38.39.2.20.9 Port Time Sync Time Low - PRTTSYN_TIME_L (0x001E4100; RW)

Field	Bit(s)	Init.	Type	Description
TSYNTIME_L	31:0	0x0	RW	**Time Synch Time Low** Bits 32...63 of the 96-bit timer. If the lowest 32 bits define the fraction of ns, this register defines the lower 32 bits of the ns units.

38.39.2.20.10 Port Time Sync Time High - PRTTSYN_TIME_H (0x001E4120; RW)

Field	Bit(s)	Init.	Type	Description
TSYNTIME_H	31:0	0x0	RW	**Time Synch Time High** Upper 32 bit of the 96-bit timer.

38.39.2.20.11 Port Time Sync Target Time Low - PRTTSYN_TGT_L[n] (0x001E4140 + 0x20*n, n=0...1; RW)

Field	Bit(s)	Init.	Type	Description
TSYNTGTT_L	31:0	0x0	RW	**Time Sync Target Time Low** 32 LS bits of the target time of an event out in one of the AUX IO signals.

38.39.2.20.12 Port Time Sync Target Time High - PRTTSYN_TGT_H[n] (0x001E4180 + 0x20*n, n=0...1; RW)

Field	Bit(s)	Init.	Type	Description
TSYNTGTT_H	31:0	0x0	RW	**Time Sync Target Time High** 32 MS bits of the target time of an event out in one of the AUX IO signals.

38.39.2.20.13 Port Time Sync Transmit Packet Time Low - PRTTSYN_TXTIME_L (0x001E41C0; RO)

Field	Bit(s)	Init.	Type	Description
TXTIME_L	31:0	0x0	RO	**Transmit Time Low** 32 LS bits of the sampled 1588 Time of a Tx packet matches the units of the TSYN_TIME_L register (Section 38.39.2.20.9).

38.39.2.20.14 Port Time Sync Transmit Packet Time High - PRTTSYN_TXTIME_H (0x001E41E0; RO)

Field	Bit(s)	Init.	Type	Description
TXTIME_H	31:0	0x0	RO	**Transmit Time High** 32 MS bits of the sampled 1588 Time of a Tx packet matches the units of the TSYN_TIME_H register (Section 38.39.2.20.10).

38.39.2.20.15 Port Time Sync Control 0 - PRTTSYN_CTL0 (0x001E4200; RW)

Field	Bit(s)	Init.	Type	Description
RESERVED	0	0b	RSV	Reserved.
TXTIME_INT_ENA	1	0b	RW	**Transmit Time Interrupt Enable** Interrupt enable when the TSYNTXTIME registers samples the transmission time in this port. The event is reported in the *TXTIME* bit in the PRTTSYN_STAT_0 register (Section 38.39.2.20.16).
EVENT_INT_ENA	2	0b	RW	**Event Interrupt Enable** Interrupt enable when an event is sampled in any of the PRTTSYN_EVNT registers.
TGT_INT_ENA	3	0b	RW	**Target Time Interrupt Enable** Interrupt enable when the target time is sampled in any of the PRTTSYN_TGT registers.
RESERVED	7:4	0x0	RSV	Reserved.
PF_ID	11:8	0x0	RW	**PF ID** Software indication for the PF Function ID that controls the 1588 logic of the port (no hardware impact). This field is expected to be loaded from NVM or set by management agent. During normal operation the PF software driver is not expected to change its setting.
TSYNACT	13:12	00b	RW	**Time Sync Active** Software indication for 1588 mode of operation (no hardware impact). 00b = Inactive agent. 01b = Synchronized to local time. 10b = Synchronized to the standard TAI. 11b = Reserved.
RESERVED	30:14	0x0	RSV	Reserved.
TSYNENA	31	0b	RW	**Time Sync Enable** Enables the 1588 logic. When the *TSYNENA* flag is cleared, the 1588 logic is not functional. This flag must be set the same as the *TSYNENA* flag in the PRTTSYN_CTL1 register (Section 38.39.2.20.1).

38.39.2.20.16Port Time Sync Status 0 - PRTTSYN_STAT_0 (0x001E4220; RCW)

Field	Bit(s)	Init.	Type	Description
EVENT0	0	0b	RCW	**Event Time 0** Set to one when the PRTTSYN_EVNT[0] captures an input event times tamp.
EVENT1	1	0b	RCW	**Event Time 1** Set to one when the PRTTSYN_EVNT[1] captures an input event time stamp.
TGT0	2	0b	RCW	**Target Time 0** Set to one when the PRTTSYN_TGT[0] timer is expired.
TGT1	3	0b	RCW	**Target Time 1** Set to one when the PRTTSYN_TGT[1] timer is expired.
TXTIME	4	0b	RCW	**Transmit Time** Set to one when the PRTTSYN_TXTIME register samples a Tx packet.
RESERVED	31:5	0x0	RSV	Reserved.

38.39.2.20.17Port Time Sync Clock Out Duration - PRTTSYN_CLKO[n] (0x001E4240 + 0x20*n, n=0...1; RW)

Field	Bit(s)	Init.	Type	Description
TSYNCLKO	31:0	0x0	RW	**Time Sync Clock Out** Clock output duration as described in the Auxiliary 1588 IO Signals section.

38.39.2.20.18Port Time Sync Adjustment - PRTTSYN_ADJ (0x001E4280; RW)

Field	Bit(s)	Init.	Type	Description
TSYNADJ	30:0	0x0	RW	**Time Sync Adjustment** Absolute value of the time adjust matched to the units of the TSYN_TIME_L register (Section 38.39.2.20.9).
SIGN	31	0b	RW	**Sign** The sign of the time adjustment. 0b = Positive adjustment 1b = Negative adjustment

38.39.2.20.19Port Time Sync AUX Control 0 - PRTTSYN_AUX_0[n] (0x001E42A0 + 0x20*n, n=0...1; RW)

Field	Bit(s)	Init.	Type	Description
OUT_ENA	0	0b	RW	**Output Enable** Synchronized output enablement. When set to 1b, the synchronized output signal is enabled according to the other parameters in this register.
OUTMOD	2:1	00b	RW	**Output Mode** Output signal mode of operation 00b = Output Level Mode. 01b = Flipped Output Mode. 10b = Output Pulse Mode. 11b = Output Clock Mode. The GPIO signals should be set as 1588 output by the GLGEN_GPIO_CTL[n] registers.
OUTLVL	3	0b	RW	**Output Level** Output level driven on the IO signal at the target time.
RESERVED	7:4	0x0	RSV	Reserved.
PULSEW	11:8	0x0	RW	**Pulse Width** Output pulse width for output pulse mode equals 16 x (*PULSEW* + 1) clocks. The clock frequency is defined per link speed in the 1588 Clock Registers section.

Field	Bit(s)	Init.	Type	Description
RESERVED	15:12	0x0	RSV	Reserved.
EVNTLVL	17:16	00b	RW	**Event Level** Event level on the I/O signal configured as 1588 input. It can be set to one of the following options: 00b = Disable. 01b = Rising edge. 10b = Falling edge. 11b = Any transition. The GPIO signals should be set as 1588 input by the GLGEN_GPIO_CTL[n] registers.
RESERVED	31:18	0x0	RSV	Reserved.

38.39.2.20.20 Port Time Sync AUX Control 1 - PRTTSYN_AUX_1[n] (0x001E42E0 + 0x20*n, n=0...1; RW)

Field	Bit(s)	Init.	Type	Description
INSTNT	0	0b	RW	**Instant** If this flag is set, the OUTSIG signal is forced to OUTLVL value. This flag is auto-cleared by hardware.
SAMPLE_TIME	1	0b	RW	**Sampling Time** Setting this flag triggers instant sampling of the PRTTSYN_TIME to the matched PRTTSYN_EVNT register. This flag is auto-cleared by hardware.
RESERVED	31:2	0x0	RSV	Reserved.

38.39.2.21 PF - Protocol Engine Registers

38.39.2.21.1 Protocol Engine VF CQP Doorbell - VFPE_CQPDB[VF] (0x00000000 + 0x4*VF, VF=0...127; RW)

This register is used to post work of the PE control QP. Software can determine if CQP has pending work by comparing *WQHEAD* to *WQTAIL* after reading this register. Software must first populate one or more WQEs in the CQP WQ and then put the index of the WQE following the last populated WQE into *WQHEAD* in order to submit work to CQP.

Field	Bit(s)	Init.	Type	Description
WQHEAD	10:0	0x0	RW	**Work Queue Head** Indicates the WQE index of the next WQE that software posts to the CQP.
RESERVED	31:11	0x0	RSV	Reserved.

38.39.2.21.2 Protocol Engine VF CQP Tail - VFPE_CQPTAIL[VF] (0x00000400 + 0x4*VF, VF=0...127; RO)

This register is used to determine how much work the PE control QP has pending. Software can determine if CQP has pending work by comparing the last value written to *WQHEAD* in the CQPDB register to *WQTAIL* after reading this register.

Field	Bit(s)	Init.	Type	Description
WQTAIL	10:0	0x0	RW	**Work Queue Tail** Indicates the WQE index of the next WQE that the CQP processes.
RESERVED	30:11	0x0	RSV	Reserved.
CQP_OP_ERR	31	0b	RW	**CQP Operation Error** Indicates that CQP encountered an error processing an operation. If software has multiple requests outstanding to CQP at the time of the error, *WQTAIL* might not indicate the WQE that caused the error.

38.39.2.21.3 Protocol Engine VF Create CQP Status - VFPE_CCQPSTATUS[VF] (0x00000800 + 0x4*VF, VF=0...127; RW)

This register is used to indicate the progress of creating the control QP for a given PCI function. During host software initialization, these bits are initially zero. Each bit is set when the associated step of the initialization process completes.

Field	Bit(s)	Init.	Type	Description
CCQP_DONE	0	0b	RW	**Create CQP Done** 0b = Indicates that CQP has not been created. 1b = Indicates that the create CQP operation triggered by writing to PECCQPHIGH and PECCQPLOW completes.
RESERVED	3:1	000b	RSV	Reserved.
HMC_PROFILE	6:4	000b	RW	**Host Memory Cache (HMC) Profile** Specifies the HMC resource profile that is active. 000b = Reserved. 001b = Default. 010b = SR-IOV VF primary. 011b = SR-IOV even distribution. All other values are reserved.
RESERVED	15:7	0x0	RSV	Reserved.
RDMA_EN_VFS	21:16	0x0	RW	**RDMA Enabled VFs** Specifies the number of RDMA enabled VFs allocated in the HMC resource profile.
RESERVED	30:22	0x0	RSV	Reserved.
CCQP_ERR	31	0b	RW	**Create CQP Error** Indicates that the CQP encountered an error processing the last create CQP request. This bit is reset when CQP is not created.

38.39.2.21.4 Protocol Engine VF Create CQP Low - VFPE_CCQPLOW[VF] (0x00000C00 + 0x4*VF, VF=0...127; RW)

This register stores the lower 32 bits of the 64-bit physical address of the control QP context for its associated PCI function. Under host software control, each PCI function uses its Create Control QP High/Low registers to create its corresponding Control QP (QP0). The 64-bit address must always be updated by writing the *PECCQPLOW* field last.

Field	Bit(s)	Init.	Type	Description
PECCQPLOW	31:0	0x0	RW	**PE Create CQP Low** Least significant bits of the control QP context physical address in host memory.

38.39.2.21.5 Protocol Engine VF Create CQP High - VFPE_CCQPHIGH[VF] (0x00001000 + 0x4*VF, VF=0...127; RW)

This register stores the upper 32 bits of the 64-bit physical address of the control QP context for its associated PCI function.

Field	Bit(s)	Init.	Type	Description
PECCQPHIGH	31:0	0x0	RW	**PE Create CQP High** Most significant bits of the control QP context physical address in host memory.

38.39.2.21.6 Protocol Engine VF IP Config 0 - VFPE_IPCONFIG0[VF] (0x00001400 + 0x4*VF, VF=0...127; RW)

This register is used to set or view the *IPID* field that the PE writes into the IP header. The PE increments this value for each outgoing IP datagram.

Field	Bit(s)	Init.	Type	Description
PEIPID	15:0	0x0	RW	**PE IP Identification** Specifies the *IP Identification* field used for IPv4 IP header generation. This register is initialized by firmware or software device driver, and incremented by hardware with each IPv4 datagram transmitted by PE both UDA and iWARP on the given PCI function.
USEENTIREIDRANGE	16	0b	RW	**Use Entire ID Range** Specifies that the PE should use the entire 16-bit range for the IPID value.
RESERVED	31:17	0x0	RSV	Reserved.

38.39.2.21.7 Protocol Engine CQP Error Codes - VFPE_CQPERRCODES[VF] (0x00001800 + 0x4*VF, VF=0...127; RO)

This register reports errors encountered by CQP when CQ0 is not available. The contents of this register are only valid when the associated CCQPSTATUS.*CCQP_ERR* bit or CQPTAIL.*CQP_ERR* bit is set.

Field	Bit(s)	Init.	Type	Description
CQP_MINOR_CODE	15:0	0x0	RW	**CQP Minor Code** Minor code that would have been reported in a CQP completion.
CQP_MAJOR_CODE	31:16	0x0	RW	**CQP Major Code** Major code that would have been reported in a CQP completion.

38.39.2.21.8 Protocol Engine VF TCP Now Timer - VFPE_TCPNOWTIMER[VF] (0x00002C00 + 0x4*VF, VF=0...127; RO)

Field	Bit(s)	Init.	Type	Description
TCP_NOW	31:0	0x0	RO	**TCP Now** Contains the current value of tcp_now. *TCP_NOW* is a 32-bit counter and provides the TCP time measurement for the all of the timers. It is also used to calculate the TS value sent in the TCP time stamp option, tcp_now is added to tsval_tick_delta to form the TS value.

38.39.2.21.9 Protocol Engine VF MRTE Index Mask - VFPE_MRTEIDXMASK[VF] (0x00003000 + 0x4*VF, VF=0...127; RW)

Field	Bit(s)	Init.	Type	Description
MRTEIDXMASKBITS	4:0	0x0	RW	**MRTE Index Mask Bits** Specifies the number of bits to be used for the MRTE index from the software device driver portion of STag. The remaining bits (24 - *MRTEIDEMASKBITS*) are randomized by the software device driver. The minimum value for this field is 14 and the maximum is 22. Values outside of this range are normalized to 14 or 22 by the hardware.
RESERVED	31:5	0x0	RSV	Reserved.

38.39.2.21.10 Protocol Engine VF Unexpected Error - VFPE_RCVUNEXPECTEDERROR[VF] (0x00003400 + 0x4*VF, VF=0...127; RO)

Field	Bit(s)	Init.	Type	Description
TCP_RX_UNEXP_ERR	23:0	0x0	RW	**TCP Received Unexpected Error** For PE accelerated connections only, counts the total number of Ethernet frames received by the product that failed TCP sanity checks indicating unexpected filtering error in preceding module.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.21.11 Protocol Engine CQP Doorbell - PFPE_CQPDB (0x00008000; RW)

This register is used to post the work of the PE control QP. Software can determine if CQP has pending work by comparing WQHEAD to WQTAIL after reading this register. Software must first populate one or more WQEs in the CQP WQ and then put the index of the WQE following the last populated WQE into *WQHEAD* in order to submit work to CQP.

Field	Bit(s)	Init.	Type	Description
WQHEAD	10:0	0x0	RW	**Work Queue Head** Indicates the WQE index of the next WQE that software posts to the CQP.
RESERVED	31:11	0x0	RSV	Reserved.

38.39.2.21.12Protocol Engine CQP Tail - PFPE_CQPTAIL (0x00008080; RO)

This register is used to determine how much work the PE control QP has pending. Software can determine if the CQP has pending work by comparing the last value written to *WQHEAD* in the CQPDB register to *WQTAIL* after reading this register. This register is updated after the CQP operation completes.

Field	Bit(s)	Init.	Type	Description
WQTAIL	10:0	0x0	RW	**Work Queue Tail** Indicates the WQE index of the next WQE that the CQP processes.
RESERVED	30:11	0x0	RSV	Reserved.
CQP_OP_ERR	31	0b	RW	**CQP Operation Error** Indicates that the CQP encountered an error processing an operation. If software has multiple requests outstanding to the CQP at the time of the error, *WQTAIL* might not indicate the WQE that caused the error.

38.39.2.21.13Protocol Engine Create CQP Status - PFPE_CCQPSTATUS (0x00008100; RW)

This register is used to indicate the progress of creating the control QP for a given PCI function. During host software initialization, these bits are initially 0. Each bit is set when the associated step of the initialization process completes.

Field	Bit(s)	Init.	Type	Description
CCQP_DONE	0	0b	RW	**Create CQP Done** 0b = Indicates that the CQP has not been created. 1b = Indicates that the create CQP operation triggered by writing to PECCQPHIGH and PECCQPLOW completed.
RESERVED	3:1	000b	RSV	Reserved.
HMC_PROFILE	6:4	000b	RW	**HMC Profile** Specifies the HMC resource profile that is active. 000b = Reserved. 001b = Default. 010b = SR-IOV VF primary. 011b = SR-IOV even distribution. All other values are reserved.
RESERVED	15:7	0x0	RSV	Reserved.
RDMA_EN_VFS	21:16	0x0	RW	**RDMA Enabled VFs** Specifies the number of RDMA enabled VFs allocated in the HMC resource profile.
RESERVED	30:22	0x0	RSV	Reserved.
CCQP_ERR	31	0b	RW	**Create CQP Error** Indicates that the CQP encountered an error processing the last create CQP request. This bit is reset when CQP is not created.

38.39.2.21.14Protocol Engine Create CQP Low - PFPE_CCQPLOW (0x00008180; RW)

This register stores the lower 32 bits of the 64-bit physical address of the control QP context for its associated PCI function. Under host software control, each PCI function uses its Create Control QP High/Low registers to create its corresponding Control QP.

Field	Bit(s)	Init.	Type	Description
PECCQPLOW	31:0	0x0	RW	**PE Create CQP Low** Least significant bits of the control QP context physical address in host memory.

38.39.2.21.15Protocol Engine Create CQP High - PFPE_CCQPHIGH (0x00008200; RW)

This register stores the upper 32 bits of the 64-bit physical address of the control QP context for its associated PCI function.

Field	Bit(s)	Init.	Type	Description
PECCQPHIGH	31:0	0x0	RW	**PE Create CQP High** Most significant bits of the control QP context physical address in host memory.

38.39.2.21.16Protocol Engine IP Config 0 - PFPE_IPCONFIG0 (0x00008280; RW)

This register is used to set or view the *IPID* field that the PE writes into the IP header. The PE increments this value for each outgoing IP datagram.

Field	Bit(s)	Init.	Type	Description
PEIPID	15:0	0x0	RW	**PE IP Identification** Specifies the *IP Identification* field used for IPv4 IP header generation. This register is initialized by firmware or the software device driver, and incremented by hardware with each IPv4 datagram transmitted by PE both UDA and iWARP on the given PCI function.
USEENTIREIDRANGE	16	0b	RW	**Use Entire ID Range** Specifies that the PE should use the entire 16-bit range for the IPID value.
RESERVED	31:17	0x0	RSV	Reserved.

38.39.2.21.17Protocol Engine TCP Now Timer - PFPE_TCPNOWTIMER (0x00008580; RO

Field	Bit(s)	Init.	Type	Description
TCP_NOW	31:0	0x0	RO	**TCP Now** Contains the current value of tcp_now. *TCP_NOW* is a 32-bit counter and provides the TCP time measurement for the all of the timers. It is also used to calculate the TS value sent in the TCP time stamp option, tcp_now is added to tsval_tick_delta to form the TS value.

38.39.2.21.18Protocol Engine MRTE Index Mask - PFPE_MRTEIDXMASK (0x00008600; RW)

Field	Bit(s)	Init.	Type	Description
MRTEIDXMASKBITS	4:0	0x0	RW	**MRTE Index Mask Bits** Specifies the number of bits to be used for the MRTE index from the software device driver portion of STag. The remaining bits (24 - *MRTEIDEMASKBITS*) are randomized by the software device driver. The minimum value for this field is 14 and the maximum is 22. Values outside of this range are normalized to 14 or 22 by hardware.
RESERVED	31:5	0x0	RSV	Reserved.

38.39.2.21.19 Protocol Engine Unexpected Error - PFPE_RCVUNEXPECTEDERROR (0x00008680; RO)

Field	Bit(s)	Init.	Type	Description
TCP_RX_UNEXP_ERR	23:0	0x0	RW	**TCP Received Unexpected Error** For PE accelerated connections only, counts the total number of Ethernet frames received by the product that failed TCP sanity checks indicating unexpected filtering error in preceding module.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.21.20 Protocol Engine UDA Control Register - PFPE_UDACTRL (0x00008700; RW)

This register is used to set up the user-space direct access acceleration configuration.

Field	Bit(s)	Init.	Type	Description
IPV4MCFRAGRESBP	0	0b	RW	**IPv4 Multicast Fragment Resolution Bypass** Indicates whether IPv4/UDP multicast fragments are allowed to bypass fragment resolution phase, and be replicated based on the destination IP address and multicast group context configuration. 0b = All UDA accelerated UDP multicast fragments must have their destination UDP port resolved, and replicated based on pair of destination IP address and destination UDP port. 1b = All UDA accelerated UDP multicast fragments should bypass fragment resolution logic and be replicated based on Multicast Group Context configuration.
IPV4UCFRAGRESBP	1	0b	RW	**IPv4 Unicast Fragment Resolution Bypass** Indicates whether IPv4/UDP unicast fragments are allowed to bypass fragment resolution logic. 0b = UDP unicast fragments must be processed by fragment resolution logic. 1b = UDP unicast fragments can bypass fragment resolution logic, and be processed based on configuration of PF/VFPE_UDAUCFBQPN register.
IPV6MCFRAGRESBP	2	0b	RW	**IPv6 Multicast Fragment Resolution Bypass** Indicates whether IPv6/UDP multicast fragments are allowed to bypass fragment resolution phase, and be replicated based on the destination IP address. and multicast group context configuration. 0b = All UDA accelerated UDP multicast fragments must have their destination UDP port resolved, and replicated based on pair of destination IP address and destination UDP port. 1b = All UDA accelerated UDP multicast fragments should bypass fragment resolution logic and be replicated based on Multicast Group Context configuration.
IPV6UCFRAGRESBP	3	0b	RW	**IPv6 Unicast Fragment Resolution Bypass** Indicates whether IPv6/UDP unicast fragments are allowed to bypass fragment resolution logic. 0b = UDP unicast fragments must be processed by fragment resolution logic. 1b = UDP unicast fragments can bypass fragment resolution logic, and be processed based on configuration of PF/VFPE_UDAUCFBQPN register
UDPMCFRAGRESFAIL	4	0b	RW	**UDP Multicast Fragment Resolution Failed** Defines handling of the multicast fragment that failed fragment resolution. 0b = Fragment should be silently dropped. 1b = Fragment should be replicated based on Multicast Group Context Configuration.
RESERVED	31:5	0x0	RSV	Reserved.

38.39.2.21.21 Protocol Engine UDA Unicast Fallback QP Number - PFPE_UDAUCFBQPN (0x00008780; RW)

This register specifies an UDA QP Number that should be used as a target QP for all UDP unicast fragments that failed fragment resolution. If QPN is not valid, fragment should be silently dropped.

Field	Bit(s)	Init.	Type	Description
QPN	17:0	0x0	RW	**QPN** UDA QPN.
RESERVED	30:18	0x0	RSV	Reserved.
VALID	31	0b	RW	**Valid** 0b = QPN is not valid. 1b = QPN is valid.

38.39.2.21.22 Protocol Engine CQP Error Codes - PFPE_CQPERRCODES (0x00008880; RO)

This register reports errors encountered by CQP when CQ0 is not available. The contents of this register are only valid when the associated CCQPSTATUS.*CCQP_ERR* bit or CQPTAIL.*CQP_ERR* bit is set.

Field	Bit(s)	Init.	Type	Description
CQP_MINOR_CODE	15:0	0x0	RW	**CQP Minor Code** Minor code that would have been reported in a CQP completion.
CQP_MAJOR_CODE	31:16	0x0	RW	**CQP Major Code** Major code that would have been reported in a CQP completion.

38.39.2.21.23 Protocol Engine FLM transmit Allocate Error - PFPE_FLMXMITALLOCERR (0x00008900; RO)

This register holds a count of the transmit free list allocation errors.

Field	Bit(s)	Init.	Type	Description
ERROR_COUNT	15:0	0x0	RO	**Error Count** The number of failed free list allocations for this function. Counter does not roll over. Software read resets.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.21.24 Protocol Engine FLM Q1 Allocate Error - PFPE_FLMQ1ALLOCERR (0x00008980; RO)

This register holds a count of the Q1 free list allocation errors.

Field	Bit(s)	Init.	Type	Description
ERROR_COUNT	15:0	0x0	RO	**Error Count** The number of failed free list allocations for this function. Counter does not roll over. Software read resets.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.21.25 Protocol Engine UDA Control Register - GLPE_VFUDACTRL[n] (0x0000C000 + 0x4*n, n=0...31; RW)

This register is used to setup user-space direct access acceleration configuration. Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
IPV4MCFRAGRESBP	0	0b	RW	**IPv4 Multicast Fragment Resolution Bypass** Indicates whether IPv4/UDP multicast fragments are allowed to bypass fragment resolution phase, and be replicated based on the destination IP address and multicast group context configuration. 0b = All UDA accelerated UDP multicast fragments must have their destination UDP port resolved, and replicated based on pair of destination IP address and destination UDP port. 1b = All UDA accelerated UDP multicast fragments should bypass fragment resolution logic and be replicated based on multicast group context configuration.
IPV4UCFRAGRESBP	1	0b	RW	**IPv4 Unicast Fragment Resolution Bypass** Indicates whether IPv4/UDP unicast fragments are allowed to bypass fragment resolution logic. 0b = UDP unicast fragments must be processed by fragment resolution logic. 1b = UDP unicast fragments can bypass fragment resolution logic, and be processed based on configuration of PF/VFPE_UDAUCFBQPN register.
IPV6MCFRAGRESBP	2	0b	RW	**IPv6 Multicast Fragment Resolution Bypass** Indicates whether IPv6/UDP multicast fragments are allowed to bypass fragment resolution phase, and be replicated based on the destination IP address. and multicast group context configuration. 0b = All UDA accelerated UDP multicast fragments must have their destination UDP port resolved, and replicated based on pair of destination IP address and destination UDP port. 1b = All UDA accelerated UDP multicast fragments should bypass fragment resolution logic and be replicated based on multicast group context configuration.
IPV6UCFRAGRESBP	3	0b	RW	**IPv6 Unicast Fragment Resolution Bypass** Indicates whether IPv6/UDP unicast fragments are allowed to bypass fragment resolution logic. 0b = UDP unicast fragments must be processed by fragment resolution logic. 1b = UDP unicast fragments can bypass fragment resolution logic, and be processed based on configuration of PF/VFPE_UDAUCFBQPN register
UDPMCFRAGRESFAIL	4	0b	RW	**UDP Multicast Fragment Resolution Failed** Defines handling of the multicast fragment that failed fragment resolution. 0b = Fragment should be silently dropped. 1b = Fragment should be replicated based on multicast group context Configuration.
RESERVED	31:5	0x0	RSV	Reserved.

38.39.2.21.26 Protocol Engine UDA Unicast Fallback QP Number - GLPE_VFUDAUCFBQPN[n] (0x0000C100 + 0x4*n, n=0...31; RW)

This register specifies an UDA QP number that should be used as a target QP for all UDP unicast fragments that failed fragment resolution. Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index. If QPN is not valid, fragment should be silently dropped.

Field	Bit(s)	Init.	Type	Description
QPN	17:0	0x0	RW	**QPN** UDA QPN.
RESERVED	30:18	0x0	RSV	Reserved.
VALID	31	0b	RW	**Valid** 0b = QPN is not valid. 1b = QPN is valid.

38.39.2.21.27 Protocol Engine VF FLM transmit Allocate Error - GLPE_VFFLMXMITALLOCERR[n] (0x0000C600 + 0x4*n, n=0...31; RO)

This register holds a count of the transmit free list allocation errors.

Field	Bit(s)	Init.	Type	Description
ERROR_COUNT	15:0	0x0	RO	**Error Count** The number of failed free list allocations for this function. Counter does not roll over. Software read resets.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.21.28 Protocol Engine VF FLM Q1 Allocate Error - GLPE_VFFLMQ1ALLOCERR[n] (0x0000C700 + 0x4*n, n=0...31; RO)

This register holds a count of the Q1 free list allocation errors.

Field	Bit(s)	Init.	Type	Description
ERROR_COUNT	15:0	0x0	RO	**Error Count** The number of failed free list allocations for this function. Counter does not roll over. Software read resets
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.21.29 Protocol Engine CPU Status 0 - GLPE_CPUSTATUS0 (0x0000D040; RW)

Field	Bit(s)	Init.	Type	Description
PECPUSTATUS0	31:0	0x0	RW	**PE CPU Status 0** Provides the host with the current status of one of the PE internal CPUs. Typically, the exclusive writer of this register is CQP, and the exclusive reader is host software, but other usage models are supported. Details on the meaning of each status code are not defined in this datasheet and might change with a firmware revision.

38.39.2.21.30 Protocol Engine CPU Status 1 - GLPE_CPUSTATUS1 (0x0000D044; RW)

Field	Bit(s)	Init.	Type	Description
PECPUSTATUS1	31:0	0x0	RW	**PE CPU Status 1** Provides the host with the current status of one of the PE internal CPUs. Typically, the exclusive writer of this register is TEP, and the exclusive reader is host software, but other usage models are supported. Details on the meaning of each status code are not defined in this datasheet and might change with a firmware revision.

38.39.2.21.31 Protocol Engine CPU Status 2 - GLPE_CPUSTATUS2 (0x0000D048; RW)

Field	Bit(s)	Init.	Type	Description
PECPUSTATUS2	31:0	0x0	RW	**PE CPU Status 2** Provides the host with the current status of one of the PE internal CPUs. Typically, the exclusive writer of this register is OOP, and the exclusive reader is host software, but other usage models are supported. Details on the meaning of each status code are not defined in this datasheet and might change with a firmware revision.

38.39.2.21.32 Protocol Engine CPU Trigger 0 - GLPE_CPUTRIG0 (0x0000D060; RW)

This register provides the host and other CPUs with the ability to signal another CPU that CPU-specific registers have been changed, but other usage models are supported.

Field	Bit(s)	Init.	Type	Description
PECPUTRIG0	15:0	0x0	RW	**PE CPU Trigger 0** Host software writes 0x1 to request CQP to re-read register configuration. A write of 0x0 has no effect. Hardware auto-clears these bits when this register is read by the PE CPU.
RESERVED	16	0b	RSV	Reserved.
TEPREQUEST0	17	0b	RW	**TEP Request 0** TEP writes 1b to request CQP to re-read register configuration. A write of 0b has no effect. Hardware auto-clears this bit when this register is read by the PE CPU.
OOPREQUEST0	18	0b	RW	**OOP Request 0** OOP writes 1b to request CQP to re-read register configuration. A write of 0b has no effect. Hardware auto-clears this bit when this register is read by the PE CPU.
RESERVED	31:19	0x0	RSV	Reserved.

38.39.2.21.33 Protocol Engine VF FLM Object Control - GLPE_VFFLMOBJCTRL[n] (0x0000D400 + 0x4*n, n=0...31; RW)

This register is used for FLM VF object control.

Field	Bit(s)	Init.	Type	Description
XMIT_BLOCKSIZE	2:0	001b	RW	**Transmit Block Size** The number of transmit entries per block using a power of two encoding. Entries per block = 2^XMIT_BLOCKSIZE The supported range is 2^1, 2^2, 2^3, or 2, 4, 8.
RESERVED	7:3	0x0	RSV	Reserved.
Q1_BLOCKSIZE	10:8	000b	RW	**Q1 Block Size** The number of Q1 entries per block using a power of two encoding. entries per block = 2^Q1_BLOCKSIZE The supported range is 2^0, 2^1, 2^2, 2^3, or 1, 2, 4, 8.
RESERVED	31:11	0x0	RSV	Reserved.

38.39.2.21.34 Protocol Engine RUPM General Control Register - GLPE_RUPM_GCTL (0x0000DA00; RW)

General control register for PE upper pipe monitor.

Field	Bit(s)	Init.	Type	Description
ALLOFFTH	7:0	0x0	RW	**All PFC-enabled TCs OFF Threshold** High threshold on the total number of spads allocated to PFC TCs. Only valid when spads are shared across all ports (*SWLB_MODE* set to 1b).
RESERVED	25:8	0x0	RSV	Reserved.
RUPM_P0_DIS	26	0b	RW	**RDMA Upper Pipe Monitor Port 0 Disable** When set, disables the regular pipe monitor of port 0.
RUPM_P1_DIS	27	0b	RW	**RDMA Upper Pipe Monitor Port 1 Disable** When set, disables the regular pipe monitor of port 1.
RUPM_P2_DIS	28	0b	RW	**RDMA Upper Pipe Monitor Port 2 Disable** When set, disables the regular pipe monitor of port 2.
RUPM_P3_DIS	29	0b	RW	**RDMA Upper Pipe Monitor Port 3 Disable** When set, disables the regular pipe monitor of port 3.
RUPM_DIS	30	0b	RW	**RDMA Upper Pipe Monitor Disable** 0b = Enabled. 1b = Disabled (all full and flow control signals are driven to 0).
SWLB_MODE	31	0b	RW	**RDMA Upper Pipe Monitor spad buffer mode** Per port spad allocation. 0b = spad allocation shared across ports. 1b = This bit should be set to 0b.

38.39.2.21.35 Protocol Engine RUPM DUAL40 Config Register - GLPE_DUAL40_RUPM (0x0000DA04; RW)

Field	Bit(s)	Init.	Type	Description
RESERVED	31:0	0x0	RSV	Reserved.

38.39.2.21.36 Protocol Engine RUPM Enable Register - GLPE_RUPM_TXHOST_EN (0x0000DA08; RW)

Field	Bit(s)	Init.	Type	Description
TXHOST_EN	0	1b	RW	**Host Transmit Enable** When cleared, no new transmit spads are provided to the host. It affects PE traffic. Those spads already allocated are processed for completion.
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.21.37 Protocol Engine RUPM Spad Thresholds - PRTPE_RUPM_THRES (0x0000DA20; RW)

This register sets minimum and maximum thresholds for spads across ports and TCs. It is independent of PFC or LL/bulk thresholds.

Field	Bit(s)	Init.	Type	Description
MINSPADSPERTC	7:0	0x2	RW	**Minimum Spads per TC** Minimum number of spads per TC on the port.
MAXSPADS	15:8	0x1B	RW	**Maximum Spads** Maximum number of spads on the port.

Field	Bit(s)	Init.	Type	Description
MAXSPADSPERTC	23:16	0x1B	RW	**Maximum Spads per TC** Maximum number of spads per TC on the port.
RESERVED	31:24	0x0	RSV	Reserved.

38.39.2.21.38 Protocol Engine RUPM Control Register - PRTPE_RUPM_CTL (0x0000DA40; RW)

This register holds RUPM control bits.

Field	Bit(s)	Init.	Type	Description
RESERVED	12:0	0x0	RSV	Reserved.
LLTC	20:13	0x0	RW	**Low Latency TC** Per bit TC configuration. 0b = TC is bulk. 1b = TC is LL.
RESERVED	29:21	0x0	RSV	Reserved.
RUPM_MODE	30	0b	RW	**Per TC RUPM Mode** 0b = Conditioned XOFF forwarding mode (default). XOFF notifications received from RLPM are internally forwarded upward to the Tx-scheduler only once the TUPM for this TC is full. This mode provides better XON recovering time. 1b = Immediate XOFF forwarding mode. XOFF notification received form the TC data pipe monitor are immediately forwarded internally upward to the Tx-scheduler. This mode is useful to support PFC-enabled TCs of the port like true independent TCs when more than two such TCs are configured over the port. This bit should be set to 0b.
RESERVED	31	0b	RSV	Reserved.

38.39.2.21.39 Protocol Engine RUPM PFC Control Register - PRTPE_RUPM_PFCCTL (0x0000DA60; RW)

Field	Bit(s)	Init.	Type	Description
TC2PFC	7:0	0x0	RW	**Per bit TC configuration** 0b = TC ignores PFC thresholds. 1b = TC is configured as PFC.
RESERVED	31:8	0x0	RSV	Reserved.

38.39.2.21.40 Protocol Engine RUPM PFC Port Control Register - PRTPE_RUPM_PFCPC (0x0000DA80; RW)

This register configures the port threshold for PFC traffic.

Field	Bit(s)	Init.	Type	Description
PORTOFFTH	7:0	0x0	RW	**Port OFF Threshold** High threshold on the total amount of spads in WQM for all PFC-enabled TCs.
RESERVED	31:8	0x0	RSV	Reserved.

38.39.2.21.41 Protocol Engine RUPM PFC TC Control Register - PRTPE_RUPM_PFCTCC (0x0000DAA0; RW)

This register configures the TCs for PFC traffic.

Field	Bit(s)	Init.	Type	Description
TCOFFTH	7:0	0x0	RW	**TC OFF Threshold** The total amount of spads that can accumulate in WQM for a single PFC-enabled TC.
RESERVED	15:8	0x0	RSV	Reserved.
LL_PRI_TH	23:16	0x0	RW	**Low Latency PRI Threshold** Port Low Latency spad threshold.
RESERVED	30:24	0x0	RSV	Reserved.
LL_PRI_EN	31	0b	RW	**Low Latency PRI Enable** 0b = LL_PRI_THRESH is ignored. 1b = Block bulk (non-Low latency) traffic when total port spads exceeds LL_PRI_THRESH.

38.39.2.21.42 Protocol Engine RUPM Push Pool - GLPE_RUPM_PUSHPOOL (0x0000DAC0; RW)

Field	Bit(s)	Init.	Type	Description
PUSHSPADS	7:0	0x0	RW	**Push Spads** Number of spads reserved for push WQEs.
RESERVED	31:8	0x0	RSV	Reserved.

38.39.2.21.43 Protocol Engine RUPM FLR Pool - GLPE_RUPM_FLRPOOL (0x0000DAC4; RW)

Field	Bit(s)	Init.	Type	Description
FLRSPADS	7:0	0x8	RW	**FLR Spads** Number of spads reserved for FLR markers.
RESERVED	31:8	0x0	RSV	Reserved.

38.39.2.21.44 Protocol Engine RUPM PTX Pool - GLPE_RUPM_PTXPOOL (0x0000DAC8; RW)

Field	Bit(s)	Init.	Type	Description
PTXSPADS	7:0	0x12	RW	**PTX Spads** Number of spads reserved for PTX spads.
RESERVED	31:8	0x0	RSV	Reserved.

38.39.2.21.45 Protocol Engine RUPM CQP Pool - GLPE_RUPM_CQPPOOL (0x0000DACC; RW)

Field	Bit(s)	Init.	Type	Description
CQPSPADS	7:0	0x8	RW	**CQP Spads** Number of spads reserved for CQP flush markers.
RESERVED	31:8	0x0	RSV	Reserved.

38.39.2.21.46 Protocol Engine RUPM TC Counters 0-3 - PRTE_RUPM_TCCNTR03 (0x0000DAE0; RO)

Field	Bit(s)	Init.	Type	Description
TC0COUNT	7:0	0x0	RO	**TC0 Count** Number of spads on TC0 not yet sent to PTX.
TC1COUNT	15:8	0x0	RO	**TC1 Count** Number of spads on TC1 not yet sent to PTX.
TC2COUNT	23:16	0x0	RO	**TC2 Count** Number of spads on TC2 not yet sent to PTX.
TC3COUNT	31:24	0x0	RO	**TC3 Count** Number of spads on TC3 not yet sent to PTX.

38.39.2.21.47 Protocol Engine RUPM TC Counters 4-7 - PRTPE_RUPM_TCCNTR47 (0x0000DB00; RO)

Field	Bit(s)	Init.	Type	Description
TC4COUNT	7:0	0x0	RO	**TC4 Count** Number of spads on TC4 not yet sent to PTX.
TC5COUNT	15:8	0x0	RO	**TC5 Count** Number of spads on TC5 not yet sent to PTX.
TC6COUNT	23:16	0x0	RO	**TC6 Count** Number of spads on TC6 not yet sent to PTX.
TC7COUNT	31:24	0x0	RO	**TC7 Count** Number of spads on TC7 not yet sent to PTX.

38.39.2.21.48 Protocol Engine RUPM Counters - PRTPE_RUPM_CNTR (0x0000DB20; RO)

Field	Bit(s)	Init.	Type	Description
COUNT	7:0	0x0	RO	**Count** Number of spads not yet sent to PTX.
RESERVED	31:8	0x0	RSV	Reserved.

38.39.2.21.49 Protocol Engine RUPM PTX TC Counters 0-3 - PRTPE_RUPM_PTXTCCNTR03 (0x0000DB40; RO)

Field	Bit(s)	Init.	Type	Description
TC0COUNT	7:0	0x0	RO	**TC0 Count** Number of spads on TC0 sent to PTX.
TC1COUNT	15:8	0x0	RO	**TC1 Count** Number of spads on TC1 sent to PTX.
TC2COUNT	23:16	0x0	RO	**TC2 Count** Number of spads on TC2 sent to PTX.
TC3COUNT	31:24	0x0	RO	**TC3 Count** Number of spads on TC3 sent to PTX.

38.39.2.21.50Protocol Engine RUPM PTX TC Counters 4-7 - PRTPE_RUPM_PTCTCCNTR47 (0x0000DB60; RO)

Field	Bit(s)	Init.	Type	Description
TC4COUNT	7:0	0x0	RO	**TC4 Count** Number of spads on TC4 sent to PTX.
TC5COUNT	15:8	0x0	RO	**TC5 Count** Number of spads on TC5 sent to PTX.
TC6COUNT	23:16	0x0	RO	**TC6 Count** Number of spads on TC6 sent to PTX.
TC7COUNT	31:24	0x0	RO	**TC7 Count** Number of spads on TC7 sent to PTX.

38.39.2.21.51DCB Transmit Command Monitoring Status per TC - PRTDCB_TCMSTC_RLPM[n] (0x0001F040 + 0x20*n, n=0...7; RO)

One register per TC. Register index corresponds to TCID.

Field	Bit(s)	Init.	Type	Description
MSTC	19:0	0x0	RW	**Monitoring Status of the TC** Number of commands that are in transit from the host to the TCB (TCB waiting list included) for TC n, where n is the index of the register in the array.
RESERVED	31:20	0x0	RSV	Reserved.

38.39.2.21.52DCB TC to PFC Mapping - PRTDCB_RLPMC (0x0001F140; RW)

Field	Bit(s)	Init.	Type	Description
TC2PFC	7:0	0x0	RW	**TC to PFC** Bitmap that controls the use of PFC per each TC. Bit n set to: 0b = The device does not issue PFC pause frames with bits set to 1b in the priority_enable_vector for the UPs attached to that TC. It does not react to bits set to 1b for the UPs attached to that TC in the priority_enable_vector of a received PFC pause frame. The TC is referred as a drop UP. 1b = TC n uses PFC in Rx and Tx. The TC is referred as a no-drop TC.
RESERVED	31:8	0x0	RSV	Reserved.

38.39.2.21.53DCB Transmit Command Pipe Monitor Control - PRTDCB_TCPMC_RLPM (0x0001F1A0; RW)

Field	Bit(s)	Init.	Type	Description
CPM	12:0	0x098	RW	**Command Pipe Monitor** Depth of the per Port Monitor applied over the Tx Command Pipe. It is expressed in commands units.
LLTC	20:13	0x0	RW	**Low Latency TC** 0b = TC is bulk. 1b = TC is LL.
RESERVED2	29:21	0x0	RSV	Reserved.

Field	Bit(s)	Init.	Type	Description
TCPM_MODE	30	0b	RW	**Per TC Tx Command Pipe Monitor Mode.** 0b = Conditioned XOFF forwarding mode (default). XOFF notifications received from the TC data pipe monitor are internally forwarded upward to the Tx-scheduler only once the command pipe monitor for this TC is full. This mode provides better XON recovering time. 1b = Immediate XOFF forwarding mode. XOFF notifications received from the TC data pipe monitor are immediately forwarded internally upward to the Tx-scheduler. This mode is useful to support PFC-enabled TCs of the port like true independent TCs when more than two such TCs are configured over the port.
RESERVED1	31	0b	RSV	Reserved.

38.39.2.21.54 Protocol Engine VF CQ Arm - VFPE_CQARM[VF] (0x00130400 + 0x4*VF, VF=0...127; RW)

Note: This register is also located in the PE doorbell page section of the BAR.

Field	Bit(s)	Init.	Type	Description
PECQID	16:0	0x0	RW	**PE Completion Queue ID** Used to arm a PE completion queue for generating events in conjunction with the completion queue doorbell shadow area located in host memory. Arming is also frequently referred to as requesting notification for a completion queue. Events can be generated when the next completion is generated or when the next completion related to a solicited operation is generated. On read, this register returns the value of 0x0.
RESERVED	31:17	0x0	RSV	Reserved.

38.39.2.21.55 Protocol Engine VF CQ ACK - VFPE_CQACK[VF] (0x00130800 + 0x4*VF, VF=0...127; RW)

Field	Bit(s)	Init.	Type	Description
PECQID	16:0	0x0	RW	**PE Completion Queue ID** Used to enable new events for a PE completion queue. The interrupt processing logic that handles completion events must write this register in order to enable new events for a completion queue.
RESERVED	31:17	0x0	RSV	Reserved.

38.39.2.21.56 Protocol Engine VF AEQ Allocate - VFPE_AEQALLOC[VF] (0x00130C00 + 0x4*VF, VF=0...127; RW)

Field	Bit(s)	Init.	Type	Description
AECOUNT	31:0	0x0	RW	**Asynchronous Event Count** Specifies the number of asynchronous event queue entries that have been processed by software and can now be reused by hardware.

38.39.2.21.57 Protocol Engine CQ Arm - PFPE_CQARM (0x00131080; RW)

Note: This register is also located in the PE doorbell page section of the BAR.

Field	Bit(s)	Init.	Type	Description
PECQID	16:0	0x0	RW	**PE Completion Queue ID** Used to arm a PE completion queue for events in conjunction with the completion queue doorbell shadow area located in host memory. Arming is also frequently referred to as requesting notification for a completion queue. Events can be generated when the next completion is generated or when the next completion related to a solicited operation is generated. On read, this register returns the value of 0x0.
RESERVED	31:17	0x0	RSV	Reserved.

38.39.2.21.58 Protocol Engine CQ ACK - PFPE_CQACK (0x00131100; RW)

This register is used to acknowledge a completion event for a PE completion queue. The interrupt processing logic that handles completion events must write this register in order to enable new events for a completion queue.

Field	Bit(s)	Init.	Type	Description
PECQID	16:0	0x0	RW	**PE Completion Queue ID** Used to enable new events for a PE completion queue.
RESERVED	31:17	0x0	RSV	Reserved.

38.39.2.21.59 Protocol Engine AEQ Allocate - PFPE_AEQALLOC (0x00131180; RW)

Field	Bit(s)	Init.	Type	Description
AECOUNT	31:0	0x0	RW	**Asynchronous Event Count** Specifies the number of asynchronous event queue entries that have been processed by software and can now be reused by hardware.

38.39.2.21.60 Protocol Engine CQE Drop Count - GLPE_PFCQEDROPCNT[n] (0x00131340 + 0x4*n, n=0...15; RW1C)

Note: There are 16 of these registers, one per PF.

Field	Bit(s)	Init.	Type	Description
CQEDROPCNT	15:0	0x0	RW1C	**CQE Drop Count** Counts the number of CQEs that are dropped due to the *Valid* bit being cleared in the CQ context for this PF.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.21.61 Protocol Engine CEQE Drop Count - GLPE_PFCEQEDROPCNT[n] (0x001313C0 + 0x4*n, n=0...15; RW1C)

Note: There are 16 of these registers, one per PF.

Field	Bit(s)	Init.	Type	Description
CEQEDROPCNT	15:0	0x0	RW1C	**CEQE Drop Count** Counts the number of CEQEs that are dropped due to the *Valid* bit being cleared in the CEQ Context for this PF.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.21.62 Protocol Engine AEQE Drop Count - GLPE_PFAEQEDROPCNT[n] (0x00131440 + 0x4*n, n=0...15; RW1C)

Note: There are 16 of these registers, one per PF.

Field	Bit(s)	Init.	Type	Description
AEQEDROPCNT	15:0	0x0	RW1C	**AEQE Drop Count** Counts the number of AEQEs that are dropped due to the *Valid* bit being cleared in the AEQ context for this PF.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.21.63 Protocol Engine CQE Drop Count - GLPE_VFCQEDROPCNT[n] (0x00132340 + 0x4*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
CQEDROPCNT	15:0	0x0	RW1C	**CQE Drop Count** Counts the number of CQEs that are dropped due to the *Valid* bit being cleared in the CQ context for this VF.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.21.64 Protocol Engine CEQE Drop Count - GLPE_VFCEQEDROPCNT[n] (0x00132440 + 0x4*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
CEQEDROPCNT	15:0	0x0	RW1C	**CEQE Drop Count** Counts the number of CEQEs that are dropped due to the *Valid* bit being cleared in the CEQ Context for this VF.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.21.65 Protocol Engine AEQE Drop Count - GLPE_VFAEQEDROPCNT[n] (0x00132540 + 0x4*n, n=0...31; RW1C)

Up to 32 PCI VFs can be enabled for PE support. This register is indexed by the HMC VF index.

Field	Bit(s)	Init.	Type	Description
AEQEDROPCNT	15:0	0x0	RW1C	**AEQE Drop Count** Counts the number of AEQEs that are dropped due to the *Valid* bit being cleared in the AEQ Context for this VF.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.21.66 Protocol Engine VF WQE Allocate Register - VFPE_WQEALLOC[VF] (0x00138000 + 0x4*VF, VF=0...127; RW)

This register is used to post the work of the PE queue pairs.

Note: This register is also located in the PE doorbell page section of the BAR.

Field	Bit(s)	Init.	Type	Description
PEQPID	17:0	0x0	RW	**PE Queue Pair ID** This field should be used to notify hardware the new work is available to be processed.
RESERVED	19:18	00b	RSV	Reserved.
WQE_DESC_INDEX	31:20	0x0	RW	**WQE Index** Indicates a high 12 bits of the WQE index in the send queue. This field is used to identify a push mode message in the head of the SQ. Push mode message is processed by hardware only if it is in the head of the send queue; otherwise, it is dropped and processed when WQE is re-fetched during regular transmit operation. This field is used for the push mode doorbells only. For regular doorbells, this field should be treated as reserved, and ignored by hardware.

38.39.2.21.67 Protocol Engine WQE Allocate Register - PFPE_WQEALLOC (0x00138C00; RW)

This register is used to post work the PE queue pairs.

Note: This register is also located in the PE doorbell page section of the BAR.

Field	Bit(s)	Init.	Type	Description
PEQPID	17:0	0x0	RW	**PE Queue Pair ID** This field should be used to notify hardware the new work is available to be processed.
RESERVED	19:18	00b	RSV	Reserved.
WQE_DESC_INDEX	31:20	0x0	RW	**WQE Index** Indicates a high 12 bits of the WQE Index in the send queue. This field is used to identify a push mode message in the head of the SQ. Push mode message is processed by hardware only if it is in the head of the send queue; otherwise, it is dropped and processed when WQE is re-fetched during regular transmit operation. This field is used for the push mode doorbells only. For regular doorbells, this field should be treated as reserved, and ignored by hardware.

38.39.2.22 PF - Manageability Registers

38.39.2.22.1 Firmware Reset Count - GL_FWRESETCNT (0x00083100; RO)

Field	Bit(s)	Init.	Type	Description
FWRESETCNT	31:0	0x0	RO	**Firmware Reset Count** Updated by Hardware. saturates at 0xFFFF,FFFF.

38.39.2.22.2 Flexible TCO Filter Table Registers - Mask - PRT_MNG_FTFT_MASK[n] (0x00085160 + 0x20*n, n=0...7; RO)

Note: The mask field must be 8-byte aligned even if the length field is not 8-byte aligned. Because hardware implementation compares eight bytes at a time, it should get extra masks until the end of the next Qword. Any mask bit that is located after the length should be set to 0x0, indicating no comparison should be done.

Note: In case the actual length defined by the length field register and the mask bits is not 8-byte aligned, there might be a case that a packet that is shorter than the actual required length passes the flexible filter. This can occur due to comparison of up to 7 bytes that come after the packet, but are not a real part of the packet.

Field	Bit(s)	Init.	Type	Description
MASK	15:0	0x0	RW	**Mask** Masks for the filter bytes: PRT_MNG_FTFT_MASK[0] : Mask for bytes 0:15. PRT_MNG_FTFT_MASK[1] : Mask for bytes 16:31. .... PRT_MNG_FTFT_MASK[6] : Mask for bytes 96:111. PRT_MNG_FTFT_MASK[7] : Mask for bytes 112:127. Values are: Ignore. Compare.
RESERVED	31:16	0x0	RSV	Reserved.

38.39.2.22.3 Flexible TCO Filter Table Registers - Length - PRT_MNG_FTFT_LENGTH (0x00085260; RO)

Field	Bit(s)	Init.	Type	Description
LENGTH	7:0	0x0	RW	**Length** Contains the length of the filter defining the number of bytes from the beginning of the packet compared by this filter. If actual packet length is less than length specified by this field, the filter fails. Otherwise, it depends on the result of actual byte comparison. The value should not be greater than 128.
RESERVED	31:8	0x0	RSV	Reserved.

38.39.2.22.4 Flexible TCO Filter Table Registers - Data - PRT_MNG_FTFT_DATA[n] (0x000852A0 + 0x20*n, n=0...31; RO)

The Flexible TCO Filter Table registers (FTFT) contains a 128-byte pattern and a corresponding 128-bit mask array. If enabled, the first 128 bytes of the received packet are compared against the masked bytes in the PRT_MNG_FTFT_DATA registers.

The 128-byte filter is composed of 32 DW entries, accompanied by a 128-bit mask (PRT_MNG_FTFT_MASK, one bit per filter byte). The bytes in each of the Dwords are written in network order (such as, Byte0 written to Bits[7:0], Byte1 to Bits[15:8], and so on). The mask field is set so that Bit[0] in the mask masks Byte0, Bit[1] masks

Byte1, and so on. A value of 1b in the mask field means that the appropriate byte in the filter should be compared to the appropriate byte in the incoming packet.

The PRT_MNG_FTFT_LENGTH register contains a length field defining the number of bytes from the beginning of the packet compared by this filter. If actual packet length is less than length specified by this field, the filter fails. Otherwise, it depends on the result of actual byte comparison. The value should not be greater than 128.

Field	Bit(s)	Init.	Type	Description
DWORD	31:0	0x000000	RW	**Dword** Filter data.

38.39.2.22.5 Hardware Arbitration Control - GL_MNG_HWARB_CTRL (0x000B6130; RO)

Field	Bit(s)	Init.	Type	Description
NCSI_ARB_EN	0	0b	RW	**Hardware Arbitration Enable** If this bit is set, it is assumed the NCSI_ARB_IN and NCSI_ARB_OUT are connected to a hardware arbitration ring. Otherwise, the NCSI_ARB_IN pin is pulled up internally.
RESERVED	31:1	0x0	RSV	Reserved.

38.39.2.22.6 Firmware Semaphore - GL_MNG_FWSM (0x000B6134; RO)

Field	Bit(s)	Init.	Type	Description
FW_MODES	2:0	00b	RW	**Firmware Mode** Indicates in which mode the firmware operates. 000 = Normal mode. 001 = Debug mode. 010 = GLOBR/CORER recovery mode. 011 = Debug and GLOBR/CORER recovery mode. 100 = NVM recovery mode. 101 = Debug and NVM recovery mode.
RESERVED	9:23	0x0	RSV	Reserved.
EEP_RELOAD_IND	10	0b	RW	**NVM Reloaded Indication** Set to 1b after firmware reloads the NVM configuration after a Core reset. Cleared by firmware once the first AQ command is received from one of the drivers.
CRC_ERROR_MODULE	14:11	0x0	RW	**CRC Error Module** Index of (first) module for which a CRC error was found by EMP check. 0x0 = No CRC error found by EMP. ***Note:*** The PE might have found a CRC error in one of the modules it handles. 0x1 = CRC error on EMP Image module. 0x2 = Reserved 0x3 = CRC error on PHY Analog module. 0x4 = CRC error on EMP Global module. 0x5 = CRC error on Manageability module. 0x6 = CRC error on EMP Settings module. All other values are reserved.

Field	Bit(s)	Init.	Type	Description
FW_STATUS_VALID	15	0b	RW	**Firmware Status Valid** (firmware Valid bit) Hardware clears Bits[15:0] in EMP reset de-assertion so software can know firmware status is invalid. Firmware should set this bit to 1b when it is ready (end of init sequence). Whenever setting this bit to 1b, a PFINT_ICR0.*ADMINQ* interrupt must be issued to host (see Section 38.39.2.9.25).
RESERVED	18:16	000b	RW	Reserved.
EXT_ERR_IND	24:19	0x0	RW	**External Error Indication** Firmware writes here the reason that the firmware operation has stopped. For example, NVM CRC error, etc. Possible values: 0x0 = No error. 0x1 = CRC error on a module handled by EMP. Refer to *CRC_ERROR_MODULE* field for details. 0x2 = Switch module failed. 0x3 = Scheduler module failed. 0x4 = DCB module failed. 0x5 = Link module failed. 0x6 = LLDP module failed. 0x7 = Manage module failed. All other values are reserved. ***Note:*** Following error detection and GL_MNG_FWSM.*EXT_IND_ERR* update, the PFINT_ICR0.*ADMINQ* bit is set, and an interrupt is sent to the Host. However, when a value of 0x0 is placed in this field, the PFINT_ICR0.*ADMINQ* bit is not set, and an interrupt is not generated.
RESERVED	25	0b	RSV	Reserved.
PHY_SERDES0_CONFIG_ERR	26	0b	RW	**PHY/SerDes Configuration Error Indication - Port 0** Set by firmware when it fails to configure LAN PHY/SerDes. Cleared by firmware upon successful configuration of LAN PHY/SerDes.
PHY_SERDES1_CONFIG_ERR	27	0b	RW	**PHY/SerDes Configuration Error Indication - Port 1** Set by firmware when it fails to configure LAN PHY/SerDes. Cleared by firmware upon successful configuration of LAN PHY/SerDes.
PHY_SERDES2_CONFIG_ERR	28	0b	RW	**PHY/SerDes Configuration Error Indication - Port 2** Set by firmware when it fails to configure LAN PHY/SerDes. Cleared by firmware upon successful configuration of LAN PHY/SerDes.
PHY_SERDES3_CONFIG_ERR	29	0b	RW	**PHY/SerDes Configuration Error Indication - Port 3** Set by firmware when it fails to configure LAN PHY/SerDes. Cleared by firmware upon successful configuration of LAN PHY/SerDes.
RESERVED	31:30	00b	RSV	Reserved.

38.39.2.22.7 Manageability IPv6 Address Filter - PRT_MNG_MIPAF6[n] (0x00254200 + 0x20*n, n=0...15; RO)

The Manageability IPv6 Address Filter register stores IPv6 addresses for manageability filtering.

Note: These registers should be written in network order.

Field	Bit(s)	Init.	Type	Description
MIPAF	31:0	0x000000	RW	**Manageability IP Address Filters** For each n, m, m=0...3, n=0...3, MIPAF[m,n] register holds Dword `n' of IPv6 filter `m' (4 x IPv6 filters).

38.39.2.22.8 Management Flex UDP/TCP Ports - PRT_MNG_MFUTP[n] (0x00254E00 + 0x20*n, n=0...15; RO)

Each 32-bit register (n=0...15) refers to one UDP/TCP port filter. The MFUTP registers are written by the BMC and are not accessible to the host for writing. The registers are used to filter manageability packets.

The MFUTP registers are cleared on LAN_PWR_GOOD only. The initial values for this register can be loaded from the EEPROM after power-up reset

Note: The MFUTP_N fields should be written in network order.

Field	Bit(s)	Init.	Type	Description
MFUTP_N	15:0	0x0	RW	**Management Flex UDP/TCP Port N** n-th management flex UDP/TCP port.
UDP	16	0b	RW	**UDP** Match if port is UDP.
TCP	17	0b	RW	**TCP** Match if port is TCP.
SOURCE_DESTINATION	18	0b	RW	**Source/Destination** 0b = Compare destination port. 1b = Compare source port.
RESERVED	31:19	0x0	RSV	Reserved.

38.39.2.22.9 Management VLAN TAG Value - PRT_MNG_MAVTV[n] (0x00255900 + 0x20*n, n=0...7; RO)

The MAVTV registers are written by the BMC and are not accessible to the host for writing. The registers are used to filter manageability packets.

Field	Bit(s)	Init.	Type	Description
VID	11:0	0x0	RW	**VLAN ID** Contains the VLAN ID that should be compared with the incoming packet's inner VLAN ID, if the corresponding bit in MDEF is set.
RESERVED	31:12	0x0	RSV	Reserved.

38.39.2.22.10Manageability Decision Filters1 - PRT_MNG_MDEF[n] (0x00255D00 + 0x20*n, n=0...7; RO)

Field	Bit(s)	Init.	Type	Description
MAC_EXACT_AND	3:0	0x0	RW	**MAC Exact AND** Controls the inclusion of exact MAC address 0 to 3 in the manageability filter decision (AND section). Bit[0] corresponds to exact MAC address 0 (MMAL0 and MMAH0), etc.
BROADCAST_AND	4	0b	RW	**Broadcast AND** Controls the inclusion of broadcast address filtering in the manageability filter decision (AND section).
VLAN_AND	12:5	0x0	RW	**VLAN AND** Controls the inclusion of VLAN tag 0 to 7, respectively, in the manageability filter decision (AND section). Bit[5] corresponds to VLAN tag 0, etc.
IPV4_ADDRESS_AND	16:13	0x0	RW	**IPv4 Address AND** Controls the inclusion of IPV4 address 0 to 3, respectively, in the manageability filter decision (AND section). Bit[13] corresponds to IPV4 address 0, etc. ***Note:*** These bits are set also for an ARP request packet if the Target IP match the IP address configured in the MIPAF register
IPV6_ADDRESS_AND	20:17	0x0	RW	**IPv6 Address AND** Controls the inclusion of IPV6 address 0 to 3, respectively, in the manageability filter decision (AND section). Bit[17] corresponds to IPV6 address 0, etc
MAC_EXACT_OR	24:21	0x0	RW	**MAC Exact OR** Controls the inclusion of exact MAC address 0 to 3 in the manageability filter decision (OR section). Bit[21] corresponds to exact MAC address 0 (MMAL0 and MMAH0), etc.
BROADCAST_OR	25	0b	RW	**Broadcast OR** Controls the inclusion of broadcast address filtering in the manageability filter decision (OR section).
MULTICAST_AND	26	0b	RW	**Multicast AND** Controls the inclusion of multicast address filtering in the manageability filter decision (AND section). Broadcast packets are not included by this bit.
ARP_REQUEST_OR	27	0b	RW	**ARP Request OR** Controls the inclusion of ARP request filtering in the manageability filter decision (OR section).
ARP_RESPONSE_OR	28	0b	RW	**ARP Response OR** Controls the inclusion of ARP response filtering in the manageability filter decision (OR section).
NEIGHBOR_DISCOVERY_134_OR	29	0b	RW	**Neighbor Discovery 134 OR** Controls the inclusion of neighbor discovery filtering in the manageability filter decision (OR section). The neighbor type accepted by this filter is type 0x86 (134).
PORT_0X298_OR	30	0b	RW	**Port 0x298 OR** Controls the inclusion of port 0x298 filtering in the manageability filter decision (OR section).
PORT_0X26F_OR	31	0b	RW	**Port 0x26F OR** Controls the inclusion of port 0x26F filtering in the manageability filter decision (OR section).

38.39.2.22.11Manageability Decision Filters - PRT_MNG_MDEF_EXT[n] (0x00255F00 + 0x20*n, n=0...7; RO)

Field	Bit(s)	Init.	Type	Description
L2_ETHERTYPE_AND	3:0	0x0	RW	**L2 EtherType AND** Controls the inclusion of L2 EtherType filtering in the manageability filter decision (AND section).
L2_ETHERTYPE_OR	7:4	0x0	RW	**L2 EtherType OR** Controls the inclusion of L2 EtherType filtering in the manageability filter decision (OR section).
FLEX_PORT_OR	23:8	0x0	RW	**Flex Port OR** Controls the inclusion of flex port filtering in the manageability filter decision (OR section). Bit[16] corresponds to flex port 0, etc.
FLEX_TCO	24	0b	RW	**Flex TCO** Controls the inclusion of flex TCO filtering in the manageability filter decision (OR section). Bit[24] corresponds to Flex TCO filter. ***Note:*** Supported only for Network traffic.
NEIGHBOR_DISCOVERY_135_OR	25	0b	RW	**Neighbor Discovery 135 OR** Controls the inclusion of neighbor discovery filtering in the manageability filter decision (OR section). The neighbor type accepted by this filter is type 0x87 (135).
NEIGHBOR_DISCOVERY_136_OR	26	0b	RW	**Neighbor Discovery 136 OR** Controls the inclusion of neighbor discovery filtering in the manageability filter decision (OR section). The neighbor type accepted by this filter is type 0x88 (136).
NEIGHBOR_DISCOVERY_137_OR	27	0b	RW	**Neighbor Discovery 137 OR** Controls the inclusion of neighbor discovery filtering in the manageability filter decision (OR section). The neighbor type accepted by this filter is type 0x89 (137).
ICMP_OR	28	0b	RW	**ICMP OR** Controls the inclusion of ICMP filtering in the manageability filter decision (OR section).
MLD	29	0b	RW	**MLD OR** Control the inclusion of MLD packets. These are ICMPv6 packets with the following types: 130, 131, 132, 143.
APPLY_TO_NETWORK_TRAFFIC	30	0b	RW	**Apply to Network Traffic** 0b =This decision filter does not apply to traffic received from the network. 1b =This decision filter applies to traffic received from the network.
APPLY_TO_HOST_TRAFFIC	31	0b	RW	**Apply to Host Traffic** 0b = This decision filter does not apply to traffic received from the host. 1b = This decision filter applies to traffic received from the host.

38.39.2.22.12 Manageability IPv4 Address Filter - PRT_MNG_MIPAF4[n] (0x00256280 + 0x20*n, n=0...3; RO)

The Manageability IPv4 Address Filter register stores IPv4 addresses for manageability filtering.

Note: These registers should be written in network order.

Field	Bit(s)	Init.	Type	Description
MIPAF	31:0	0x000000	RW	**Manageability IP Address Filters** For each n, m, m=0...3, n=0...3, MIPAF[m,n] register holds Dword `n' of IPv4 filter `m' (4 x IPv4 filters).

38.39.2.22.13 Manageability MAC Address High - PRT_MNG_MMAH[n] (0x00256380 + 0x20*n, n=0...3; RO)

These registers contain the upper 16 bits of the 48-bit Ethernet address. The complete address is (MMAH, MMAL). The MMAH registers are written by the BMC and are not accessible to the host for writing. The registers are used to filter manageability packets.

The initial values for this register can be loaded from the NVM after power-up reset or firmware reset.

Note: The *MMAH* field should be written in network order.

Field	Bit(s)	Init.	Type	Description
MMAH	15:0	0x000000	RW	**Manageability MAC Address High** The upper 16 bits of the 48-bit Ethernet address. Appears in big endian order (MS byte of MMAH is last on the wire).
RESERVED	31:16	0x0	RSV	Reserved. Reads as 0x0. Ignored on write.

38.39.2.22.14 Manageability MAC Address Low - PRT_MNG_MMAL[n] (0x00256480 + 0x20*n, n=0...3; RO)

These registers contain the lower 32 bits of the 48-bit Ethernet address. The MMAL registers are written by internal firmware and are not accessible to the host for writing. The registers are used to filter manageability packets.

The MMAL registers are cleared on LAN_PWR_GOOD only. The initial values for this register can be loaded from the NVM after power-up reset.

Note: The *MMAL* field should be written in network order.

Field	Bit(s)	Init.	Type	Description
MMAL	31:0	0x000000	RW	**Manageability MAC Address Low** The lower 32 bits of the 48-bit Ethernet address. Appears in big endian order (LS byte of MMAL is first on the wire).

38.39.2.22.15 Management Decision Filters Buffers - PRT_MNG_MDEFVSI[n] (0x00256580 + 0x20*n, n=0...3; RO)

This register is used to define the VSIs used to receive packets that matched a specific MDEF. In case of multiple match the VSI assigned to the MDEF with the highest index is used.

Field	Bit(s)	Init.	Type	Description
MDEFVSI_2N	15:0	0x0	RW	**Management Decision Filter VSI 2n** Defines the VSI used for packets matching MDEF 2*n.
MDEFVSI_2NP1	31:16	0x0	RW	**Management Decision Filter VSI 2n+1** Defines the VSI used for packets matching MDEF 2*n+1.

38.39.2.22.16 Management Ethernet Type Filters - PRT_MNG_METF[n] (0x00256780 + 0x20*n, n=0...3; RO)

The METF registers are written by the BMC and are not accessible to the host for writing. The registers are used to filter manageability packets.

The METF registers are cleared on LAN_PWR_GOOD only. The initial values for this register might be loaded from the EEPROM after power-up reset.

Field	Bit(s)	Init.	Type	Description
ETYPE	15:0	0x0	RW	**EtherType** Value to be compared against the *L2 EtherType* field in the Rx packet. Appears in little endian order (high byte first on the wire).
RESERVED	29:16	0x0	RSV	Reserved.
POLARITY	30	0b	RW	**Polarity** 0b = Positive filter. Filter enters the decision filters if a match occurred. 1b = Negative filter. Filter enters the decision filters if a match did not occur.
RESERVED	31	0b	RSV	Reserved.

38.39.2.22.17 Management Control Register - PRT_MNG_MANC (0x00256A20; RO)

The MANC register can be written by the BMC, and is not accessible to the host for writing.

Field	Bit(s)	Init.	Type	Description
FLOW_CONTROL_DISCARD	0	0b	RW	**Flow Control Discard** 0b = Apply filtering rules to packets with Flow control EtherType. 1b = Discard packets with Flow control EtherType. ***Note:*** Flow control EtherType is 0x8808.
NCSI_DISCARD	1	0b	RW	**NC-SI Discard** 0b = Apply filtering rules to packets with NC-SI EtherType. 1b = Discard packets with NC-SI EtherType. ***Note:*** NC-SI EtherType is 0x88F8.
RESERVED	16:2	0x0	RSV	Reserved.
RCV_TCO_EN	17	0b	RW	**Receive TCO Packets Enabled** When this bit is set, it enables the receive flow to the manageability block. This bit should be set only if at least one of MANC.*EN_BMC2OS* or MANC.*EN_BMC2NET* bits are set.
RESERVED	24:18	0x0	RSV	Reserved.

Field	Bit(s)	Init.	Type	Description
FIXED_NET_TYPE	25	0b	RW	**Fixed Next Type** If set, only packets matching the net type defined by the *NET_TYPE* field are passed to manageability. Otherwise, both tagged and un-tagged packets may be forwarded to manageability engine.
NET_TYPE	26	0b	RW	**Net Type** 0b = Pass only un-tagged packets. 1b = Pass only VLAN tagged packets. Valid only if *FIXED_NET_TYPE* is set.
RESERVED	27	0b	RSV	Reserved.
EN_BMC2OS	28	0b	RW	**Enable BMC to OS and OS to BMC Traffic** 0b = The BMC can not communicate with the operating system. 1b = The BMC can communicate with the operating system. When cleared the BMC traffic is not forwarded to the operating system, even if the host address filtering indicates that it should. When cleared, operating system traffic is not forwarded to the BMC even if the manageability decision filters indicates it should. This bit does not impact the BMC to network traffic. ***Note:*** Initial value loaded according to value of port n traffic types field in NVM.
EN_BMC2NET	29	0b	RW	**Enable BMC to network and network to BMC traffic** 0b = The BMC can not communicate with the network. 1b = The BMC can communicate with the network When cleared, the BMC traffic is not forwarded to the network and the network traffic is not forwarded to the BMC even if the decision filters indicates it should. This bit does not impact the host to BMC traffic. ***Note:*** Initial value loaded according to value of port n traffic types field in NVM.
RESERVED	31:30	00b	RSV	Reserved.

38.39.2.22.18Management Only Traffic Register - PRT_MNG_MNGONLY (0x00256A60; RO)

The MNGONLY register enables exclusive filtering of certain types of traffic to the BMC. Exclusive filtering enables the BMC to define certain packets that are forwarded to the BMC but not to the host. The packets are not forwarded to the host even if they pass the host L2 filtering process.

Each manageability decision filter (MDEF and MDEF_EXT) has a corresponding bit in the MNGONLY register. When a manageability decision filter (MDEF and MDEF_EXT) forwards a packet to manageability, it might also block the packet from being forwarded to the host if the corresponding *MNGONLY* bit is set.

Field	Bit(s)	Init.	Type	Description
EXCLUSIVE_TO_MANAGEABILITY	7:0	0x0	RW	**Exclusive to MNG** When set, indicates that packets forwarded by the manageability filters to manageability are not sent to the host. Bits[0...7] correspond to decision rules defined in registers MDEF[0...7] and MDEF_EXT[0...7].
RESERVED	31:8	0x0	RSV	Reserved.

38.39.2.22.19Manageability Special Filters Modifiers - PRT_MNG_MSFM (0x00256AA0; RO)

Field	Bit(s)	Init.	Type	Description
PORT_26F_UDP	0	1b	RW	**Port 0x26F UDP** Port 0x26F match if protocol is UDP.
PORT_26F_TCP	1	1b	RW	**Port 0x26F TCP** Port 0x26F match if protocol is TCP.
PORT_298_UDP	2	1b	RW	**Port 0x298 UDP** Port 0x298 match if protocol is UDP.
PORT_298_TCP	3	1b	RW	**Port 0x298 TCP** Port 0x298 match if protocol is TCP.
IPV6_0_MASK	4	0b	RW	**IPv6 Address 0 Mask** Compare only 24 LSB bits of IPv6 Address 0 (MIPAF[0]).
IPV6_1_MASK	5	0b	RW	**IPv6 Address 1 Mask** Compare only 24 LSB bits of IPv6 Address 1 (MIPAF[1]).
IPV6_2_MASK	6	0b	RW	**IPv6 Address 2 Mask** Compare only 24 LSB bits of IPv6 Address 2 (MIPAF[2]).
IPV6_3_MASK	7	0b	RW	**IPv6 Address 3 Mask** Compare only 24 LSB bits of IPv6 Address 3 (MIPAF[3]).
RESERVED	31:8	0x0	RSV	Reserved.

38.39.3 BAR3 Registers Summary

Table 38-610.BAR3 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
PF - MSI-X Table Registers			
0x00000000 + 0x10*n, n=0...128	MSIX_TADD[n]	MSI-X Message Address Low	3681
0x00000004 + 0x10*n, n=0...128	MSIX_TUADD[n]	MSI-X Message Address High	3681
0x00000008 + 0x10*n, n=0...128	MSIX_TMSG[n]	MSI-X Message Data	3681
0x0000000C + 0x10*n, n=0...128	MSIX_TVCTRL[n]	MSI-X Vector Control	3682
0x00001000 + 0x4*n, n=0...5	MSIX_PBA[n]	MSI-X PBA Structure	3682
0x00002000 + 0x4*n, n=0...19	VFMSIX_PBA[n]	VF MSI-X PBA Structure	3682
0x00002100 + 0x10*n, n=0...639	VFMSIX_TADD[n]	VF MSI-X Message Address Low	3682
0x00002104 + 0x10*n, n=0...639	VFMSIX_TUADD[n]	VF MSI-X Message Address High	3683
0x00002108 + 0x10*n, n=0...639	VFMSIX_TMSG[n]	VF MSI-X Message Data	3683
0x0000210C + 0x10*n, n=0...639	VFMSIX_TVCTRL[n]	VF MSI-X Vector Control	3683

38.39.4 Detailed Register Description - PF BAR3

38.39.4.1 PF - MSI-X Table Registers

This category contains registers in the separate MSI-X BAR.

38.39.4.1.1 MSI-X Message Address Low - MSIX_TADD[n] (0x00000000 + 0x10*n, n=0...128; RW)

Message lower address for MSI-X table entries.

Field	Bit(s)	Init.	Type	Description
MSIXTADD10	1:0	0x0	RW	**Message Address 1:0** For proper Dword alignment, software must always write zeros to these two bits; otherwise, the result is undefined. The state of these bits after reset must be 0b. These bits are permitted to be read-only or read/write.
MSIXTADD	31:2	0x0	RW	**Message Address** System-specified message lower address. For MSI-X messages, the contents of this field from an MSI-X table entry specifies the lower portion of the Dword-aligned address (AD[31:02]) for the memory write transaction. This field is read/write.

38.39.4.1.2 MSI-X Message Address High - MSIX_TUADD[n] (0x00000004 + 0x10*n, n=0...128; RW)

Message upper address for MSI-X table entries.

Field	Bit(s)	Init.	Type	Description
MSIXTUADD	31:0	0x0	RW	**Message Upper Address** System-specified message upper address bits. If this field is zero, Single Address Cycle (SAC) messages are used. If this field is non-zero, Dual Address Cycle (DAC) messages are used. This field is read/write.

38.39.4.1.3 MSI-X Message Data - MSIX_TMSG[n] (0x00000008 + 0x10*n, n=0...128; RW)

Message data for MSI-X table entries.

Field	Bit(s)	Init.	Type	Description
MSIXTMSG	31:0	0x0	RW	**Message Data** System-specified message data. For MSI-X messages, the contents of this field from an MSI-X table entry specifies the data driven on AD[31:0] during the memory write transaction's data phase. This field is read/write.

38.39.4.1.4 MSI-X Vector Control - MSIX_TVCTRL[n] (0x0000000C + 0x10*n, n=0...128; RW)

Vector control for MSI-X table entries.

Field	Bit(s)	Init.	Type	Description
MASK	0	1b	RW	**Mask Bit** When this bit is set, the function is prohibited from sending a message using this MSI-X table entry. However, any other MSI-X table entries programmed with the same vector are still capable of sending an equivalent message unless they are also masked. This bit's state after reset is 1b (entry is masked).
RESERVED	31:1	0x0	RSV	Reserved. After reset, the state of these bits must be 0b. However, for potential future use, software must preserve the value of these reserved bits when modifying the value of other *Vector Control* bits. If software modifies the value of these reserved bits, the result is undefined.

38.39.4.1.5 MSI-X PBA Structure - MSIX_PBA[n] (0x00001000 + 0x4*n, n=0...5; RO)

Pending bits for MSI-X PBA entries.

Field	Bit(s)	Init.	Type	Description
PENBIT	31:0	0x0	RO	**MSI-X Pending Bits** Each bit is set to 1b when the appropriate interrupt request is set, and cleared to 0b when the appropriate interrupt request is cleared.

38.39.4.1.6 VF MSI-X PBA Structure - VFMSIX_PBA[n] (0x00002000 + 0x4*n, n=0...19; RO)

Pending bits for MSI-X PBA entries.

Field	Bit(s)	Init.	Type	Description
PENBIT	31:0	0x0	RO	**MSI-X Pending Bits** Each bit is set to 1b when the appropriate interrupt request is set, and cleared to 0b when the appropriate interrupt request is cleared.

38.39.4.1.7 VF MSI-X Message Address Low - VFMSIX_TADD[n] (0x00002100 + 0x10*n, n=0...639; RW)

Message lower address for MSI-X table entries.

Field	Bit(s)	Init.	Type	Description
MSIXTADD10	1:0	0x0	RW	**Message Address 1:0** For proper Dword alignment, software must always write zeros to these two bits; otherwise, the result is undefined. The state of these bits after reset must be 0b. These bits are permitted to be read-only or read/write.
MSIXTADD	31:2	0x0	RW	**Message Address** System-specified message lower address. For MSI-X messages, the contents of this field from an MSI-X table entry specifies the lower portion of the Dword-aligned address (AD[31:02]) for the memory write transaction. This field is read/write.

38.39.4.1.8 VF MSI-X Message Address High - VFMSIX_TUADD[n] (0x00002104 + 0x10*n, n=0...639; RW)

Message upper address for MSI-X table entries.

Field	Bit(s)	Init.	Type	Description
MSIXTUADD	31:0	0x0	RW	**Message Upper Address** System-specified message upper address bits. If this field is zero, SAC messages are used. If this field is non-zero, DAC messages are used. This field is read/write.

38.39.4.1.9 VF MSI-X Message Data - VFMSIX_TMSG[n] (0x00002108 + 0x10*n, n=0...639; RW)

Message data for MSI-X table entries.

Field	Bit(s)	Init.	Type	Description
MSIXTMSG	31:0	0x0	RW	**Message Data** System-specified message data. For MSI-X messages, the contents of this field from an MSI-X table entry specifies the data driven on AD[31:0] during the memory write transaction's data phase. This field is read/write.

38.39.4.1.10 VF MSI-X Vector Control - VFMSIX_TVCTRL[n] (0x0000210C + 0x10*n, n=0...639; RW)

Vector control for MSI-X table entries.

Field	Bit(s)	Init.	Type	Description
MASK	0	1b	RW	**Mask Bit** When this bit is set, the function is prohibited from sending a message using this MSI-X table entry. However, any other MSI-X table entries programmed with the same vector are still capable of sending an equivalent message unless they are also masked. This bit's state after reset is 1b (entry is masked).
RESERVED	31:1	0x0	RSV	Reserved. After reset, the state of these bits must be 0b. However, for potential future use, software must preserve the value of these reserved bits when modifying the value of other *Vector Control* bits. If software modifies the value of these reserved bits, the result is undefined.

38.40 Device Registers - VF

38.40.1 VF Registers Mapping in the PF Space

Table 38-611.VF Registers Summary

Abbreviation	Virtual Address	Physical Address
VFMSIX_TADD	0x00000000 + 0x10*n, n=0...16	0x00002100 + 0x10*n, n=0...639
QTX_TAIL	0x00000000 + 0x4*Q, Q=0...15	0x00108000 + 0x4*Q, Q=0...1535
VFMSIX_TUADD	0x00000004 + 0x10*n, n=0...16	0x00002104 + 0x10*n, n=0...639
VFMSIX_TMSG	0x00000008 + 0x10*n, n=0...16	0x00002108 + 0x10*n, n=0...639
VFMSIX_TVCTRL	0x0000000C + 0x10*n, n=0...16	0x0000210C + 0x10*n, n=0...639

Table 38-611.VF Registers Summary

Abbreviation	Virtual Address	Physical Address
VFMSIX_PBA	0x00002000	0x00002000 + 0x4*n, n=0...19
QRX_TAIL	0x00002000 + 0x4*Q, Q=0...15	0x00128000 + 0x4*Q, Q=0...1535
VFINT_ITRN	0x00002800 + 0x40*n + 0x4*INTVF, n=0...2, INTVF=0...15	0x00020000 + 0x800*n + 0x4*INTVF, n=0...2, INTVF=0...511
VFINT_DYN_CTLN	0x00003800 + 0x4*INTVF, INTVF=0...15	0x00024800 + 0x4*INTVF, INTVF=0...511
VFINT_ICR0	0x00004800	0x0002BC00 + 0x4*VF, VF=0...127
VFINT_ITR0	0x00004C00 + 0x4*n, n=0...2	0x00028000 + 0x400*n + 0x4*VF, n=0...2, VF=0...127
VFINT_ICR0_ENA	0x00005000	0x0002C000 + 0x4*VF, VF=0...127
VFINT_STAT_CTL0	0x00005400	0x0002A000 + 0x4*VF, VF=0...127
VFINT_DYN_CTL0	0x00005C00	0x0002A400 + 0x4*VF, VF=0...127
VF_ARQBAH	0x00006000	0x00081400 + 0x4*VF, VF=0...127
VF_ATQH	0x00006400	0x00082000 + 0x4*VF, VF=0...127
VF_ATQLEN	0x00006800	0x00081800 + 0x4*VF, VF=0...127
VF_ARQBAL	0x00006C00	0x00080C00 + 0x4*VF, VF=0...127
VF_ARQT	0x00007000	0x00082C00 + 0x4*VF, VF=0...127
VF_ARQH	0x00007400	0x00082400 + 0x4*VF, VF=0...127
VF_ATQBAH	0x00007800	0x00081000 + 0x4*VF, VF=0...127
VF_ATQBAL	0x00007C00	0x00080800 + 0x4*VF, VF=0...127
VF_ARQLEN	0x00008000	0x00081C00 + 0x4*VF, VF=0...127
VF_ATQT	0x00008400	0x00082800 + 0x4*VF, VF=0...127
VFGEN_RSTAT	0x00008800	0x00074400 + 0x4*VF, VF=0...127
VFPE_IPCONFIG0	0x00008C00	0x00001400 + 0x4*VF, VF=0...127
VFPE_MRTEIDXMASK	0x00009000	0x00003000 + 0x4*VF, VF=0...127
VFPE_RCVUNEXPECTEDERROR	0x00009400	0x00003400 + 0x4*VF, VF=0...127
VFPE_CCQPHIGH	0x00009800	0x00001000 + 0x4*VF, VF=0...127
VFPE_CQPERRCODES	0x00009C00	0x00001800 + 0x4*VF, VF=0...127
VFPE_CQPTAIL	0x0000A000	0x00000400 + 0x4*VF, VF=0...127
VFPE_AEQALLOC	0x0000A400	0x00130C00 + 0x4*VF, VF=0...127
VFPE_TCPNOWTIMER	0x0000A800	0x00002C00 + 0x4*VF, VF=0...127
VFPE_CCQPLOW	0x0000AC00	0x00000C00 + 0x4*VF, VF=0...127
VFPE_CQACK	0x0000B000	0x00130800 + 0x4*VF, VF=0...127
VFPE_CQARM	0x0000B400	0x00130400 + 0x4*VF, VF=0...127
VFPE_CCQPSTATUS	0x0000B800	0x00000800 + 0x4*VF, VF=0...127
VFPE_CQPDB	0x0000BC00	0x00000000 + 0x4*VF, VF=0...127
VFPE_WQEALLOC	0x0000C000	0x00138000 + 0x4*VF, VF=0...127
VFQF_HENA	0x0000C400 + 0x4*n, n=0...1	0x00230800 + 0x400*n + 0x4*VF, n=0...1, VF=0...127
VFQF_HREGION	0x0000D400 + 0x4*n, n=0...7	0x0022E000 + 0x400*n + 0x4*VF, n=0...7, VF=0...127
PFPCI_VF_FLUSH_DONE	0x0000E400	0x0009C600 + 0x4*VF, VF=0...127

38.40.2 BAR0 Registers Summary

Table 38-612.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
PF - PCIe Registers			
0x0000E400	PFPCI_VF_FLUSH_DONE	PCIe VF Flush Done	3686
VF - General Registers			
0x00008800	VFGEN_RSTAT	VF Reset Status	3686
VF - Interrupt Registers			
0x00002800 + 0x40*n + 0x4*INTVF, n=0...2, INTVF=0...15	VFINT_ITRN[n,INTVF]	VF Interrupt Throttling for Interrupt N	3687
0x00003800 + 0x4*INTVF, INTVF=0...15	VFINT_DYN_CTLN[INTVF]	VF Interrupt N Dynamic Control	3687
0x00004800	VFINT_ICR0	VF Interrupt Zero Cause	3688
0x00004C00 + 0x4*n, n=0...2	VFINT_ITR0[n]	VF Interrupt Throttling for Interrupt Zero	3688
0x00005000	VFINT_ICR0_ENA	VF Interrupt Zero Cause Enablement	3689
0x00005400	VFINT_STAT_CTL0	VF Interrupt Zero Static Control	3689
0x00005C00	VFINT_DYN_CTL0	VF Interrupt Zero Dynamic Control	3689
VF - Admin Queue			
0x00006000	VF_ARQBAH	VF Admin Receive Queue Base Address High	3690
0x00006400	VF_ATQH	VF Admin Transmit Head	3690
0x00006800	VF_ATQLEN	VF Admin Transmit Queue Length	3690
0x00006C00	VF_ARQBAL	VF Admin Receive Queue Base Address Low	3691
0x00007000	VF_ARQT	VF Admin Receive Queue Tail	3691
0x00007400	VF_ARQH	VF Admin Receive Queue Head	3691
0x00007800	VF_ATQBAH	VF Admin Transmit Queue Base Address High	3691
0x00007C00	VF_ATQBAL	VF Admin Transmit Queue Base Address Low	3691
0x00008000	VF_ARQLEN	VF Admin Receive Queue Length	3692
0x00008400	VF_ATQT	VF Admin Transmit Tail	3692
VF - LAN Transmit Receive Registers			
0x00000000 + 0x4*Q, Q=0...15	QTX_TAIL[Q]	Global Transmit Queue Tail	3692
0x00002000 + 0x4*Q, Q=0...15	QRX_TAIL[Q]	Global Receive Queue Tail	3692
VF - Rx Filters Registers			
0x0000C400 + 0x4*n, n=0...1	VFQF_HENA[n]	VF Queue Filter Hash Enabled Packet Type	3693
0x0000D400 + 0x4*n, n=0...7	VFQF_HREGION[n]	VF Queue Filter Hash Region of Queues	3693
VF - Protocol Engine Registers			
0x00008C00	VFPE_IPCONFIG0	Protocol Engine VF IP Config 0	3694
0x00009000	VFPE_MRTEIDXMASK	Protocol Engine VF MRTE Index Mask	3694
0x00009400	VFPE_RCVUNEXPECTEDERROR	Protocol Engine VF Unexpected Error	3694

Table 38-612.BAR0 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
0x00009800	VFPE_CCQPHIGH	Protocol Engine VF Create CQP High	3694
0x00009C00	VFPE_CQPERRCODES	Protocol Engine CQP Error Codes	3695
0x0000A000	VFPE_CQPTAIL	Protocol Engine CQP Tail	3695
0x0000A400	VFPE_AEQALLOC	Protocol Engine VF AEQ Allocate	3695
0x0000A800	VFPE_TCPNOWTIMER	Protocol Engine VF TCP Now Timer	3695
0x0000AC00	VFPE_CCQPLOW	Protocol Engine VF Create CQP Low	3696
0x0000B000	VFPE_CQACK	Protocol Engine VF CQ ACK	3696
0x0000B400	VFPE_CQARM	Protocol Engine VF CQ Arm	3696
0x0000B800	VFPE_CCQPSTATUS	Protocol Engine VF Create CQP Status	3696
0x0000BC00	VFPE_CQPDB	Protocol Engine VF CQP Doorbell	3697
0x0000C000	VFPE_WQEALLOC	Protocol Engine VF WQE Allocate Register	3697

38.40.3 Detailed Register Description - VF BAR0

38.40.3.1 PF - PCIe Registers

38.40.3.1.1 PCIe VF Flush Done - PFPCI_VF_FLUSH_DONE (0x0000E400; RO)

Field	Bit(s)	Init.	Type	Description
FLUSH_DONE	0	0b	RO	**Flush Done**
RESERVED	31:1	0x0	RSV	Reserved.

38.40.3.2 VF - General Registers

This section describes the registers allocated to a VF for generic control and status. These registers are tied to the VF and are not dependent of any resource allocation.

38.40.3.2.1 VF Reset Status - VFGEN_RSTAT (0x00008800; RW)

Field	Bit(s)	Init.	Type	Description
VFR_STATE	1:0	00b	RW	**VFR State** Defines the VFR reset progress as follows: 00b = VFR in progress. 01b = VFR completed. 10b = Reserved. 11b = Reserved. This field is used to communicate the reset progress to the VF with no impact on hardware functionality.
RESERVED	31:2	0x0	RSV	Reserved.

38.40.3.3 VF - Interrupt Registers

38.40.3.3.1 VF Interrupt Throttling for Interrupt N - VFINT_ITRN[n,INTVF] (0x00002800 + 0x40*n + 0x4*INTVF, n=0…2, INTVF=0…15; RW)

Register index 'n' relates to interrupt 'n+1', while interrupt zero is controlled by the VFINT_ITR0 register (Section 38.39.2.9.6).

Field	Bit(s)	Init.	Type	Description
INTERVAL	11:0	0x0	RW	**Interval** ITR 'n' interval, where 'n' is the register index = 0,1,2 for the three ITRs per interrupt. It is defined in 2 us units, enabling interval range from zero to 8160 us (0xFF0). Setting the INTERVAL to zero enables immediate interrupt. This register can also be programmed by setting the *INTERVAL* field in the matched xxINT_DYN_CTLx register.
RESERVED	31:12	0x0	RSV	Reserved.

38.40.3.3.2 VF Interrupt N Dynamic Control - VFINT_DYN_CTLN[INTVF] (0x00003800 + 0x4*INTVF, INTVF=0…15; RW)

Register index 'n' relates to interrupt 'n+1', while interrupt zero is controlled by the VFINT_DYN_CTL0 register (Section 38.39.2.9.8).

Field	Bit(s)	Init.	Type	Description
INTENA	0	0b	RW	**Interrupt Enable** 0b = Interrupt disabled. 1b = Interrupt enabled. Refer to auto-clear policy in the Interrupt Enablement section. This bit is meaningful only if the *INTENA_MSK* flag in this register is not set.
CLEARPBA	1	0b	RW1C	**Clear PBA** Setting this bit clears the matched PBA bit. This bit is auto-cleared by hardware.
SWINT_TRIG	2	0b	RW1C	**Software Interrupt Trigger** When this bit is set, a software interrupt is triggered. This bit is auto-cleared by hardware.
ITR_INDX	4:3	00b	RW1C	**ITR Index** Defines the ITR index to be updated, as follows: 00b = ITR0. 01b = ITR1. 10b = ITR2. 11b = No ITR update. This field is auto-cleared by hardware.
INTERVAL	16:5	0x0	RW1C	**Interval** The interval for the ITR defined by the *ITR_INDX* field in this register. This field is auto-cleared by hardware.
RESERVED	23:17	0x0	RSV	Reserved.
SW_ITR_INDX_ENA	24	0b	RW1C	**Software ITR Index Enable** Enables the programming of the *SW_ITR_INDX* field in this register. This flag is auto-cleared by hardware.

Field	Bit(s)	Init.	Type	Description
SW_ITR_INDX	26:25	00b	RW	**Software ITR Index** ITR index of the software interrupt: 00b = ITR0. 01b = ITR1. 10b = ITR2. 11b = No ITR. When programming this field, the *SW_ITR_INDX_ENA* flag in this register should be set as well.
RESERVED	29:27	000b	RSV	Reserved.
WB_ON_ITR	30	0b	RW	When this bit is set, completed descriptors are indicated to host memory on ITR completion (or no ITR) regardless of the interrupt enablement in this register.
INTENA_MSK	31	0b	RW1C	**Interrupt Enable Mask** When this bit is set, the *INTENA* setting does not impact the device setting. This bit is auto-cleared by hardware.

38.40.3.3.3 VF Interrupt Zero Cause - VFINT_ICR0 (0x00004800; RCW)

Field	Bit(s)	Init.	Type	Description
INTEVENT	0	0b	RCW	**Interrupt Event indication** This bit is set on assertion of any causes for this interrupt, and cleared when the interrupt is asserted to the rate limit logic.
QUEUE_0	1	0b	RCW	**Queue 0** Queue 0 interrupt for LAN transmit and receive queues and PE CEQs.
QUEUE_1	2	0b	RCW	**Queue 1** Queue 1 interrupt for LAN transmit and receive queues and PE CEQs.
QUEUE_2	3	0b	RCW	**Queue 2** Queue 2 interrupt for LAN transmit and receive queues and PE CEQs.
QUEUE_3	4	0b	RCW	**Queue 3** Queue 3 interrupt for LAN transmit and receive queues and PE CEQs.
RESERVED	29:5	0x0	RSV	Reserved.
ADMINQ	30	0b	RCW	**Admin Queue** Send/receive admin queue interrupt indication.
SWINT	31	0b	RCW	**Software Interrupt indication**

38.40.3.3.4 VF Interrupt Throttling for Interrupt Zero - VFINT_ITR0[n] (0x00004C00 + 0x4*n, n=0...2; RW)

Field	Bit(s)	Init.	Type	Description
INTERVAL	11:0	0x0	RW	**Interval** ITR 'n' interval, where 'n' is the register index = 0,1,2 for the three ITRs per interrupt. It is defined in 2 us units, enabling interval range from zero to 8160 us (0xFF0). Setting the INTERVAL to zero enables immediate interrupt. This register can also be programmed by setting the *INTERVAL* field in the matched xxINT_DYN_CTLx register.
RESERVED	31:12	0x0	RSV	Reserved.

38.40.3.3.5 VF Interrupt Zero Cause Enablement - VFINT_ICR0_ENA (0x00005000; RW)

Field	Bit(s)	Init.	Type	Description
RESERVED	29:0	0x0	RSV	Reserved.
ADMINQ	30	0b	RW	**Admin Queue** Enable this interrupt at 1b.
RESERVED	31	0b	RW	Reserved.

38.40.3.3.6 VF Interrupt Zero Static Control - VFINT_STAT_CTL0 (0x00005400; RW)

In case of MSI or legacy INTA mode of operation, interrupt zero is the only valid interrupt.

Field	Bit(s)	Init.	Type	Description
RESERVED	1:0	00b	RSV	Reserved.
OTHER_ITR_INDX	3:2	00b	RW	**Other ITR Index** ITR index of the other interrupt causes: 00b = ITR0. 01b = ITR1. 10b = ITR2. 11b = No ITR.
RESERVED	31:4	0x0	RSV	Reserved.

38.40.3.3.7 VF Interrupt Zero Dynamic Control - VFINT_DYN_CTL0 (0x00005C00; RW)

Field	Bit(s)	Init.	Type	Description
INTENA	0	0b	RW	**Interrupt Enable** 0b = Interrupt disabled. 1b = Interrupt enabled. Refer to auto-clear policy in the Interrupt Enablement section. This bit is meaningful only if the *INTENA_MSK* flag in this register is not set.
CLEARPBA	1	0b	RW1C	**Clear PBA** Setting this bit clears the matched PBA bit. This bit is auto-cleared by hardware.
SWINT_TRIG	2	0b	RW1C	**Software Interrupt Trigger** When this bit is set, a software interrupt is triggered. This bit is auto-cleared by hardware.
ITR_INDX	4:3	00b	RW1C	**ITR Index** Defines the ITR index to be updated, as follows: 00b = ITR0. 01b = ITR1. 10b = ITR2. 11b = No ITR update. This field is auto-cleared by hardware.
INTERVAL	16:5	0x0	RW1C	**Interval** The interval for the ITR defined by the *ITR_INDX* field in this register. This field is auto-cleared by hardware.
RESERVED	23:17	0x0	RSV	Reserved.
SW_ITR_INDX_ENA	24	0b	RW1C	**Software ITR Index Enable** Enables the programming of the *SW_ITR_INDX* field in this register. This flag is auto-cleared by hardware.

Field	Bit(s)	Init.	Type	Description
SW_ITR_INDX	26:25	00b	RW	**Software ITR Index** ITR index of the software interrupt: 00b = ITR0. 01b = ITR1. 10b = ITR2. 11b = No ITR. When programming this field, the *SW_ITR_INDX_ENA* flag in this register should be set as well.
RESERVED	29:27	000b	RSV	Reserved.
WB_ON_ITR	30	0b	RW	When this bit is set, completed descriptors are indicated to host memory on ITR completion (or No ITR) regardless of the interrupt enablement in this register.
INTENA_MSK	31	0b	RW1C	**Interrupt Enable Mask** When this bit is set, the *INTENA* setting does not impact the device setting. This bit is auto-cleared by hardware.

38.40.3.4 VF - Admin Queue

38.40.3.4.1 VF Admin Receive Queue Base Address High - VF_ARQBAH (0x00006000; RW)

Field	Bit(s)	Init.	Type	Description
ARQBAH	31:0	0x0	RW	**Admin Receive Queue Base Address High** Receive descriptor base address high.

38.40.3.4.2 VF Admin Transmit Head - VF_ATQH (0x00006400; RW)

Field	Bit(s)	Init.	Type	Description
ATQH	9:0	0x0	RW	**Admin Transmit Queue Head** Transmit queue head pointer. At queue initialization, software clears the head pointer. During normal operation, firmware increments the head following command execution.
RESERVED	31:10	0x0	RSV	Reserved.

38.40.3.4.3 VF Admin Transmit Queue Length - VF_ATQLEN (0x00006800; RW)

Field	Bit(s)	Init.	Type	Description
ATQLEN	9:0	0x0	RW	**Admin Transmit Queue Length** Descriptor ring length. Maximum size is 1024.
RESERVED	27:10	0x0	RSV	Reserved.
ATQVFE	28	0b	RW	**Admin Transmit Queue** VF Error Set by firmware on a PF queue when one of its VFs has an admin queue error.
ATQOVFL	29	0b	RW	**Admin Transmit Queue** Overflow Error Set by firmware when a message is lost because there is no room in the queue.
ATQCRIT	30	0b	RW	**Admin Transmit Queue** Critical Error Set by firmware when a critical error is detected on this queue.
ATQENABLE	31	0b	RW	**Admin Transmit Queue** Enable Set by the software device driver to indicate that the queue is active. When setting the enable bit, software should initialize all other fields. This flag is cleared by VFR.

38.40.3.4.4 VF admin receive queue base address low - VF_ARQBAL (0x00006C00; RW)

Field	Bit(s)	Init.	Type	Description
ARQBAL	31:0	0x0	RW	**Admin Receive Queue Base Address Low** Receive descriptor base address low. Must be 64-byte aligned.

38.40.3.4.5 VF Admin Receive Queue Tail - VF_ARQT (0x00007000; RW)

Field	Bit(s)	Init.	Type	Description
ARQT	9:0	0x0	RW	**Admin Receive Queue Tail** Receive queue tail pointer. Incremented to indicate that there are new valid descriptors on the ring. Software can only write to this register once the queue is fully configured, and clears to zero at queue initialization.
RESERVED	31:10	0x0	RSV	Reserved.

38.40.3.4.6 VF Admin Receive Queue Head - VF_ARQH (0x00007400; RW)

Field	Bit(s)	Init.	Type	Description
ARQH	9:0	0x0	RW	**Admin Receive Queue Head** Receive queue head pointer. At queue initialization, software clears the head pointer. During normal operation, firmware increments the head following command execution.
RESERVED	31:10	0x0	RSV	Reserved.

38.40.3.4.7 VF Admin Transmit Queue Base Address High - VF_ATQBAH (0x00007800; RW)

Field	Bit(s)	Init.	Type	Description
ATQBAH	31:0	0x0	RW	**Admin Transmit Queue Base Address High** Transmit descriptor base address high.

38.40.3.4.8 VF Admin Transmit Queue Base Address Low - VF_ATQBAL (0x00007C00; RW)

Field	Bit(s)	Init.	Type	Description
ATQBAL	31:0	0x0	RW	**Admin Transmit Queue Base Address Low** Transmit descriptor base address low. Must be 64-byte aligned.

38.40.3.4.9 VF Admin Receive Queue Length - VF_ARQLEN (0x00008000; RW)

Field	Bit(s)	Init.	Type	Description
ARQLEN	9:0	0x0	RW	**Admin Receive Queue Length** Descriptor ring length. Maximum size is 1024.
RESERVED	27:10	0x0	RSV	Reserved.
ARQVFE	28	0b	RW	**Admin Receive Queue** VF Error Set by firmware on a PF queue when one of its VFs has an admin queue error.
ARQOVFL	29	0b	RW	**Admin Receive Queue** Overflow Error Set by firmware when a message is lost because there is no room in the queue.
ARQCRIT	30	0b	RW	**Admin Receive Queue** Critical Error Set by firmware when a critical error is detected on this queue.
ARQENABLE	31	0b	RW	**Admin Receive Queue** Enable Set by software device driver to indicate that the queue is active. When setting the enable bit, software should initialize all other fields. This flag is cleared by PFR.

38.40.3.4.10 VF Admin Transmit Tail - VF_ATQT (0x00008400; RW)

Field	Bit(s)	Init.	Type	Description
ATQT	9:0	0x0	RW	**Admin Transmit Queue Tail** Transmit queue tail pointer. Incremented to indicate that there are new valid descriptors on the ring. Software can only write to this register once both transmit and receive queues are properly initialized, and clears to zero at queue initialization.
RESERVED	31:10	0x0	RSV	Reserved.

38.40.3.5 VF - LAN Transmit Receive Registers

38.40.3.5.1 Global Transmit Queue Tail - QTX_TAIL[Q] (0x00000000 + 0x4*Q, Q=0...15; RW)

Field	Bit(s)	Init.	Type	Description
TAIL	12:0	0x0	RW	**Transmit Tail** Defines the first descriptor that software prepares for hardware (it is the last valid descriptor plus one). The tail is a relative descriptor index to the beginning of the transmit descriptor ring.
RESERVED	31:13	0x0	RSV	Reserved.

38.40.3.5.2 Global Receive Queue Tail - QRX_TAIL[Q] (0x00002000 + 0x4*Q, Q=0...15; RW)

Field	Bit(s)	Init.	Type	Description
TAIL	12:0	0x0	RW	**Receive Tail** Defines the first descriptor that software hands to hardware (it is the last valid descriptor plus one). The tail is a relative descriptor index to the beginning of the receive descriptor ring.
RESERVED	31:13	0x0	RSV	Reserved.

38.40.3.6 VF - Rx Filters Registers

38.40.3.6.1 VF Queue Filter Hash Enabled Packet Type - VFQF_HENA[n] (0x0000C400 + 0x4*n, n=0...1; RW)

Field	Bit(s)	Init.	Type	Description
PTYPE_ENA	31:0	0x0	RW	**Packet Type Enable** Packet type enablement of the hash filter for the function. Bit 'm' in register 'n' enables packet type '32*n+m' as defined in the Packet Types for the Classification Filters table.

38.40.3.6.2 VF Queue Filter Hash Region of Queues - VFQF_HREGION[n] (0x0000D400 + 0x4*n, n=0...7; RW)

Field	Bit(s)	Init.	Type	Description
OVERRIDE_ENA_0	0	0b	RW	**Override Enable 0** Override TC region for packet type '8*n', where 'n' is the register index.
REGION_0	3:1	000b	RW	**Region 0** Receive queue region for packet type '8*n', where 'n' is the register index. This field is meaningful only if the *OVERRIDE_ENA_1* flag is set.
OVERRIDE_ENA_1	4	0b	RW	**Override Enable 1** Override TC region for packet type '8*n+1', where 'n' is the register index.
REGION_1	7:5	000b	RW	**Region 1** Receive queue region for packet type '8*n+1', where 'n' is the register index. This field is meaningful only if the *OVERRIDE_ENA_1* flag is set.
OVERRIDE_ENA_2	8	0b	RW	**Override Enable 2** Override TC region for packet type '8*n+2', where 'n' is the register index.
REGION_2	11:9	000b	RW	**Region 2** Receive queue region for packet type '8*n+2', where 'n' is the register index. This field is meaningful only if the *OVERRIDE_ENA_1* flag is set.
OVERRIDE_ENA_3	12	0b	RW	**Override Enable 3** Override TC region for packet type '8*n+3', where 'n' is the register index.
REGION_3	15:13	000b	RW	**Region 3** Receive queue region for packet type '8*n+3', where 'n' is the register index. This field is meaningful only if the *OVERRIDE_ENA_1* flag is set.
OVERRIDE_ENA_4	16	0b	RW	**Override Enable 4** Override TC region for packet type '8*n+4', where 'n' is the register index.
REGION_4	19:17	000b	RW	**Region 4** Receive queue region for packet type '8*n+4', where 'n' is the register index. This field is meaningful only if the *OVERRIDE_ENA_1* flag is set.
OVERRIDE_ENA_5	20	0b	RW	**Override Enable 5** Override TC region for packet type '8*n+5', where 'n' is the register index.
REGION_5	23:21	000b	RW	**Region 5** Receive queue region for packet type 8*n+5', where 'n' is the register index. This field is meaningful only if the *OVERRIDE_ENA_1* flag is set.
OVERRIDE_ENA_6	24	0b	RW	**Override Enable 6** Override TC region for packet type '8*n+6', where 'n' is the register index.
REGION_6	27:25	000b	RW	**Region 6** Receive queue region for packet type '8*n+6', where 'n' is the register index. This field is meaningful only if the *OVERRIDE_ENA_1* flag is set.
OVERRIDE_ENA_7	28	0b	RW	**Override Enable 7** Override TC region for packet type '8*n+7', where 'n' is the register index.

Field	Bit(s)	Init.	Type	Description
REGION_7	31:29	000b	RW	**Region 7** Receive queue region for packet type '8*n+7', where 'n' is the register index. This field is meaningful only if the *OVERRIDE_ENA_1* flag is set.

38.40.3.7 VF - Protocol Engine Registers

38.40.3.7.1 Protocol Engine VF IP Config 0 - VFPE_IPCONFIG0 (0x00008C00; RW)

This register is used to set or view the *IPID* field that the PE writes into the IP header. The PE increments this value for each outgoing IP datagram.

Field	Bit(s)	Init.	Type	Description
PEIPID	15:0	0x0	RW	**PE IP Identification** Specifies the *IP Identification* field used for IPv4 IP header generation. This register is initialized by firmware or the software device driver, and incremented by hardware with each IPv4 datagram transmitted by PE both UDA and iWARP on the given PCI function.
USEENTIREIDRANGE	16	0b	RW	**Use Entire ID Range** Specifies that the PE should use the entire 16-bit range for the IPID value.
RESERVED	31:17	0x0	RSV	Reserved.

38.40.3.7.2 Protocol Engine VF MRTE Index Mask - VFPE_MRTEIDXMASK (0x00009000; RW)

Field	Bit(s)	Init.	Type	Description
MRTEIDXMASKBITS	4:0	0x0	RW	**MRTE Index Mask Bits** Specifies the number of bits to be used for the MRTE index from the software device driver portion of STag. The remaining bits (24 - *MRTEIDEMASKBITS*) are randomized by the software device driver. The minimum value for this field is 14 and the maximum is 22. Values outside of this range are normalized to 14 or 22 by the hardware.
RESERVED	31:5	0x0	RSV	Reserved.

38.40.3.7.3 Protocol Engine VF Unexpected Error - VFPE_RCVUNEXPECTEDERROR (0x00009400; RO)

Field	Bit(s)	Init.	Type	Description
TCP_RX_UNEXP_ERR	23:0	0x0	RW	**TCP Received Unexpected Error** For PE accelerated connections only, counts the total number of Ethernet frames received by the product that failed TCP sanity checks indicating unexpected filtering error in preceding module.
RESERVED	31:24	0x0	RSV	Reserved.

38.40.3.7.4 Protocol Engine VF Create CQP High - VFPE_CCQPHIGH (0x00009800; RW)

This register stores the upper 32 bits of the 64-bit physical address of the control QP context for its associated PCI function.

Field	Bit(s)	Init.	Type	Description
PECCQPHIGH	31:0	0x0	RW	**PE Create CQP High** Most significant bits of the control QP context physical address in host memory.

38.40.3.7.5 Protocol Engine CQP Error Codes - VFPE_CQPERRCODES (0x00009C00; RO)

This register reports errors encountered by CQP when CQ0 is not available. The contents of this register are only valid when the associated CCQPSTATUS.*CCQP_ERR* bit or CQPTAIL.*CQP_ERR* bit is set.

Field	Bit(s)	Init.	Type	Description
CQP_MINOR_CODE	15:0	0x0	RW	**CQP Minor Code** Minor code that would have been reported in a CQP Completion.
CQP_MAJOR_CODE	31:16	0x0	RW	**CQP Major Code** Major code that would have been reported in a CQP Completion.

38.40.3.7.6 Protocol Engine CQP Tail - VFPE_CQPTAIL (0x0000A000; RO)

This register is used to determine how much work the PE control QP has pending. Software can determine if CQP has pending work by comparing the last value written to *WQHEAD* in the CQPDB register to *WQTAIL* after reading this register.

Field	Bit(s)	Init.	Type	Description
WQTAIL	10:0	0x0	RW	**Work Queue Tail** Indicates the WQE index of the next WQE that the CQP processes.
RESERVED	30:11	0x0	RSV	Reserved.
CQP_OP_ERR	31	0b	RW	**CQP Operation Error** Indicates that the CQP encountered an error processing an operation. If software has multiple requests outstanding to the CQP at the time of the error, *WQTAIL* might not indicate the WQE that caused the error.

38.40.3.7.7 Protocol Engine VF AEQ Allocate - VFPE_AEQALLOC (0x0000A400; RW)

Field	Bit(s)	Init.	Type	Description
AECOUNT	31:0	0x0	RW	**Asynchronous Event Count** Specifies the number of asynchronous event queue entries that have been processed by software and can now be reused by hardware.

38.40.3.7.8 Protocol Engine VF TCP Now Timer - VFPE_TCPNOWTIMER (0x0000A800; RO)

Field	Bit(s)	Init.	Type	Description
TCP_NOW	31:0	0x0	RO	**TCP Now** Contains the current value of tcp_now. *TCP_NOW* is a 32-bit counter and provides the TCP time measurement for the all of the timers. It is also used to calculate the TS value sent in the TCP time stamp option, tcp_now is added to tsval_tick_delta to form the TS value.

38.40.3.7.9 Protocol Engine VF Create CQP Low - VFPE_CCQPLOW (0x0000AC00; RW)

This register stores the lower 32 bits of the 64-bit physical address of the control QP context for its associated PCI function. Under host software control, each PCI function uses its Create Control QP High/Low registers to create its corresponding Control QP (QP0). The 64-bit address must always be updated by writing the *PECCQPLOW* field last.

Field	Bit(s)	Init.	Type	Description
PECCQPLOW	31:0	0x0	RW	**PE Create CQP Low** Least significant bits of the control QP context physical address in host memory.

38.40.3.7.10 Protocol Engine VF CQ ACK - VFPE_CQACK (0x0000B000; RW)

This register is used to acknowledge a completion event for a PE completion queue. The interrupt processing logic that handles completion events must write this register in order to enable new events for a completion queue.

Field	Bit(s)	Init.	Type	Description
PECQID	16:0	0x0	RW	**PE Completion Queue ID** Used to enable new events for a PE completion queue.
RESERVED	31:17	0x0	RSV	Reserved.

38.40.3.7.11 Protocol Engine VF CQ Arm - VFPE_CQARM (0x0000B400; RW)

Note: This register is also located in the PE doorbell page section of the BAR.

Field	Bit(s)	Init.	Type	Description
PECQID	16:0	0x0	RW	**PE Completion Queue ID** Used to arm a PE completion queue for generating events in conjunction with the completion queue doorbell shadow area located in host memory. Arming is also frequently referred to as requesting notification for a completion queue. Events can be generated when the next completion is generated or when the next completion related to a solicited operation is generated. On read, this register returns the value of 0x0.
RESERVED	31:17	0x0	RSV	Reserved.

38.40.3.7.12 Protocol Engine VF Create CQP Status - VFPE_CCQPSTATUS (0x0000B800; RW)

This register is used to indicate the progress of creating the control QP for a given PCI function. During host software initialization, these bits are initially zero. Each bit is set when the associated step of the initialization process completes.

Field	Bit(s)	Init.	Type	Description
CCQP_DONE	0	0b	RW	**Create CQP Done** 0b = Indicates that the CQP has not been created. 1b = Indicates that the create CQP operation triggered by writing to PECCQPHIGH and PECCQPLOW completed.
RESERVED	3:1	000b	RSV	Reserved.

Field	Bit(s)	Init.	Type	Description
HMC_PROFILE	6:4	000b	RW	**HMC Profile** Specifies the HMC resource profile that is active. 000b = Reserved. 001b = Default. 010b = SR-IOV VF primary. 011b = SR-IOV even distribution. All other values are reserved.
RESERVED	15:7	0x0	RSV	Reserved.
RDMA_EN_VFS	21:16	0x0	RW	**RDMA Enabled VFs** Specifies the number of RDMA enabled VFs allocated in the HMC resource profile.
RESERVED	30:22	0x0	RSV	Reserved.
CCQP_ERR	31	0b	RW	**Create CQP Error** Indicates that the CQP encountered an error processing the last create CQP request. This bit is reset when CQP is not created.

38.40.3.7.13 Protocol Engine VF CQP Doorbell - VFPE_CQPDB (0x0000BC00; RW)

This register is used to post work the PE control QP. Software can determine if CQP has pending work by comparing *WQHEAD* to *WQTAIL* after reading this register. Software must first populate one or more WQEs in the CQP WQ and then put the index of the WQE following the last populated WQE into *WQHEAD* in order to submit work to CQP.

Field	Bit(s)	Init.	Type	Description
WQHEAD	10:0	0x0	RW	**Work Queue Head** Indicates the WQE index of the next WQE that software posts to the CQP.
RESERVED	31:11	0x0	RSV	Reserved.

38.40.3.7.14 Protocol Engine VF WQE Allocate Register - VFPE_WQEALLOC (0x0000C000; RW)

This register is used to post work the PE queue pairs.

Note: This register is also located in the PE doorbell page section of the BAR.

Field	Bit(s)	Init.	Type	Description
PEQPID	17:0	0x0	RW	**PE Queue Pair ID** This field should be used to notify hardware the new work is available to be processed.
RESERVED	19:18	00b	RSV	Reserved.
WQE_DESC_INDEX	31:20	0x0	RW	**WQE Index** Indicates a high 12 bits of the WQE index in the send queue. This field is used to identify a push mode message in the head of the SQ. Push mode message is processed by hardware only if it is in the head of the send queue, and otherwise it is dropped and processed when WQE is re-fetched during regular transmit operation. This field is used for the push mode doorbells only. For the regular doorbells, this field should be treated as reserved, and ignored by hardware.

38.40.4 BAR3 Registers Summary

Table 38-613.BAR3 Registers Summary

Offset / Alias Offset	Abbreviation	Name	Page
VF - MSI-X Table Registers			
0x00000000 + 0x10*n, n=0...16	VFMSIX_TADD[n]	VF MSI-X Message Address Low	3698
0x00000004 + 0x10*n, n=0...16	VFMSIX_TUADD[n]	VF MSI-X Message Address High	3698
0x00000008 + 0x10*n, n=0...16	VFMSIX_TMSG[n]	VF MSI-X Message Data	3699
0x0000000C + 0x10*n, n=0...16	VFMSIX_TVCTRL[n]	VF MSI-X Vector Control	3699
0x00002000	VFMSIX_PBA	VF MSI-X PBA Structure	3699

38.40.5 Detailed Register Description - VF BAR3

38.40.5.1 VF - MSI-X Table Registers

38.40.5.1.1 VF MSI-X Message Address Low - VFMSIX_TADD[n] (0x00000000 + 0x10*n, n=0...16; RW)

Message lower address for MSI-X table entries.

Field	Bit(s)	Init.	Type	Description
MSIXTADD10	1:0	0x0	RW	**Message Address 1:0** For proper Dword alignment, software must always write zeros to these two bits; otherwise, the result is undefined. The state of these bits after reset must be 0b. These bits are permitted to be read-only or read/write.
MSIXTADD	31:2	0x0	RW	**Message Address** System-specified message lower address. For MSI-X messages, the contents of this field from an MSI-X table entry specifies the lower portion of the Dword-aligned address (AD[31:02]) for the memory write transaction. This field is read/write.

38.40.5.1.2 VF MSI-X Message Address High - VFMSIX_TUADD[n] (0x00000004 + 0x10*n, n=0...16; RW)

Message upper address for MSI-X table entries.

Field	Bit(s)	Init.	Type	Description
MSIXTUADD	31:0	0x0	RW	**Message Upper Address** System-specified message upper address bits. If this field is zero, SAC messages are used. If this field is non-zero, DAC messages are used. This field is read/write.

38.40.5.1.3 VF MSI-X Message Data - VFMSIX_TMSG[n] (0x00000008 + 0x10*n, n=0…16; RW)

Message data for MSI-X table entries.

Field	Bit(s)	Init.	Type	Description
MSIXTMSG	31:0	0x0	RW	**Message Data** System-specified message data. For MSI-X messages, the contents of this field from an MSI-X table entry specifies the data driven on AD[31:0] during the memory write transaction's data phase. This field is read/write.

38.40.5.1.4 VF MSI-X Vector Control - VFMSIX_TVCTRL[n] (0x0000000C + 0x10*n, n=0…16; RW)

Vector control for MSI-X table entries.

Field	Bit(s)	Init.	Type	Description
MASK	0	1b	RW	**Mask Bit** When this bit is set, the function is prohibited from sending a message using this MSI-X table entry. However, any other MSI-X table entries programmed with the same vector are still capable of sending an equivalent message unless they are also masked. This bit's state after reset is 1b (entry is masked).
RESERVED	31:1	0x0	RSV	Reserved. After reset, the state of these bits must be 0x0. However, for potential future use, software must preserve the value of these reserved bits when modifying the value of other *Vector Control* bits. If software modifies the value of these reserved bits, the result is undefined.

38.40.5.1.5 VF MSI-X PBA Structure - VFMSIX_PBA (0x00002000; RO)

Pending bits for MSI-X PBA entries.

Field	Bit(s)	Init.	Type	Description
PENBIT	31:0	0x0	RO	**MSI-X Pending Bits** Each bit is set to 1b when the appropriate interrupt request is set, and cleared to 0b when the appropriate interrupt request is cleared.

38.41 PCIe Programming Interface

The 10 GbE controller supports the following configuration register sets:

- PCI basic configuration registers.
- PCI and PCIe capabilities in the PCI configuration space (see Section 38.43).
  - Includes the PCIe capability structure.
- PCIe capabilities residing in the PCIe extended configuration Space (see Section 38.44).
- SR-IOV VF configuration space (see Section 38.45).

38.41.1 Functions Mapping

The 10 GbE controller is a multi-function device with the following characteristics:

- Up to 128 SR-IOV VFs
  - Each PF can be allocated a different number of VFs in the range {0,...,128} as long as the total number of VFs does not exceed 128.

The following rules apply regarding allocation of PCI functions to LAN ports:

- Each PCI function is associated with a single LAN port as indicated in the PFGEN_PORTNUM.PORT_NUM register field.

The number of enabled physical functions might be deducted from the PFPCI_STATUS1.FUNC_VALID bits (one per PF). Refer to the section about how enabled functions are determined.

The ARI capability enables interpretation of the device number part of the RID as part of the function number inside a device. Thus, a single device can span more than eight physical or virtual functions. Table 38-614 and Table 38-615 map the physical functions to PCI requester ID.

Table 38-614.RID Per PF - ARI Mode

PF#	B,D,F	Binary	Notes
PF 0	B,0,0	B,00000,000	PF #0
PF 1	B,0,1	B,00000,001	PF #1
PF 2	B,0,2	B,00000,010	PF #2
PF 15	B,1,7	B,00001,111	PF #15

Table 38-615.RID Per PF - non-ARI Mode

PF#	B,D,F	Binary	Notes
PF 0	B,0,0	B,00000,000	PF #0
PF 1	B,0,1	B,00000,001	PF #1
PF 2	B,0,2	B,00000,010	PF #2
PF 7	B,0,7	B,00000,111	PF #7

The requester ID of a PF (bus, device, and function numbers) is captured in the PF_FUNC_RID register.

38.41.2 Supported Features

Table 38-616 lists the PCI and PCIe capabilities supported per PCI function type. Some capabilities do not necessarily appear with each function or in all cases. See Section 38.43 and Section 38.44 for details on specific capabilities.

Table 38-616.PCI Capabilities Supported by Function

PCI Capability	PFs	VFs
PCI Configuration	Mandatory	Mandatory
Power Management	Yes	Yes
MSI	Yes	No
MSI-X	Yes	Yes
Vital Product Data (VPD)	Yes	No
PCIe	Yes	Yes
Advanced Error Reporting (AER)	Yes	Yes
Device Serial Number	Yes	No (N/A)
Alternative RID Interpretation (ARI)	Yes	Yes
Single Root I/O Virtualization (SR-IOV)	Yes	No (N/A)
TPH Requester	Yes	Yes
Access Control Services (ACS)	Yes	Yes
Secondary PCIe	Yes	No (N/A)

38.42 PCI Configuration Space

Configuration registers are assigned one of the attributes listed in the table that follows.

38.42.1 Register Attributes

The following table lists the register attributes used in this section.

RD/WR	Description
RO	Read-only register: Register bits are read-only and cannot be altered by software.
RW	Read-write register: Register bits are read-write and can be either set or reset.
RW1C	Read-only status, Write-1b-to-clear status register, Writing a 0b to RW1C bits has no effect.
ROS	Read-only register with sticky bits: Register bits are read-only and cannot be altered by software. Bits are neither initialized nor modified by PCIe in-band reset or FLR. Specific bits listed below are also not reset on PERST# when aux power consumption is enabled.
RWS	Read-write register: Register bits are read-write and can be either set or cleared by software to the desired state. Bits are neither initialized nor modified by PCIe in-band reset or FLR. Specific bits listed below are also not reset on PERST# when aux power consumption is enabled.
RW1CS	Read-only status, Write-1b-to-clear status register: Register bits indicate status when read, a set bit, indicating a status event, can be cleared by writing a 1b to it. Writing a 0b to RW1C bits has no effect. Bits are neither initialized nor modified by PCIe in-band reset or FLR. Specific bits listed in the sections that follow are also not reset on PERST# when aux power consumption is enabled.
HwInit	Hardware initialized: Register bits are initialized by firmware or hardware mechanisms such as pin strapping or serial NVM. Bits are read-only after initialization and can only be reset (for write-once by firmware) with the PWRGOOD signal.

RD/WR	Description
RsvdP	Reserved and preserved: Reserved for future read/write implementations; software must preserve value read for writes to these bits.
RsvdZ	Reserved and zero: Reserved for future RW1C implementations; software must use 0b for writes to these bits.

38.42.2 Reset Rules

Reset of the PCI configuration space (including any capability lists) is per the PCIe specification. Several cases require special attention.

38.42.2.1 Sticky REGISTERS

The following sticky register fields are also not reset on PERST# when aux power consumption is enabled (AUX_PWR pin is set).

- PME related fields:
  - Power Management Capabilities register — PME_En bit
  - Power Management Capabilities register — PME_Status bit
  - Device Control register — *Aux Power PM Enable* bit
  - The function Requester ID
- AER related fields:
  - Uncorrectable Error Status register
  - Uncorrectable Error Mask register
  - Uncorrectable Error Severity register
  - Correctable Error Status register
  - Correctable Error Mask register
  - Advanced Error Capabilities and Control register
  - Header log

38.42.2.2 Reset on FLR

The following registers are not affected by FLR:

- The *Max Payload Size* field in the Device Control register
- The *Active State Link PM Control* field in the Link Control register
- The *Common Clock Configuration* field in the Link Control register
- The *Extended Sync* field in the Link Control register
- The *Link Equalization Request* field in the Link Status 2 register

38.42.3 PCI Configuration Space Summary

Table 38-617 lists the PCI configuration registers while their detailed description is given in the sections that follow. Fields that have meaningful default values are indicated in parenthesis — (**value**).

The PCI configuration space from address 0x40 on is allocated for PCI capability structures as described in Section 38.43. However, the region of 0x98-0x9F (8 bytes) is dedicated to an address/data port, which provides access to the device I/O address space.

Table 38-617.PCI Configuration Space - PF

Section	Byte Offset	Byte 3	Byte 2	Byte 1	Byte 0
Mandatory PCI Register	0x0	Device ID		Vendor ID	
	0x4	Status Register		Command Register	
	0x8	Class Code (0x020000/0x010000)			Revision ID
	0xC	Reserved	Header Type (0x0/ 0x80)	Latency Timer	Cache Line Size (0x10)
	0x10	Base Address Register 0			
	0x14	Base Address Register 1			
	0x18	Base Address Register 2			
	0x1C	Base Address Register 3			
	0x20	Base Address Register 4			
	0x24	Base Address Register 5			
	0x28	CardBus CIS Pointer (0x0000)			
	0x2C	Subsystem ID		Subsystem Vendor ID	
	0x30	Expansion ROM Base Address			
	0x34	Reserved			Cap Ptr (0x40)
	0x38	Reserved			
	0x3C	Max Latency (0x00)	Min Grant (0x00)	Interrupt Pin (0x01...0x04)	Interrupt Line (0x00)

38.42.4 Sharing Among PCI Functions

The 10 GbE controller supports multiple PCI functions. As each function exposes a PCIe configuration space, each register and each field is either shared among the functions or is replicated per each PCI function. This section summarizes configuration sharing of the fixed PCI configuration space. See the description of each PCI capability structure for configuration sharing within it. Also, the description of each field describes special considerations regarding configuration sharing.

Table 38-618.Configuration Sharing of PCI Configuration Space

Field	Sub-field	Shared?	Replicated?	Comments
Vendor ID	Vendor ID	X		
Device ID	Device ID		X	
Command Register	I/O Access Enable		X	Issue UR per PF if disabled.
	Memory Access Enable		X	Issue UR per PF if disabled.
	Bus Master Enable		X	
	Parity Error Response		X	Enables certain error reporting per PF.
	SERR# Enable		X	Controls error reporting per PF.
	Interrupt Disable		X	Selection of interrupt method per PF.
Status Register	Interrupt Status		X	
	Capabilities List	X		Hardwired to 1b.
	Data Parity Reported / Master Data Parity Error		X	Reports poisoned packets per PF.
	Signaled Target Abort		X	Reports completer abort per PF.
	Received Target Abort		X	Reports receiving a completer abort per PF.

Table 38-618.Configuration Sharing of PCI Configuration Space

Field	Sub-field	Shared?	Replicated?	Comments
	Received Master Abort		X	Reports receiving an UR per PF.
	Signaled System Error		X	Reports Fatal / non-fatal message per PF.
	Detected Parity Error		X	Reports receiving a poisoned TLP per PF.
Revision Register		X		
Class Code Register			X	Per function type.
Cache Line Size Register			X	Does not affect device behavior.
Latency Timer		X		Hardwired to 0x00 in PCIe.
Header Type Register		X		
Reserved		X		
BARs	Memory BAR		X	
	I/O BAR		X	
	MSI-X BAR		X	See MSI-X capability.
I/O BAR mapping	IOADDR, IODATA		X	
Subsystem Vendor ID		X		
Subsystem ID			X	
Expansion ROM		X		Each PF has its own BAR.
Cap_Ptr Register		X		
Interrupt Line Register			X	Just store the register value.
Interrupt Pin Register			X	Separate interrupt number (A-D) per PF.
Min Grant		X		
Max Latency		X		

38.42.5 Mandatory PCI Configuration Registers — Except BARs

38.42.5.1 Vendor ID Register (0x0; RO)

This is a read-only register that has the same value for all PCI functions.

- Vendor ID is loaded from the NVM if the *GLPCI_CAPSUP.LOAD_DEV_ID* bit is set.
- The value for all PFs is loaded from the NVM *GLPCI_VENDORID.VENDOR_ID* field.

38.42.5.2 Device ID Register (0x2; RO)

This is a read-only register that identifies individual Product Name PCI functions. All functions have the same default value of 0x3A84, and can be auto-loaded from the NVM during initialization with different values for each function as well as the dummy function.

Device ID is loaded from the NVM according to the following rules:

- The Device ID is loaded from the NVM if the *GLPCI_CAPSUP.LOAD_DEV_ID* bit is set.
- The value of each PF is loaded to the respective *PFPCI_DEVID.PF_DEV_ID* field.

38.42.5.3 Command register (0x4; RW)

Shaded bits are not used by this implementation and are hardwired to 0b. Each function has its own Command register (see Table 38-618).

Bit(s)	Initial Value	RW	Description
0	0b	RW (see comment)	I/O Access Enable. This bit is RO if an I/O BAR is not supported by the device.
1	0b	RW	Memory Access Enable.
2	0b	RW	Enable mastering, also named Bus Master Enable (BME). • LAN functions RW field. • Dummy function RO as zero field.
3	0b	RO	Special Cycle Monitoring – Hardwired to 0b.
4	0b	RO	MWI Enable – Hardwired to 0b.
5	0b	RO	Palette Snoop Enable – Hardwired to 0b.
6	0b	RW	Parity Error Response.
7	0b	RO	Wait Cycle Enable – Hardwired to 0b.
8	0b	RW	SERR# Enable.
9	0b	RO	Fast Back-to-Back Enable – Hardwired to 0b.
10	0b	RW	Interrupt Disable. When set, devices are prevented from generating legacy interrupt messages. RO as 1b for a dummy function.
15:11	0b	RO	Reserved.

38.42.5.4 Status Register (0x6; RO)

Shaded bits are not used by this implementation and are hardwired to 0b. Each function has its own Status register.

Bits	Initial Value	RW	Description
2:0	0b		Reserved.
3	0b	RO	Interrupt Status.[1]
4	1b	RO	New Capabilities. Indicates that a device implements extended capabilities. The 10 GbE controller sets this bit and implements a capabilities list to indicate that it supports PCI and PCIe capabilities.
5	0b		66 MHz Capable – Hardwired to 0b.
6	0b		Reserved.
7	0b		Fast Back-to-Back Capable – Hardwired to 0b.
8	0b	RW1C	Data Parity Reported.
10:9	00b		DEVSEL Timing – Hardwired to 0b.
11	0b	RW1C	Signaled Target Abort.
12	0b	RW1C	Received Target Abort.
13	0b	RW1C	Received Master Abort.
14	0b	RW1C	Signaled System Error.
15	0b	RW1C	Detected Parity Error.

Notes:

1. The *Interrupt Status* field is a RO field that indicates that an interrupt message is pending internally to the device.

38.42.5.5 Revision Register (0x8; RO)

The default revision ID of this device is 0x00 for A0 step and 0x01 for B0 step. The default value is readable through the *GLPCI_DREVID* register. The value in the Revision register is a logic XOR between the default value and a value loaded from the NVM, reflected via the *GLPCI_REVID* register.

38.42.5.6 Class Code Register (0x9; RO)

The class code is a read-only value that identifies the device functionality:

- Class Code = 0x020000 (Ethernet adapter) if NVM->*Storage Class* = 0b.
- Class Code = 0x010000 (SCSI storage device) if NVM->*Storage Class* = 1b.

In the dummy function the class code equals to 0xFF0000.

Device default value is class code of an Ethernet adapter. The value is overwritten from the NVM. It is loaded to the RO PFPCI_CLASS register.

38.42.5.7 Cache Line Size Register (0xC; RW)

This field is implemented by PCIe devices as a read/write field for legacy compatibility purposes but has no impact on any PCIe device functionality. All functions are initialized to the same value of 0x00 (specification required).

38.42.5.8 Latency Timer (0xD; RO)

Not used. Hardwired to 0b.

38.42.5.9 Header Type Register (0xE; RO)

This indicates if a device is single- or multi-function:

- 0x00 — A single function is enabled
- 0x80 — At least two functions are enabled
- Dummy functions are considered as regular functions in this regard.
- Functions might be disabled during the power on reset flow (through strapping pins, SMASH/CLP commands, NC-SI commands) affecting this bit.

38.42.5.10 Subsystem Vendor ID Register (0x2C; RO)

This value is loaded from the NVM if the *GLPCI_CAPSUP.LOAD_SUBSYS_ID* bit is set. A value of 0x8086 is the default for this field. It can be read through the *GLPCI_SUBVENID* register.

38.42.5.11 Subsystem ID Register (0x2E; RO)

This value is loaded from the NVM if the *GLPCI_CAPSUP.LOAD_SUBSYS_ID* bit is set. Each PF is loaded to the respective PFPCI_SUBSYSID.PF_SUBSYS_ID field.

38.42.5.12 Capabilities Pointer Register (0x34; RO)

The *Capabilities Pointer* field (Cap_Ptr) is an 8-bit field that provides an offset in the 10 GbE controller's PCI configuration space for the location of the first item in the capabilities linked list. The 10 GbE controller supports this field and implements a capabilities list. Its value is 0x40, which is the address of the first entry: PCI power management.

38.42.5.13 Interrupt Line Register (0x3C; RW)

Read/write register programmed by software to indicate which of the system interrupt request lines the 10 GbE controller's interrupt pin is bound to. Refer to the PCI definition for more details.

38.42.5.14 Interrupt Pin Register (0x3D; RO)

Read-only register. — A value of 0x1...0x4 indicates that this function implements a legacy interrupt on INTA#...INTD# respectively. Loaded from the NVM. It can be read through the RO PFPCI_CNF register. Device default value is 0x0.

If only a single function is enabled, the I*nterrupt Pin* field of the enabled function reports INTA# usage.

Reports a value of 0x0 (function uses no legacy interrupt message) for a dummy function.

38.42.5.15 MIN_GNT and MAX_LAT (0x3E; RO)

Not used. Hardwired to 0b.

38.42.6 Mandatory PCI Configuration Registers — BARs

38.42.6.1 Memory and I/O BARs (0x10...0x27; RW)

BARs are used to map the 10 GbE controller register space of the device functions. The 10 GbE controller has a memory BAR, an I/O BAR (optional) an MSI-X BAR as listed in Table 38-619. The BARs location and sizes are listed in Table 38-619 and Table 38-620. The fields within each BAR are then listed in Table 38-621.

Table 38-619.Product Name BAR Description

Mapping Windows	Mapping Description
Memory BAR	The internal registers memories and external Flash devices are accessed as direct memory mapped offsets from the BAR. Software can access a Dword or 64 bits.
I/O BAR	All internal registers and memories can be accessed using I/O operations. There are two 4-byte registers in the I/O mapping window: Addr Reg and Data Reg accessible as Dword entities. The I/O BAR is supported depending on the NVM configuration. Device default value is that an I/O BAR is not supported. The state of the I/O BAR is exposed in the PFPCI_CNF register. This BAR is not present in dummy functions.
MSI-X BAR	The MSI-X vectors and PBA structures are accessed as direct memory mapped offsets from the MSI-X BAR. Software can access Dword entities. This BAR is not present in dummy functions.

Table 38-620.Product Name base Address Setting in 64-bit BARs Mode

BAR	Addr	31 5	4	3	2	1	0
0	0x10	Memory CSR + Flash BAR Low: 31:24 = RW; 23:18 = RW or 0b; 17:4 = 0b		0/1	1	0	0
1	0x14	Memory CSR + Flash BAR High (RW)					
2	0x18	IO BAR (RW — 31:5) (optional)	0	0	0	0	1
3	0x1C	MSI-X BAR Low (RW — 31:15; RO 0b — 14:4)		0/1	1	0	0
4	0x20	MSI-X BAR High (RW)					
5	0x24	Reserved (RO — 0)					

Table 38-621.Base Address Registers' Fields

Field	Bits	RW	Description
Memory and I/O Space Indication	0	RO	0b = Indicates memory space. 1b = Indicates I/O.
Memory Type	2:1	RO	This field is loaded from the NVM. Hardware default is 64-bit. The field is exposed in the GLPCI_LBARCTRL register. 00b = 32-bit BAR. 10b = 64-bit BAR.
Prefetch Memory	3	RO	This bit is loaded from NVM. This bit should be set only on systems that do not generate prefetchable cycles. Device default is 1b (prefetchable). It is exposed in the GLPCI_LBARCTRL register. 0b = Non-prefetchable space. 1b = Prefetchable space.
Address Space (low register for 64-bit memory BARs)	31:4	RW	The length of the RW bits and RO 0b bits depend on the mapping window sizes. Initial value of the RW fields is 0x0.

Mapping Window	RO bits
Memory space for CSRs and Flash memory access.	16:4 for 128 KB 17:4 for 256 KB and so on...
MSI-X space is 32 KB.	14:4
I/O space size is 32 bytes.	4:0

38.42.6.2 Expansion ROM Base Address Register (0x30; RW)

This register is used to define the address and size information for boot-time access to the optional Flash memory. This register returns a zero value for functions without an expansion ROM window and for dummy functions.

The expansion ROM BAR is disabled through the PFPCI_CNF.EXROM_DIS register field.

Field	Bit(s)	RW	Initial Value	Description
En	0	RW	0b	1b = Enables expansion ROM access. 0b = Disables expansion ROM access.
Reserved	10:1	R	0b	Always read as 0b. Writes are ignored.
Address	31:11	RW	0b	The number of bits that are not hardwired to 0b is determined by the value of the GLPCI_LBARCTRL.EXROM_SIZE register field, loaded from the NVM.

38.43 Capabilities in PCI Configuration Space

The first entry of the PCI capabilities link list is pointed to by the Cap_Ptr register. Table 38-622 lists the capabilities supported by the 10 GbE controller that reside in the PCI configuration space.

Table 38-622.PCI Capabilities List

Address range	Item	Cases Where Capability Does Not Exist	Next Pointer
0x40:4F	Power Management	None (always exists).	0x50 / 0xA0
0x50:6F	MSI	Dummy function.	0x70 / 0xA0
0x70:8F	MSI-X	1. PFPCI_CNF.MSI_EN is 0b. 2. Dummy function.	0xA0

Table 38-622.PCI Capabilities List

Address range	Item	Cases Where Capability Does Not Exist	Next Pointer
0xA0:DF	PCIe	None (always exists).	0xE0 / 0x00
0xE0:0xEF	VPD	1. VPD Enable bit (GLPCI_CAPCTRL.VPD_EN) is cleared. 2. Dummy function.	0x00

38.43.1 PCI power management capability

Byte Offset	Byte 3	Byte 2	Byte 1	Byte 0
0x40	Power Management Capabilities		Next Pointer	Capability ID (0x01)
0x44	Data	Bridge Support Extensions	Power Management Control and Status	

Table 38-623 lists the sharing of the Power Management Capability registers among the different PCI functions.

Table 38-623.Sharing the Power Management Capability Registers

Field	Sub-field	Shared?	Replicated?	Comments
Capability ID		x		
Next Pointer			x	
Power Management Capabilities	PME_Support	x		
	D2_Support	x		
	D1_Support	x		
	AUX Current	x		
	DSI	x		
	PME Clock	x		Hardwired to 0b
	Version	x		
Power Management Control / Status	PME_Status		x	
	Data_Scale	x		
	Data_Select		x	
	PME_En		x	
	No_Soft_Reset	x		
	PowerState		x	
Data Register			x	

38.43.1.1 Capability ID Register (0x40; RO)

This field equals 0x01 indicating the linked list item as being the PCI Power Management registers.

38.43.1.2 Next Pointer Register (0x41; RO)

This field provides an offset to the next capability item in the capability list. See Table 38-622 for possible values of the next pointer register.

38.43.1.3 Power Management Capabilities — PMCR (0x42; RO)

This field describes the device functionality during the power management states as listed in the following table.

Bits	Default	RW	Description
15:11		RO	PME_Support. This 5-bit field indicates the power states in which the function can generate a PME event. condition functionality values are as follows: • No AUX Pwr PME at D0 and $D3_{HOT}$ = 01001b • AUX Pwr PME at D0, $D3_{HOT}$, and $D3_{COLD}$ = 11001b ***Note:*** For dummy function, this field is RO - zero.
10	0b	RO	D2_Support – The 10 GbE controller does not support the D2 state.
9	0b	RO	D1_Support – The 10 GbE controller does not support the D1 state.
8:6	000b	RO	AUX Current – Required current defined in the Data register.
5	1b	RO	DSI – The 10 GbE controller requires its device driver to be executed following a transition to the D0 uninitialized state.
4	0b	RsvdP	Reserved.
3	0b	RO	PME_Clock – Disabled. Hardwire to 0b.
2:0	011b	RO	Version – The 10 GbE controller complies with the PCI PM specification revision 1.2.

38.43.1.4 Power Management Control / Status Register — PMCSR (0x44; RW)

This register (listed in the following table) is used to control and monitor power management events in the device.

Bits	Default	RW	Description
15	0b (at power up)	RW1CS	PME_Status. This bit is set to 1b when the function detects a wake-up event independent of the state of the *PME_En* bit. Writing a 1b clears this bit.
14:13	00b	RO	Data_Scale. This field indicates the scaling factor that's used when interpreting the value of the Data register. This field is loaded from the NVM (through the GLPCI_PWRDATA.DATA_SCALE register) for legal values of Data_Select [0, 3, 4, 7, (and 8 for function 0)]. The normal value is 01b (indicating 0.1 watt/units). Reserved (00b) for any other values of Data_Select.
12:9	0000b	RW	Data_Select. This 4-bit field is used to select which data is to be reported through the Data register and *Data_Scale* field.
8	0b (at power up)	RWS	PME_En. Writing a 1b to this register enables wake up.
7:4	0000b	RO	Reserved.
3	1b	RO	No_Soft_Reset. This bit is always set to 1b to indicate that the 10 GbE controller does not perform an internal reset upon a transition from $D3_{HOT}$ to D0 via software control of the *PowerState* bits. Configuration context is maintained when performing the soft reset. Upon transition from the $D3_{HOT}$ to the D0 state, an initialization sequence is not needed in order to return the 10 GbE controller to the D0 Initialized state.
2	0b	RO	Reserved for PCIe.
1:0	00b	RW	PowerState. This field is used to set and report the power state of a function as follows: 00b = D0. 01b = D1 (cycle ignored if written with this value). 10b = D2 (cycle ignored if written with this value). 11b = D3

38.43.1.5 PMCSR_BSE Bridge Support Extensions Register (0x46; RO)

This register is not implemented in the 10 GbE controller; values set to 0x00.

38.43.1.6 Data Register (0x47; RO)

This optional register is used to report power consumption and heat dissipation. The reported register is controlled by the *Data_Select* field in the PMCSR; the power scale is reported in the *Data_Scale* field in the PMCSR. The data for this field is loaded from the NVM with a default value of 0x00. It is exposed through the GLPCI_PWRDATA register.

The values for the 10 GbE controller functions are as follows (the relevant column is selected based on the value of the Data_Select field):

Field	D0 (Consume/ Dissipate)	D3 (Consume/ Dissipate)	Common
Data_Select	(0x0/0x4)	(0x3/0x7)	(0x8)
Function 0	Loaded from NVM	Loaded from NVM	Multi-function value: Loaded from the NVM Single-function value: 0x00
Other functions	Loaded from NVM	Loaded from NVM	0x00

Note: For other Data_Select values the Data register output is reserved (0x00).

38.43.2 MSI Capability

This structure is enabled when the PFPCI_CNF.MSI_EN bit is set for the function

Byte Offset	Byte 3	Byte 2	Byte 1	Byte 0
0x50	Message Control (0x0080)		Next Pointer	Capability ID (0x05)
0x54	Message Address			
0x58	Message Upper Address			
0x5C	Reserved		Message Data	
0x60	Mask Bits			
0x64	Pending Bits			

Table 38-624 lists configuration sharing of the MSI Capability registers among the different PCI functions.

Table 38-624.Configuration Sharing of the MSI Capability

Field	Sub-Field	Shared?	Replicated?	Comments
Capability ID		x		
Next Pointer			x	
Message Control	MSI Enable		x	
	Multiple Messages Capable	x		
	Multiple Message Enable		x	
	64-bit Capable	x		
	MSI Per-vector Masking	x		
Message Address Low			x	
Message Address High			x	

Table 38-624.Configuration Sharing of the MSI Capability

Field	Sub-Field	Shared?	Replicated?	Comments
Message Data			x	
Mask Bits			x	
Pending Bits			x	

38.43.2.1 Capability ID Register (0x50; RO)

This field equals 0x05 indicating that the linked list item as being the MSI registers.

38.43.2.2 Next pointer Register (0x51; RO)

This field provides an offset to the next capability item in the capability list. See Table 38-622 for possible values of the next pointer register.

38.43.2.3 Message Control Register (0x52; RW)

Bits	Default	RW	Description
0	0b	RW	MSI Enable. 1b = Message Signaled Interrupts. The 10 GbE controller generates an MSI for interrupt assertion instead of INTx signaling.
3:1	000b	RO	Multiple Messages Capable. The 10 GbE controller indicates a single requested message per function.
6:4	000b	RW	Multiple Message Enable. Software writes to this field to indicate the number of allocated vectors Since The 10 GbE controller requests a single vector in the *Multiple Messages Capable* field, software is expected to write 000b to this field.
7	1b	RO	64-bit Capable. A value of 1b indicates that the 10 GbE controller is capable of generating 64-bit message addresses.
8	1b[1]	RO	MSI Per-vector Masking. A value of 0b indicates that the 10 GbE controller is not capable of per-vector masking. A value of 1b indicates that the 10 GbE controller is capable of per-vector masking. Exposed through the GLPCI_CAPSUP register.
15:9	0b	RO	Reserved. Reads as 0b

Notes:

1. Value loaded from the NVM.

38.43.2.4 Message Address Low Register (0x54; RW)

Written by the system to indicate the lower 32 bits of the address to use for the MSI memory write transaction. The lower two bits always return 0b regardless of the write operation.

38.43.2.5 Message Address High Register (0x58; RW)

Written by the system to indicate the upper 32 bits of the address to use for the MSI memory write transaction.

38.43.2.6 Message Data Register (0x5C; RW)

Written by the system to indicate the lower 16 bits of the data written in the MSI memory write Dword transaction. The upper 16 bits of the transaction are written as 0b.

38.43.2.7 Mask Bits Register (0x60; RW)

The Mask Bits and Pending Bits registers enable software to disable or defer message sending on a per-vector basis. Because the 10 GbE controller only supports one message, only bit 0 of these registers are implemented.

Bits	Default	RW	Description
0	0b	RW	MSI Vector 0 Mask. If set, the 10 GbE controller is prohibited from sending MSI messages.
31:1	0x0	RO	Reserved.

38.43.2.8 Pending Bits Register (0x64; RW)

Bits	Default	RW	Description
0	0b	RO	If set, the 10 GbE controller has a pending MSI message.
31:1	0x0	RO	Reserved.

38.43.3 MSI-X Capability

More than one MSI-X capability structure per function is prohibited while a function is permitted to have both an MSI and an MSI-X capability structure.

In contrast to the MSI capability structure, which directly contains all of the control/status information for the function's vectors, the MSI-X capability structure instead points to an MSI-X table structure and an MSI-X Pending Bit Array (PBA) structure, each residing in memory space.

A BAR is allocated for the MSI-X structures, described in Section 38.42.6. A BAR Indicator Register (BIR) indicates which BAR and a Qword-aligned offset indicates where the structure begins relative to the base address associated with the BAR. The BAR can be either 32-bit or is 64-bit.

The number of MSI-X vectors per PF (denoted as N) varies with the number of physical and virtual functions.

The MSI-X BAR is 32 KB long.

The location and size of the MSI-X vector table and the MSI-X pending bits table are determined as follows:

- MSI-X vector table:
  - The MSI-X table structure (Section 38.43.3.5) typically contains multiple entries, each consisting of several fields: *Message Address*, *Message Upper Address*, *Message Data*, and *Vector Control*. Each entry is capable of specifying a unique vector.
  - Starts at offset 0x0000 from start of BAR.
  - Contains the MSI-X vectors for the PF as well as the number of entries in table (N). The maximum value of N is 129 per PF (for the case of up to 4 PFs).
  - The vectors start with the Vector 0 (one per PF), followed by the other vectors allocated to the PF.
- MSI-X Pending Bits table:
  - The PBA structure [Section 38.43.3.5.2] contains the function's pending bits, one per table entry, organized as a packed array of bits within Qwords. The last Qword is not necessarily fully populated.

— Starts at offset 0x1000 (4 KB) from start of BAR.
— Contains the pending bits for the PF. The PF may be allocated a multiple of 64-bit registers. The total number of 32-bit registers is per the number of MSI-X vectors. The maximum number of registers is 6 (for the case of 129 vectors).
— The bits start with the Vector 0 bit (one per PF), followed by bits for the other vectors allocated to the PF.

To request service using a given MSI-X table entry, a function performs a Dword memory write transaction using:

- The contents of the *Message Data* field entry for data.
- The contents of the *Message Upper Address* field for the upper 32 bits of the address.
- The contents of the *Message Address* field entry for the lower 32 bits of the address.

A memory read transaction from the address targeted by the MSI-X message produces undefined results.

The MSI-X table and MSI-X PBA are permitted to co-reside within a naturally aligned 4 KB address range, though they must not overlap with each other.

38.43.3.1 Capability Structure

Byte Offset	Byte 3	Byte 2	Byte 1	Byte 0
0x70	Message Control (0x00090)		Next Pointer	Capability ID (0x11)
0x74	Table Offset			
0x78	PBA Offset			

Table 38-625 lists configuration sharing of the MSI-X capability registers among the different PCI functions.

Table 38-625.Configuration Sharing of the MSI-X Capability

Field	Sub-field	Shared?	Replicated?	Comments
Capability ID		x		
Next Pointer			x	
Message Control	Table Size	x		
Function Mask			x	
MSI-X Enable			x	
MSI-X Table Offset	Table BIR		x	
	Table Offset		x	
MSI-X Pending Bit Array	PBA BIR		x	
	PBA Offset		x	
MSI-X Table			x	
MSI-X PBA Structure			x	

38.43.3.2 Capability ID Register (0x70; RO)

This field equals 0x11 indicating that the linked list item as being the MSI-X registers.

38.43.3.3 Next Pointer Register (0x71; RO)

This field provides an offset to the next capability item in the capability list. See Table 38-622 for possible values of the next pointer register.

38.43.3.4 Message Control Register (0x72; RW)

Bits	Default	RW	Description
10:0	0x80 (129 vectors)	RO	Table Size. System software reads this field to determine the MSI-X Table Size N, which is encoded as N-1. N varies with the number of physical functions. This field is loaded from the NVM. It is reflected in the GLPCI_CNF2.MSI_X_PF_N CSR field.
13:11	0x0	RO	Always returns 0b on a read. A write operation has no effect.
14	0b	RW	Function Mask. If 1b, all of the vectors associated with the function are masked, regardless of their per-vector *Mask* bit states. If 0b, each vector's *Mask* bit determines whether the vector is masked or not. Setting or clearing the MSI-X *Function Mask* bit has no effect on the state of the per-vector *Mask* bits.
15	0b	RW	MSI-X Enable. If 1b and the MSI *Enable* bit in the MSI Message Control register is 0b, the function is permitted to use MSI-X to request service and is prohibited from using its INTx# pin. System configuration software sets this bit to enable MSI-X. A device driver is prohibited from writing this bit to mask a function's service request. If 0b, the function is prohibited from using MSI-X to request service.

38.43.3.4.1 MSI-X Table Offset Register (0x74; RW)

Bits	Default	RW	Description
2:0	0x3	RO	Table BIR. Indicates which one of a function's BARs, beginning at 0x10 in the configuration space, is used to map the function's MSI-X table into the memory space. While BIR values: 0...5 correspond to BARs 0x10:0x 24, respectively.
31:3	0x000	RO	Table Offset. Used as an offset from the address contained in one of the function's BARs to point to the base of the MSI-X table. The lower three *Table BIR* bits are masked off (set to 0b) by software to form a 32-bit Qword-aligned offset. Note that this field is read only.

38.43.3.4.2 MSI-X Pending Bit Array — PBA Offset (0x78; RW)

Bits	Default	RW	Description
2:0	0x3	RO	PBA BIR. Indicates which one of a function's BARs, beginning at 0x10 in the configuration space, is used to map the function's MSI-X PBA into the memory space. While BIR values: 0...5 correspond to BARs 0x10:0x 24, respectively.
31:3	0x200	RO	PBA Offset. Used as an offset from the address contained in one of the functions BARs to point to the base of the MSI-X PBA. The lower three PBA BIR bits are masked off (set to 0b) by software to form a 32-bit Qword-aligned offset. This field is read only.

38.43.3.5 PF MSI-X Table Structure

The MSI-X table is made of two structures:

- MSI-X Vector Table
- MSI-X Pending Bits Table

Both are described in the sections that follow.

38.43.3.5.1 MSI-X Vector Table

Dword3 — MSIXTVCTRL	Dword2 — MSIXTMSG	Dword1 — MSIXTUADD	Dword0 — MSIXTADD	Entry Number	BAR 3 — Offset
Vector Control	Msg Data	Msg Upper Addr	Msg Lower Addr	0	Base (0x0000)
Vector Control	Msg Data	Msg Upper Addr	Msg Lower Addr	1	Base + 1*16
Vector Control	Msg Data	Msg Upper Addr	Msg Lower Addr	2	Base + 2*16
…	…	…	…	…	
Vector Control	Msg Data	Msg Upper Addr	Msg Lower Addr	(N-1)	Base + (N-1)*16

38.43.3.5.2 MSI-X Pending Bits Table

Field	Bit(s)	Init Val.	Description
PENBIT	31:0	0x0	MSI-X Pending Bits. Each bit is set to 1b when the appropriate interrupt request is set and cleared to 0b when the appropriate interrupt request is cleared.

38.43.4 VPD Registers

The 10 GbE controller supports access to a VPD structure stored in the NVM using the following set of registers. A single VPD structure is provided, accessible through any of the physical functions.

Initial values of the configuration registers are marked in parenthesis.

Note: The VPD structure is available through all physical functions. As the interface is common to the functions, accessing the VPD structure of one function while an access to the NVM is in process on another function can yield to unexpected results.

Byte Offset	Byte 3	Byte 2	Byte 1	Byte 0
0xE0	VPD Address		Next Pointer	Capability ID (0x03)
0xE4	VPD Data			

38.43.4.1 Capability ID Register (0xE0; RO)

This field equals 0x3 indicating the linked list item as being the VPD registers.

38.43.4.2 Next pointer Register (0xE1; RO)

Offset to the next capability item in the capability list. See Table 38-622 for possible values of the next pointer register.

38.43.4.3 VPD Address Register (0xE2; RW)

Word-aligned byte address of the VPD area in the NVM to be accessed. The register is read/write, and the initial value at power-up is indeterminate.

Bits	Default	Rd/Wr	Description
14:0	X	RW	Address. Dword-aligned byte address of the VPD area in the NVM to be accessed. The register is read/write, and the initial value at power-up is indeterminate. The two LSBs are RO as zero.
15	0b	RW	F. A flag used to indicate when the transfer of data between the VPD Data register and the storage component completes. The Flag register is written when the VPD Address register is written. 0b = Read. Set by hardware when data is valid. 1b = Write. Cleared by hardware when data is written to the NVM. The VPD address and data should not be modified before the action is done.

38.43.4.4 VPD Data Register (0xE4; RW)

VPD read/write data.

Bits	Default	Rd/Wr	Description
31:0	X	RW	VPD Data. VPD data can be read or written through this register. The LSB of this register (at offset 4 in this capability structure) corresponds to the byte of VPD at the address specified by the VPD Address register. The data read from or written to this register uses the normal PCI byte transfer capabilities. Four bytes are always transferred between this register and the VPD storage component. Reading or writing data outside of the VPD space in the storage component is not allowed. In a write access, the data should be set before the address and the flag is set.

38.43.5 PCIe Capability Structure

The 10 GbE controller implements the PCIe capability structure linked to the legacy PCI capability list for endpoint devices as follows:

Byte Offset	Byte 3	Byte 2	Byte 1	Byte 0
0xA0	PCIe Capability Register (0x0002)		Next Pointer	Capability ID (0x10)
0xA4	Device Capability			
0xA8	Device Status		Device Control	
0xAC	Link Capability			
0xB0	Link Status		Link Control	
0xB4	Reserved			
0xB8	Reserved		Reserved	
0xBC	Reserved			
0xC0	Reserved		Reserved	
0xC4	Device Capability 2			
0xC8	Reserved		Device Control 2	
0xCC	Reserved			
0xD0	Link Status 2		Link Control 2	
0xD4	Reserved			
0xD8	Reserved		Reserved	

Table 38-626 lists configuration sharing of the PCIe Capability registers among the different PCI functions.

Table 38-626.Configuration Sharing of the PCIe Capability (Sheet 1 of 2)

Field	Sub-field	Shared?	Replicated?	Comments
Capability ID		X		
Next Pointer			X	
PCIe Capabilities		X		
Device Capabilities	Max Payload Size Supported	X		
	Phantom Functions Supported	X		Not supported.
	Extended Tag Field Supported	X		
	Endpoint L0s Acceptable Latency	X		
	Endpoint L1 Acceptable Latency	X		
	Function Level Reset Capability	X		
Device Control	Correctable Error Reporting Enable		X	
	Non-Fatal Error Reporting Enable		X	
	Fatal Error Reporting Enable		X	
	Unsupported Request Reporting Enable		X	
	Enable Relaxed Ordering		X	
	Max Payload Size		X	Use a minimum of all configured values. In ARI mode, use value in function 0.
	Extended Tag Field Enable		X	
	Aux Power PM Enable		X	Same policy for all PFs (logical OR of the PFs' bits).
	Enable No Snoop		X	
	Max Read Request Size		X	Use a minimum of all configured values.
	Initiate Function Level Reset		X	
Device Status	Correctable Detected		X	
	Non-Fatal Error Detected		X	
	Fatal Error Detected		X	
	Unsupported Request Detected		X	
	Aux Power Detected	X		
	Transactions Pending		X	
Link Capabilities	Supported Link Speeds	X		
	Max Link Width	X		
	Active State Link PM Support	X		
	L0s Exit Latency	X		
	L1 Exit Latency	X		
	Clock Power Management	X		

Table 38-626.Configuration Sharing of the PCIe Capability (Sheet 2 of 2)

Field	Sub-field	Shared?	Replicated?	Comments
	Port Number	X		
Link Control	Active State Link PM Control		X	Same policy for all PFs (logical AND of the PFs' bits). In ARI mode, use value in function 0.
	Read Completion Boundary (RCB)		X	
	Common Clock Configuration		X	Same policy for all PFs (logical AND of the PFs' bits). In ARI mode, use value in function 0.
	Extended Sync		X	Same policy for all PFs (logical OR of the PFs' bits).
Link Status	Current Link Speed	X		
	Negotiated Link Width	X		
	Slot Clock Configuration	X		
Device Capabilities 2	Completion Timeout Ranges Supported	X		
	Completion Timeout Disable Supported	X		
	TPH Completer Supported	X		
	Extended Fmt Field Supported	X		
	OBFF Supported	X		
Device Control 2	Completion Timeout Value		X	Completion timeout decision per PF or use the largest configured value among PFs.
	Completion Timeout Disable		X	Completion timeout mechanism enabled per PF.
	IDO Request Enable		X	
	IDO Completion Enable		X	
	OBFF Enable		X	PF0 only. RsvdP on other functions.
Link Capabilities 2		X		
Link Control 2		X		PF0 only. RsvdP on other functions.
Link Status 2		X		

38.43.5.1 Capability ID Register (0xA0; RO)

This field equals 0x10 indicating that the linked list item as being the PCIe Capabilities registers.

38.43.5.2 Next Pointer Register (0xA1; RO)

Offset to the next capability item in the capability list. See Table 38-622 for possible values of the next pointer register.

38.43.5.3 PCIe Capabilities Register (0xA2; RO)

The PCIe Capabilities register identifies PCIe device type and associated capabilities.

Bits	Default	RW	Description
3:0	0010b	RO	Capability Version. Indicates the PCIe capability structure version. The 10 GbE controller supports PCIe version 2 (loaded from the NVM). It is reflected in the GLPCI_CAPSUP register.
7:4	0000b/ 1001b	RO	Device/Port Type. Indicates the type of PCIe functions. All functions are native PCI functions with a value of 0000b if *strap_gbe_pci_endpoint_type* strap is set to native and are root complex integrated endpoint if set to RCIE.
8	0b	RO	Slot Implemented. The 10 GbE controller does not implement slot options. Therefore, this field is hardwired to 0b.
13:9	00000b	RO	Interrupt Message Number. This field is hardwired to 0x0 and is assumed to be irrelevant for endpoints.
15:14	00b	RO	Reserved.

38.43.5.4 Device Capabilities Register (0xA4; RO)

This register identifies the PCIe device specific capabilities.

Bits	Rd/Wr	Default	Description
2:0	RO	100b	Max Payload Size Supported. This field indicates the maximum payload that the 10 GbE controller can support for TLPs. Set according to the strap_gbe_imps strap. If strap_gbe_imps strap = 111b, then the default is 000b (128 bytes).
4:3	RO	00b	Phantom Function Supported. Not supported by the 10 GbE controller.
5	RO	1b	Extended Tag Field Supported. Maximum supported size of the *Tag* field. The 10 GbE controller supports an 8-bit *Tag* field for all functions.
8:6	RO	011b	Endpoint L0s Acceptable Latency. This field indicates the acceptable latency that the 10 GbE controller can withstand due to the transition from the L0s state to the L0 state. All functions share the same value loaded from the NVM. It is reflected in the *GLPCI_PMSUP* register. The default value of 011b denotes a maximum 512 ns.
11:9	RO	110b	Endpoint L1 Acceptable Latency. This field indicates the acceptable latency that the 10 GbE controller can withstand due to the transition from L1 state to the L0 state. The default value of 110b denotes a maximum of 64 µs. All functions share the same value loaded from the NVM. It is reflected in the *GLPCI_PMSUP* register.
12	RO	0b	Hardwired in the 10 GbE controller to 0b for all functions.
13	RO	0b	Hardwired in the 10 GbE controller to 0b for all functions.
14	RO	0b	Hardwired in the 10 GbE controller to 0b for all functions.
15	RO	1b	Role Based Error Reporting. Hardwired in the 10 GbE controller to 1b for all functions.
17:16	RO	000b	Reserved 0b.
25:18	RO	0x00	Slot Power Limit Value. Used in upstream ports only. Hardwired in the 10 GbE controller to 0x00 for all functions.
27:26	RO	00b	Slot Power Limit Scale. Used in upstream ports only. Hardwired in the 10 GbE controller to 0b for all functions.
28	RO	1b	Function Level Reset Capability – A value of 1b indicates the function supports the optional FLR mechanism.
31:29	RO	0000b	Reserved.

38.43.5.5 Device Control Register (0xA8; RW)

This register controls the PCIe specific parameters.

Bits	RW	Default	Description
0	RW	0b	Correctable Error Reporting Enable. Enable error report.
1	RW	0b	Non-Fatal Error Reporting Enable. Enable error report.
2	RW	0b	Fatal Error Reporting Enable. Enable error report.
3	RW	0b	Unsupported Request Reporting Enable. Enable error report.
4	RW	1b	Enable Relaxed Ordering. If this bit is set, the 10 GbE controller is permitted to set the *Relaxed Ordering* bit in the *Attribute* field of write transactions that do not need strong ordering.
7:5	RW	000b (128 bytes)	Max Payload Size. This field sets the maximum TLP payload size for the 10 GbE controller functions. As a receiver, the 10 GbE controller must handle TLPs as large as the set value. As a transmitter, the 10 GbE controller must not generate TLPs exceeding the set value. In ARI mode, *Max Payload Size* is determined solely by the field in function 0 (even when it is a dummy function) while it is meaningless in the other function(s).
8	RW	1b	Extended Tag Field Enable. The 10 GbE controller uses 8-bit tags when this bit is set and a 5-bit tag when disabled. If this bit is cleared in one of the functions, 5-bit tag is used by all functions.
9	RW	0b	Phantom Functions Enable. Not implemented in the 10 GbE controller.
10	RWS	0b	Aux Power PM Enable. When set, enables the 10 GbE controller to draw AUX power independent of PME AUX power. The 10 GbE controller is a multi-function device, therefore allowed to draw AUX power if at least one of the functions has this bit set.
11	RW	0b	Enable No Snoop. Hardwired to 0b as the 10 GbE controller never sets the no snoop attribute in a TLP.
14:12	RW	010b	Max Read Request Size. This field sets maximum read request size for the 10 GbE controller as a requester. 000b = 128 bytes. 001b = 256 bytes. 010b = 512 bytes. 011b = 1024 bytes. 100b = 2048 bytes. 101b = 4096 bytes 110b = Reserved. 111b = Reserved.
15	RW	0b	Initiate FLR. A write of 1b initiates FLR to the function. The value read by software from this bit is always 0b.

38.43.5.6 Device Status Register (0xAA; RW1C)

This register provides information about PCIe device specific parameters.

Bits	RW	Default	Description
0	RW1C	0b	Correctable Detected. Indicates status of correctable error detection.
1	RW1C	0b	Non-Fatal Error Detected. Indicates status of non-fatal error detection.
2	RW1C	0b	Fatal Error Detected. Indicates status of fatal error detection.
3	RW1C	0b	Unsupported Request Detected. Indicates that the 10 GbE controller received an unsupported request. This field is separate per PF. However, in case where an error cannot be associated with a PF, this bit is set in all PFs and VFs.
4	RO	0b	Aux Power Detected. If aux power is detected, this field is set to 1b. It is a strapping signal from the periphery and is identical for all functions. Resets on RSMRST# and PE_RST_N only.
5	RO	0b	Transactions Pending. Indicates whether the 10 GbE controller has ANY transactions pending. Transactions include completions for any outstanding non-posted request for all used traffic classes.
15:6	RsvdZ	0x00	Reserved.

38.43.5.7 Link Capabilities Register (0xAC; RO)

This register identifies PCIe link-specific capabilities.

A root complex integrated endpoint MUST not implement this register and return a value of zero when accessed.

If the gbe_pci_endpoint_type strap is set to RCIE, then this register is hardwired to zero, otherwise the description is as follows:

Bits	RW	Default	Description
3:0	RO	N/A	Max Link Speed. This field indicates the maximum Link speed of the associated port. The encoding is the binary value of the bit location in the supported link speeds vector (in the Link Capabilities 2 register) that corresponds to the maximum link speed. For example, a value of 0011b in this field indicates that the maximum link speed is that corresponding to bit 2 in the supported link speeds vector, which is 8.0 GT/s. Multi-function devices associated with an upstream port must report the same value in this field for all functions. When the gbe_soc strap is set, this field must be set to 0x1 (Gen1) by setting the GLPCI_LINKCAPLINK_SPEEDS_VECTOR field in the NVM to 000001b.
9:4	RO	0x08 (0x1 if gbe_soc strap is set)	Max Link Width. Indicates the maximum link width. The 10 GbE controller supports a x1, x4 and x8-link width. This field is loaded from the NVM and is reflected in the GLPCI_LINKCAP register. Defined encoding: 000000b = Reserved. 000001b = x1. 000010b = Reserved. 000100b = x4. 001000b = x8. When the gbe_soc strap is set, this field must be set to 0x1 (x1).

Bits	RW	Default	Description
11:10	RO	11b	Active State Link PM Support. Indicates the level of the active state of power management supported in the 10 GbE controller. Defined encodings are: 00b = No ASPM support. 01b = L0s entry supported. 10b = L1 supported. 11b = L0s and L1 supported. All functions share the same value loaded from the NVM. It is reflected in the GLPCI_PMSUP register. When the gbe_soc strap is set, this field must be set to 11b (Gen1).
14:12	RO	101b (000b if gbe_soc strap is set)	L0s Exit Latency. Indicates the exit latency from L0s to L0 state. 000b = Less than 64 ns. 001b = 64 ns – 128 ns. 010b = 128ns – 256 ns. 011b = 256 ns – 512 ns. 100b = 512 ns – 1 µs. 101b = 1 µs – 2 µs. 110b = 2 µs – 4 µs. 111b = Reserved. All functions share the same value loaded from the NVM. It is reflected in the GLPCI_PMSUP register. When the gbe_soc strap is set, this field must be set to 000b (less than 64 ns).
17:15	RO	100b (000b if gbe_soc strap is set)	L1 Exit Latency. Indicates the exit latency from L1 to L0 state. 000b = Less than 1 µs. 001b = 1 µs – 2 µs. 010b = 2 µs – 4 µs. 011b = 4 µs – 8 µs. 100b = 8 µs – 16 µs. 101b = 16 µs – 32 µs. 110b = 32 µs – 64 µs. 111b = More than 64 µs. All functions share the same value loaded from the NVM. It is reflected in the GLPCI_PMSUP register. When the gbe_soc strap is set, this field must be set to 000b (less than 1 µs).
18	RO	0b	Clock Power Management (not supported).
19	RO	0b	Surprise Down Error Reporting Capable. Hardwired to 0b.
20	RO	0b	Data Link Layer Link Active Reporting Capable.
21	RO	0b	Link Bandwidth Notification Capability. Hardwired to 0b (not applicable to endpoints).
22	HwInit	1b	ASPM Optionality Compliance. Software is permitted to use the value of this bit to help determine whether to enable ASPM or whether to run ASPM compliance tests.
23	RO	0b	Reserved.
31:24	HwInit	0x0	Port Number. The PCIe port number for the given PCIe link. This field is set in the link training phase. The field is set through the gbe_DeviceNumberN strap.

38.43.5.8 Link Control Register (0xB0; RO)

This register controls PCIe link specific parameters.

If the gbe_pci_endpoint_type strap is set to RCIE, then this register is hardwired to zero, otherwise the description is as follows:

Bits	RW	Default	Description
1:0	RW	00b	Active State Link PM Control. This field controls the active state PM enabled on the link. Link PM functionality is determined by the lowest common denominator of all functions. Defined encodings are: 00b = PM Disabled. 01b = L0s entry supported. 10b = L1 entry enabled 11b = L0s and L1 supported. In ARI mode, the ASPM is determined solely by the field in function 0 (even when it is a dummy function) while it is meaningless in the other function(s).
2	RsvdP	0b	Reserved.
3	RW	0b	Read Completion Boundary.
4	RO	0b	Link Disable. Reserved for endpoint devices. Hardwired to 0b.
5	RO	0b	Retrain Clock. Not applicable for endpoint devices. Hardwired to 0b.
6	RW	0b	Common Clock Configuration. When set, indicates that the 10 GbE controller and the component at the other end of the link are operating with a common reference clock. A value of 0b indicates that they are operating with an asynchronous clock. This parameter affects the L0s exit latencies. In ARI mode, the common clock configuration is determined solely by the field in function 0 (even when it is a dummy function) while it is meaningless in the other function(s).
7	RW	0b	Extended Sync. When set, this bit forces the transmission of additional ordered sets when exiting the L0s state and when in the recovery state. For multi-function devices, if any function has this bit set, then the component must transmit the additional ordered sets when exiting L0s or when in recovery
8	RO	0b	Enable Clock Power Management. Not supported - hardwired to 0b.
9	RO/RsvdP	0b	Hardware Autonomous Width Disable. When set to 1b, this bit disables hardware from changing the link width for reasons other than attempting to correct an unreliable link operation by reducing link width. Not supported - function 0 is hardwired to 0b (RO). Other functions are RsvdP.
10	RO	0b	Link Bandwidth Management Interrupt Enable. Not applicable to endpoints. Hardwired to 0b.
11	RO	0b	Link Autonomous Bandwidth Interrupt Enable. Not applicable to endpoints. Hardwired to 0b.
15:12	RsvdP	0x0	Reserved.

38.43.5.9 Link Status Register (0xB2; RO)

This register provides information about PCIe link specific parameters.

If the gbe_pci_endpoint_type strap is set to RCIE, then this register is hardwired to zero, otherwise the description is as follows:

Bits	RW	Default	Description
3:0	RO	Undefined	Current Link Speed. This field indicates the negotiated link speed of the given PCIe link. The encoded value specifies a bit location in the supported link speeds vector (in the Link Capabilities 2 register) that corresponds to the current link speed. For example, a value of 0010b in this field indicates that the current Link speed is that corresponding to bit 1 in the supported link speeds vector, which is 5.0 GT/s. Returns 001b (Gen1) if gbe_soc strap is set.
9:4	RO	Undefined	Negotiated Link Width. Indicates the negotiated width of the link. Relevant encodings for the 10 GbE controller are: 000001b = x1. 000010b = X2. 000100b = x4. 001000b = x8. Returns 000001b (x1) if gbe_soc strap is set.
10	RO	0b	Undefined.
11	RO	0b	Link Training. Indicates that link training is in progress. This field is not applicable, is reserved for endpoint devices, and is hardwired to 0b.
12	HwInit	1b	Slot Clock Configuration. When set, indicates that the 10 GbE controller uses the physical reference clock that the platform provides at the connector. This bit must be cleared if the 10 GbE controller uses an independent clock. The *Slot Clock Configuration* bit is loaded from the NVM. It is reflected in the GLPCI_PMSUP register. Must be set if gbe_soc strap is set.
13	RO	0b	Data Link Layer Link Active. Not supported in the 10 GbE controller. Hardwired to 0b.
14	RO	0b	Link Bandwidth Management Status. Not supported in the 10 GbE controller. Hardwired to 0b.
15	RO	0b	Link Autonomous Bandwidth Status. This bit is not applicable and is reserved for endpoints.

The following registers are supported only if the capability version is two and above.

38.43.5.10 Device Capabilities 2 Register (0xC4; RO)

This register identifies the PCIe device-specific capabilities.

Bits	RW	Default	Description
3:0	RO	0011b	Completion Timeout Ranges Supported. This field indicates the 10 GbE controller's support for the optional completion timeout programmability mechanism. Four time value ranges are defined: • Range A: 50 µs – 10 ms. • Range B: 10 ms – 250 ms. • Range C: 250 ms – 4 s. • Range D: 4 s – 64 s. Bits are set according to the following values to show the timeout value ranges that the 10 GbE controller supports. • 0000b = Completion timeout programming not supported. The 10 GbE controller must implement a timeout value in the range of 50 µs – 50 ms. • 0001b = Range A. • 0010b = Range B. • 0011b = Ranges A and B. • 0110b = Ranges B and C. • 0111b = Ranges A, B and C. • 1110b = Ranges B, C and D. • 1111b = Ranges A, B, C and D. • All other values are reserved.
4	RO	1b	Completion Timeout Disable Supported. ***Note:*** For dummy functionality, a completion timeout is not relevant as a dummy function because it never sends non-posted requests.
5	RO	0b	ARI Forwarding Supported. Applicable only to Switch Downstream Ports and Root Ports; must be 0b for other function types.
10:6	RO	0x00	Not supported - hardwired to 0x00
11	RO	0b	Reserved.
13:12	RO	00b	TPH Completer Supported - Value indicates Completer support for TPH or Extended TPH This capability is not supported.
17:14	RsvdP	0x0	Reserved
19:18	HWinit	00b	OBFF Supported. 00b = OBFF not supported. 01b = Reserved. 10b = Reserved. 11b = Reserved. Loaded from the NVM with a value of 00b. The value loaded from the NVM is reflected in the GLPCI_PMSUP register.
21:20	RO	0b	Reserved.
31:22	RsvdP	0x0	Reserved

38.43.5.11 Device Control 2 Register (0xC8; RW)

This register controls the PCIe specific parameters.

Bits	RW	Default	Description
3:0	RW	0x0	Completion Timeout Value. For devices that support completion timeout programmability, this field enables system software to modify the completion timeout value. Defined encodings: • 0000b = Default range: 50 µs to 50 ms. ***Note:*** It is strongly recommended that the completion timeout mechanism not expire in less than 10 ms. Values available if range A (50 µs to 10 ms) programmability range is supported: • 0001b = 50 µs to 100 µs. • 0010b = 1 ms to 10 ms. Values available if range B (10 ms to 250 ms) programmability range is supported: • 0101b = 16 ms to 55 ms. • 0110b = 65 ms to 210 ms. Values available if range C (250 ms to 4 s) programmability range is supported: • 1001b = 260 ms to 900 ms. • 1010b = 1 s to 3.5 s. Values available if the range D (4 s to 64 s) programmability range is supported: • 1101b = 4 s to 13 s. • 1110b = 17 s to 64 s. Values not defined are reserved. Software is permitted to change the value of this field at any time. For requests already pending when the completion timeout value is changed, hardware is permitted to use either the new or the old value for the outstanding requests and is permitted to base the start time for each request either on when this value was changed or on when each request was issued. Specifically, FLR clears this field to its default, so that completions are expected to return by the default time. ***Note:*** For dummy function, this field is RO - zero.
4	RW	0b	Completion Timeout Disable. When set to 1b, this bit disables the completion timeout mechanism. Software is permitted to set or clear this bit at any time. When set, the completion timeout detection mechanism is disabled. If there are outstanding requests when the bit is cleared, it is permitted but not required for hardware to apply the completion timeout mechanism to the outstanding requests. If this is done, it is permitted to base the start time for each request on either the time this bit was cleared or the time each request was issued. ***Note:*** For dummy function, this field is RO - zero.
5	RO	0b	ARI Forwarding Enable. Applicable only to switch devices.
7:6	RO	00b	Not supported - hardwired to 00b.
8	RW	0b	IDO Request Enable. If this bit is set, the function is permitted to set the ID-Based Ordering (IDO) bit (Attribute[2]) of Requests it initiates.
9	RW	0b	IDO Completion Enable. If this bit is set, the Function is permitted to set the ID-Based Ordering (IDO) bit (Attribute[2]) of completions it returns.
10	RO / RsvdP	0b	Reserved.
12:11	RO	0x0	Reserved.
14:13	RW / RsvdP	00b	OBFF Enable. 00b = Disabled. 01b = Enabled using message signaling [variation A]. 10b = Enabled using message signaling [variation B]. 11b = Enabled using WAKE# signaling.
15	RsvdP	0b	End-End TLP Prefix Blocking. Not applicable to endpoints (RsvdP).

38.43.5.12 Link Capabilities 2 Register (0xCC)

If the gbe_pci_endpoint_type strap is set to RCIE strap or the gbe_soc strap is set, then this register is hardwired to zero, otherwise the description is as follows:

Bits	RW	Default	Description
0	RsvdP	0b	Reserved
7:1	RO	0x01	Supported Link Speeds Vector. This field indicates the supported Link speed(s) of the associated port. For each bit, a value of 1b indicates that the corresponding link speed is supported; otherwise, the Link speed is not supported. Bit definitions are: Bit 1 = 2.5 GT/s. Bit 2 = 5.0 GT/s. Bit 3 = 8.0 GT/s. Bits 7:4 = RsvdP. This field is loaded from the NVM and reflected in the GLPCI_LINKCAP register.
8	RO	0b	Crosslink Supported. When set to 1b, this bit indicates that the associated port supports crosslinks. It is recommended that this bit be set in any port that supports crosslinks even though doing so is only required for ports that also support operating at 8.0 GT/s or higher Link speeds.
31:9	RsvdP	0x00	Reserved.

38.43.5.13 Link Control 2 Register (0xD0; RWS)

If the gbe_pci_endpoint_type strap is set to RCIE strap or the gbe_soc strap is set, then this register is hardwired to zero, otherwise the description is as follows:

Bits	RW	Default	Description
3:0	RWS (func 0) / RsvdP (else)	0011b (func 0) 0000b (else)	Target Link Speed. This field is used to set the target compliance mode speed when software is using the *Enter Compliance* bit to force a link into compliance mode. The encoding is the binary value of the bit in the Supported Link Speeds Vector (in the Link Capabilities 2 register) that corresponds to the desired target Link speed. All other encodings are Reserved. For example, 5.0 GT/s corresponds to bit 1 in the Supported Link Speeds Vector, so the encoding for a 5.0 GT/s target Link speed in this field is 0010b If a value is written to this field that does not correspond to a speed included in the *Supported Link Speeds* field, the result is undefined. The default value of this field is the highest link speed supported by the 10 GbE controller (as reported in the *Supported Link Speeds* field of the Link Capabilities register).
4	RWS (func 0) / RsvdP (else)	0b	Enter Compliance. Software is permitted to force a link to enter compliance mode at the speed indicated in the *Target Link Speed* field by setting this bit to 1b in both components on a link and then initiating a hot reset on the link. The default value of this field following a fundamental reset is 0b.
5	RWS (func 0) / RsvdP (else)	0b	Hardware Autonomous Speed Disable. When set to 1b, this bit disables hardware from changing the link speed for reasons other than attempting to correct unreliable link operation by reducing link speed.
6	RO	0b	Selectable De-Emphasis. This bit is not applicable and reserved for endpoints.

Bits	RW	Default	Description
9:7	RWS (func 0) / RsvdP (else)	000b	Transmit Margin. This field controls the value of the non de emphasized voltage level at the Transmitter pins. Encodings: 000b = Normal operating range. 001b = 800-1200 mV for full swing and 400-700 mV for half-swing. 010b = (n-1) — Values must be monotonic with a non-zero slope. The value of n must be greater than 3 and less than 7. At least two of these must be below the normal operating range of n: 200-400 mV for full-swing and 100-200 mV for half-swing. 111b= (n) reserved.
10	RWS (func 0) / RsvdP (else)	0b	Enter Modified Compliance. When this bit is set to 1b, the device transmits modified compliance pattern if the LTSSM enters Polling.Compliance state. The default value of this bit is 0b.
11	RWS (func 0) / RsvdP (else)	0b	Compliance SOS. When set to 1b, the LTSSM is required to send SOS periodically in between the (modified) compliance patterns. This bit is applicable when the Link is operating at 2.5 GT/s or 5 GT/s data rates only. The default value of this bit is 0b.
15:12	RWS (func 0) / RsvdP (else)	0x0	Compliance Preset/De-emphasis. For 8.0 GT/s Data Rate. This field sets the Transmitter Preset in Polling.Compliance state if the entry occurred due to the Enter Compliance bit being 1b. For 5.0 GT/s Data Rate. This field sets the de-emphasis level in Polling.Compliance state if the entry occurred due to the Enter Compliance bit being 1b. When the link is operating at 2.5 GT/s, the setting of this bit field has no effect. Defined encodings are: 0001b = -3.5 dB. 0000b = -6 dB. The default value of this field is 0000b.

38.43.5.14 Link Status 2 Register (0xD2; RW)

If the gbe_pci_endpoint_type strap is set to RCIE strap or the gbe_soc strap is set, then this register is hardwired to zero, otherwise the description is as follows:

Bits	RW	Default	Description
0	RO	0b	Current De-emphasis Level. When the link is operating at 5 GT/s speed, this bit reflects the level of de-emphasis. it is undefined when the link is not operating at 5.0 GT/s speed. Encodings: 1b = -3.5 dB. 0b = -6 dB. ***Note:*** Same value must be reported for all functions.
1	ROS/RsvdZ	0b	Equalization Complete. When set to 1b, this bit indicates that the transmitter equalization procedure has completed. ***Note:*** This bit must be implemented in function 0 and RsvdZ in other functions.
2	ROS/RsvdZ	0b	Equalization Phase 1 Successful. When set to 1b, this bit indicates that phase 1 of the transmitter equalization procedure has successfully completed. ***Note:*** This bit must be implemented in Function 0 and RsvdZ in other Functions.
3	ROS/RsvdZ	0b	Equalization Phase 2 Successful. When set to 1b, this bit indicates that phase 2 of the transmitter equalization procedure has successfully completed. ***Note:*** This bit must be implemented in function 0 and RsvdZ in other functions.
4	ROS/RsvdZ	0b	Equalization Phase 3 Successful. When set to 1b, this bit indicates that phase 3 of the transmitter equalization procedure has successfully completed. ***Note:*** This bit must be implemented in function 0 and RsvdZ in other functions.

Bits	RW	Default	Description
5	RW1C/ RsvdZ	0b	Link Equalization Request. This bit is set by hardware to request the link equalization process to be performed on the link. ***Note:*** This bit must be implemented in Function 0 and RsvdZ in other Functions.
15:6	RsvdZ	0x00	Reserved.

38.44 PCIe Extended Configuration Space

PCIe configuration space is located in a flat memory-mapped address space. PCIe extends the configuration space beyond the 256 bytes available for PCI to 4096 bytes. The 10 GbE controller decodes an additional four bits (bits 27:24) to provide the additional configuration space as shown. PCIe reserves the remaining four bits (bits 31:28) for future expansion of the configuration space beyond 4096 bytes.

The configuration address for a PCIe device is computed using a PCI-compatible bus, device, and function numbers as follows:

31 28	27 20	19 15	14 12	11 2	1 0
0000b	Bus #	Device #	Fun #	Register Address (offset)	00b

PCIe extended configuration space is allocated using a linked list of optional or required PCIe extended capabilities following a format resembling PCI capability structures. The first PCIe extended capability is located at offset 0x100 in the device configuration space. The first Dword of the capability structure identifies the capability/version and points to the next capability.

The 10 GbE controller supports the following PCIe extended capabilities:

Table 38-627.Extended Capabilities List

Address range	Item	Cases Where Capability Does Not Exist	Next Pointer
0x100 - 0x128	Advanced Error Reporting (AER)	None (always present)	Any of the below / 0x000
0x140 - 0x148	Device Serial Number	NVM is not valid	Any of the below / 0x000
0x150 - 0x154	Alternative RID Interpretation (ARI)	ARI Enabled bit in NVM is set to 0b	Any of the below / 0x000
0x160 - 0x19C	Single Root I/O Virtualization (SR-IOV)	- The global SR-IOV Enable bit in NVM is set to 0b (exposed via the *GLPCI_CAPSUP.IOV_EN* bit) - This is a dummy function - The per-PF SR-IOV Enable bit is set to 0b (*PF_VT_PFALLOC.VALID* is cleared)	Any of the below / 0x000
0x1A0 - 0x1A8	TPH Requester	- TPH Enabled bit in NVM is set to 0b - This is a dummy function	Any of the below / 0x000
0x1B0 - 0x1B4	Access Control Services (ACS)	- ACS Enabled bit in NVM is set to 0b - A single PF is enabled and SR-IOV is disabled	Any of the below / 0x000
0x1D0 - 0x1E8	Secondary PCI Express	- Secondary PCIe Enabled bit in NVM is set to 0b - Function is not function 0	0x000

38.44.1 Advanced Error Reporting Capability (AER)

The PCIe advanced error reporting capability is an optional extended capability to support advanced error reporting. The tables that follow list the PCIe advanced error reporting extended capability structure for PCIe devices.

Byte Offset	Byte 3	Byte 2	Byte 1	Byte 0
0x100	Next Capability Ptr.	Version (0x1)	AER Capability ID (0x0001)	
0x104	Uncorrectable Error Status			
0x108	Uncorrectable Error Mask			
0x10C	Uncorrectable Error Severity			
0x110	Correctable Error Status			
0x114	Correctable Error Mask			
0x118	Advanced Error Capabilities and Control Register			
0x11C... 0x128	Header Log			

Table 38-628 summarizes configuration sharing of the AER Capability registers among the different PCI functions.

Table 38-628.Configuration Sharing of the AER Capability

Field	Sub-field	Shared?	Replicated?	Comments
Enhanced Capability Header Register	Extended Capability ID	X		
	Capability Version	X		
	Next Capability Offset		X	
Uncorrectable Error Status			X	
Uncorrectable Error Mask			X	
Uncorrectable Error Severity			X	
Correctable Error Status			X	
Correctable Error Mask			X	
Advanced Error Capabilities and Control	First Error Pointer		X	
	ECRC Generation Capable	X		
	ECRC Generation Enable		X	ECRC insertion is per PF.
	ECRC Check Capable	X		
	ECRC Check Enable		X	
Header Log			X	

38.44.1.1 Advanced Error Reporting Enhanced Capability Header Register (0x100; RO)

Bit Location	Attribute	Default Value	Description
15:0	RO	0x0001	Extended Capability ID. PCIe extended capability ID indicating advanced error reporting capability.
19:16	RO	0x2	Version Number. PCIe advanced error reporting extended capability version number.
31:20	RO	See description	Next Capability Offset. Next PCIe extended capability offset. See Table 38-627 for possible values of the next capability offset.

38.44.1.2 Uncorrectable Error Status Register (0x104; RW1CS)

The Uncorrectable Error Status register reports error status of individual uncorrectable error sources on a PCIe device. An individual error status bit that is set to 1b indicates that a particular error occurred; software can clear an error status by writing a 1b to the respective bit. Register is cleared by RSMRST#.

Bit Location	Attribute	Default Value	Description
0	RO	0b	Reserved
3:1	RsvdZ	000b	Reserved
4	RW1CS	0b	Data Link Protocol Error Status.
5	RO	0b	Reserved
11:6	RsvdZ	0x00	Reserved
12	RW1CS	0b	Poisoned TLP Status.
13	RW1CS	0b	Flow Control Protocol Error Status.
14	RW1CS	0b	Completion Timeout Status.
15	RW1CS	0b	Completer Abort Status.
16	RW1CS	0b	Unexpected Completion Status.
17	RW1CS	0b	Receiver Overflow Status.
18	RW1CS	0b	Malformed TLP Status.
19	RW1CS	0b	ECRC Error Status.
20	RW1CS	0b	Unsupported Request Error Status.
21	RO	0b	ACS Violation Status. Not supported. Hardwired to 0b.
25:22	RO	0x0	Not supported
31:26	RsvdZ	0b	Reserved

38.44.1.3 Uncorrectable Error Mask Register (0x108; RWS)

The Uncorrectable Error Mask register controls reporting of individual uncorrectable errors by device to the host bridge via a PCIe error message. A masked error (respective bit set in mask register) is not reported to the host bridge by an individual device. Note that there is a mask bit per bit of the Uncorrectable Error Status register.

Bit Location	Attribute	Default Value	Description
0	RO	0b	Reserved
3:1	RsvdP	000b	Reserved
4	RWS	0b	Data Link Protocol Error Mask.
5	RO	0b	Reserved
11:6	RsvdP	0x00	Reserved
12	RWS	0b	Poisoned TLP Mask.
13	RWS	0b	Flow Control Protocol Error Mask.
14	RWS	0b	Completion Timeout Mask.
15	RWS	0b	Completer Abort Mask.
16	RWS	0b	Unexpected Completion Mask.
17	RWS	0b	Receiver Overflow Mask.
18	RWS	0b	Malformed TLP Mask.
19	RWS	0b	ECRC Error Mask.
20	RWS	0b	Unsupported Request Error Mask.
21	RO	0b	ACS Violation Mask. Not supported. Hardwired to 0b.
25:22	RO	0x0	Not supported
31:26	RsvdP	0b	Reserved

38.44.1.4 Uncorrectable Error Severity Register (0x10C; RWS)

The Uncorrectable Error Severity register controls whether an individual uncorrectable error is reported as a fatal error. An uncorrectable error is reported as fatal when the corresponding error bit in the severity register is set. If the bit is cleared, the corresponding error is considered non-fatal.

Bit Location	Attribute	Default Value	Description
0	RO	0b	Reserved
3:1	RsvdP	000b	Reserved
4	RWS	1b	Data Link Protocol Error Severity.
5	RO	0b	Reserved
11:6	RsvdP	0x00	Reserved
12	RWS	0b	Poisoned TLP Severity.
13	RWS	1b	Flow Control Protocol Error Severity.
14	RWS	0b	Completion Timeout Severity.
15	RWS	0b	Completer Abort Severity.
16	RWS	0b	Unexpected Completion Severity.

Bit Location	Attribute	Default Value	Description
17	RWS	1b	Receiver Overflow Severity.
18	RWS	1b	Malformed TLP Severity.
19	RWS	0b	ECRC Error Severity.
20	RWS	0b	Unsupported Request Error Severity.
21	RO	0b	ACS Violation Severity. Not supported. Hardwired to 0b.
25:22	RO	0x0	Not supported
31:26	RsvdP	0b	Reserved

38.44.1.5 Correctable Error Status Register (0x110; RW1CS)

The Correctable Error Status register reports error status of individual correctable error sources on a PCIe device. When an individual error status bit is set to 1b it indicates that a particular error occurred; software can clear an error status by writing a 1b to the respective bit. Register is cleared by RSMRST#.

Bit Location	Attribute	Default Value	Description
0	RW1CS	0b	Receiver Error Status.
5:1	RsvdZ	0b	Reserved
6	RW1CS	0b	Bad TLP Status.
7	RW1CS	0b	Bad DLLP Status.
8	RW1CS	0b	REPLAY_NUM Rollover Status.
11:9	RsvdZ	0b	Reserved
12	RW1CS	0b	Replay Timer Timeout Status.
13	RW1CS	0b	Advisory Non-Fatal Error Status.
15:14	RO	0b	Reserved
31:16	RsvdZ	0x00	Reserved

38.44.1.6 Correctable Error Mask Register (0x114; RWS)

The Correctable Error Mask register controls reporting of individual correctable errors by device to the host bridge via a PCIe error message. A masked error (respective bit set in mask register) is not reported to the host bridge by an individual device. There is a mask bit per bit in the Correctable Error Status register.

Bit Location	Attribute	Default Value	Description
0	RWS	0b	Receiver Error Mask.
5:1	RsvdP	0b	Reserved.
6	RWS	0b	Bad TLP Mask.
7	RWS	0b	Bad DLLP Mask.
8	RWS	0b	REPLAY_NUM Rollover Mask.
11:9	RsvdP	0b	Reserved.
12	RWS	0b	Replay Timer Timeout Mask.
13	RWS	1b	Advisory Non-Fatal Error Mask.

Bit Location	Attribute	Default Value	Description
15:14	RO	0b	Reserved.

38.44.1.7 Advanced Error Capabilities and Control Register (0x118; RO)

Bit Location	Attribute	Default Value	Description
4:0	ROS	0b	Vector pointing to the first recorded error in the Uncorrectable Error Status register. This is a read-only field that identifies the bit position of the first uncorrectable error reported in the Uncorrectable Error Status register.
5	RO	1b	ECRC Generation Capable. If set, this bit indicates that the function is capable of generating ECRC. This bit is loaded from NVM. It is reflected in the GLPCI_CAPSUP register.
6	RWS	0b	ECRC Generation Enable. When set, ECRC generation is enabled.
7	RO	1b	ECRC Check Capable. If set, this bit indicates that the function is capable of checking ECRC. This bit is loaded from NVM. It is reflected in the GLPCI_CAPSUP register.
8	RWS	0b	ECRC Check Enable. When set Set, ECRC checking is enabled.
9	RO	0b	Multiple Header Recording Capable. Not supported. hardwired to 0b
10	RO	0b	Multiple Header Recording Enable Not supported. hardwired to 0b
11	RsvdP	0b	TLP Prefix Log Present Not supported. hardwired to 0b
15:12	RsvdP	0x0	Reserved.

38.44.1.8 Header Log Register (0x11C:0x128; RO)

The header log register captures the header for the transaction that generated an error. This register is 16 bytes.

Bit Location	Attribute	Default Value	Description
127:0	ROS	0b	Header of the packet in error (TLP or DLLP).

38.44.2 Serial Number

The PCIe device serial number capability is an optional extended capability that can be implemented by any PCIe device. The device serial number is a read-only 64-bit value that is unique for a given PCIe device.

Serial Number capability is implemented for function 0; all other functions return the same device serial number value as that reported by function 0.

The capability is disabled when the MAC address in the GLPCI_SERL and GLPCI_SERH registers is 0x00...0 (indicating that the NVM is not valid and a proper MAC address was not loaded).

Byte Offset	Byte 3	Byte 2	Byte 1	Byte 0
0x140	Next Capability Ptr.	Version (0x1)	Serial ID Capability ID (0x0003)	
0x144	Serial Number Register (Lower Dword)			
0x148	Serial Number Register (Upper Dword)			

Table 38-629 summarizes configuration sharing of the Serial Number Capability registers among the different PCI functions.

Table 38-629.Configuration Sharing of the Serial Number Capability

Field	Sub-field	Shared?	Replicated?	Comments
Enhanced Capability Header Register	Extended Capability ID	X		
	Capability Version	X		
	Next Capability Offset		X	
Serial Number Register		X		See comment above

38.44.2.1 Device Serial Number Enhanced Capability Header Register (0x140; RO)

Bit(s)	Attribute	Default Value	Description
15:0	RO	0x0003	PCIe Extended Capability ID. This field is a PCI-SIG defined ID number that indicates the nature and format of the extended capability. The extended capability ID for the device serial number capability is 0x0003.
19:16	RO	0x1	Capability Version. This field is a PCI-SIG defined version number that indicates the version of the capability structure present. ***Note:*** Must be set to 0x1 for this version of the specification.
31:20	RO	See description	Next Capability Offset. This field contains the offset to the next PCIe capability structure or 0x000 if no other items exist in the linked list of capabilities. See Table 38-627 for possible values of the next capability offset.

38.44.2.2 Serial Number Registers (0x144:0x148; RO)

The Serial Number register is a 64-bit field that contains the IEEE defined 64-bit Extended Unique Identifier (EUI-64*). The register at offset 0x144 holds the higher 32 bits and the register at offset 0x148 holds the lower 32 bits. The following figure details the allocation of register fields in the Serial Number register. The table that follows provides the respective bit definitions.

Bit(s)	Attributes	Description
63:0	RO	PCIe Device Serial Number. This field contains the IEEE defined 64-bit EUI-64*. This identifier includes a 24-bit company ID value assigned by IEEE registration authority and a 40-bit extension identifier assigned by the manufacturer.

The serial number uses the Ethernet MAC address according to the following definition.

Field	Company ID			Extension Identifier				
Order	Addr+0	Addr+1	Addr+2	Addr+3	Addr+4	Addr+5	Addr+6	Addr+7
Most Significant Byte						Least Significant Byte		
Most Significant Bit						Least Significant Bit		

The serial number can be constructed from the 48-bit Ethernet MAC address in the following form.

Field	Company ID			MAC Label		Extension identifier		
Order	Addr+0	Addr+1	Addr+2	Addr+3	Addr+4	Addr+5	Addr+6	Addr+7
Most Significant Bytes						Least Significant Byte		
Most Significant Bit						Least Significant Bit		

In this case, the MAC label is 0xFFFF.

For example, assume that the company ID is (Intel) 00-A0-C9 and the extension identifier is 23-45-67 (MAC address of 00-A0-C9-23-45-67). In this case, the 64-bit serial number is:

Field	Company ID			MAC Label		Extension Identifier		
Order	Addr+0	Addr+1	Addr+2	Addr+3	Addr+4	Addr+5	Addr+6	Addr+7
	00	A0	C9	FF	FF	23	45	67
Most Significant Byte						Least Significant Byte		
Most Significant Bit						Least Significant Bit		

The Ethernet MAC address for the serial number capability is loaded from NVM (not the same field that is loaded from NVM into the Station MAC Address registers). It is reflected in the GLPCI_SERL and GLPCI_SERH registers. In the above example:

- GLPCI_SERL = C9-23-45-67
- GLPCI_SERH = 00-00-00-A0

Note: The official document that defines EUI-64* is: http://standards.ieee.org/regauth/oui/tutorials/EUI64.html

38.44.3 Alternate Routing ID Interpretation (ARI) Capability Structure

In order to allow more than eight functions per endpoint without requesting an internal switch, as is usually needed in virtualization scenarios, the PCI-SIG defines a new capability that allows a different interpretation of the *Bus*, *Device*, and *Function* fields. The capability is exposed when the GLPCI_CAPSUP.ARI_EN bit is set from NVM.

Note: This capability should not be exposed in root complex integrated endpoints and the GLPCI_CAPSUP.ARI_EN bit should be set accordingly.

The ARI capability structure is as follows:

Byte Offset	Byte 3	Byte 2	Byte 1	Byte 0
0x150	Next Capability Ptr.	Version (0x1)	ARI Capability ID (0x000E)	
0x154	ARI Control Register		ARI Capability Register	

Table 38-630 summarizes configuration sharing of the ARI Capability registers among the different PCI functions.

Table 38-630.Configuration Sharing of the ARI Capability

Field	Sub-Field	Shared?	Replicated?	Comments
Enhanced Capability Header Register	Extended Capability ID	X		
	Capability Version	X		
	Next Capability Offset		X	
ARI capability Register	Next Function Pointer		X	

38.44.3.1 PCIe ARI Header Register (0x150; RO)

Field	Bit(s)	Initial Value	Access	Description
ID	15:0	0x000E	RO	PCIe Extended Capability ID. PCIe extended capability ID for the alternative RID interpretation.
Version	19:16	1b	RO	Capability Version. This field is a PCI-SIG defined version number that indicates the version of the capability structure present. Must be 0x1 for this version of the specification.
Next Capability Offset	31:20	See description	RO	Next Capability Offset.This field contains the offset to the next PCIe extended capability structure. See Table 38-627 for possible values of the next capability offset.

38.44.3.2 PCIe ARI Capability Register (0x154; RO)

Field	Bit(s)	Initial Value	Access	Description
M	0	0b	RO	MFVC Function Groups Capability – Applicable only for Function 0; must be 0b for all other Functions. If 1b, indicates that the ARI Device supports Function Group level arbitration via its Multi-Function Virtual Channel (MFVC) Capability structure. Not supported in the 10 GbE controller.
A	1	0b	RO	ACS Function Groups Capability (A). Applicable only for function 0; must be 0b for all other functions. If 1b, indicates that the ARI device supports function group level granularity for ACS P2P Egress Control via its ACS capability structures. Not supported in the 10 GbE controller.
Reserved	7:2	0b	RsvdP	Reserved
NFN	15:8	See description[1]	RO	Next Function Number. This field contains the pointer to the next physical function configuration space or 0x0000 if no other items exist in the linked list of functions. Function 0 is the start of the link list of functions. Functions may be disabled during the Power-On-Reset flow (through strapping pins, SMASH/CLP commands, NC-SI commands) affecting this field.
M_EN	16	0b	RO	MFVC Function Groups Enable (M) – Applicable only for Function 0; must be hard wired to 0b for all other Functions. When set, the ARI Device must interpret entries in its Function. Arbitration Table as Function Group Numbers rather than Function Numbers. Not supported in the 10 GbE controller.

Field	Bit(s)	Initial Value	Access	Description
A_EN	17	0b	RO	ACS Function Groups Enable (A) – Applicable only for Function 0; must be hard wired to 0b for all other Functions. When set, each Function in the ARI Device must associate bits within its Egress Control Vector with Function Group Numbers rather than Function Numbers. Not supported in the 10 GbE controller.
Reserved	19:18	00b	RO	Reserved
FGN	22:20	0b	RO	Function Group Number. Not supported in the 10 GbE controller.
Reserved	31:23	0b	RsvdP	Reserved

Notes:

1. If function zero is a dummy function, this register should keep its attributes according to the function number. Disabled functions are skipped.

38.44.4 SR-IOV Capability Structure

This is a structure used to support the SR-IOV capabilities reporting and control. The capability is exposed when the GLPCI_CAPSUP.IOV_EN bit is set from NVM and the PF_VT_PFALLOC.VALID is set.

The following tables shows the implementation of this structure in the 10 GbE controller.

Byte Offset	Byte 3	Byte 2	Byte 1	Byte 0
0x160	Next Capability Offset	Version (0x1)	SR-IOV Capability ID (0x0010)	
0x164	SR-IOV Capabilities			
0x168	SR-IOV Status		SR-IOV Control	
0x16C	TotalVFs (RO)		Initial VF (RO)	
0x170	Reserved	Function Dependency Link (RO)	Num VF (RW)	
0x174	VF Stride (RO)		First VF Offset (RO)	
0x178	VF Device ID		Reserved	
0x17C	Supported Page Size (0x553)			
0x180	system page Size (RW)			
0x184	VF BAR0 — Low (RW)			
0x188	VF BAR0 — High (RW)			
0x18C	VF BAR2 (RO)			
0x190	VF BAR3 — Low (RW)			
0x194	VF BAR3- High (RW)			
0x198	VF BAR5 (RO)			
0x19C	VF Migration State Array Offset (RO)			

Table 38-631 summarizes configuration sharing of the SR-IOV Capability registers among the different PCI functions.

Table 38-631.Configuration Sharing of the SR-IOV Capability

Field	Sub-field	Shared?	Replicated?	Comments
Enhanced Capability Header Register	Extended Capability ID	X		
	Capability Version	X		
	Next Capability Offset		X	
SR-IOV Capabilities	VF Migration Capable	X		Not supported.
	ARI Capable Hierarchy Preserved		X	PF0 only. RO zero in all other functions.
	VF Migration Interrupt Message Number			Not supported.
SR-IOV Control	VF Enable		X	
	Memory Space Enable		X	
	ARI Capable Hierarchy	X	X	PF0 only. RO zero in all other functions.
InitialVFs			X	
TotalVFs			X	
Num VFs			X	
Function Dependency Link			X	Each PF indicates its PF number here.
First VF Offset			X	
VF Stride			X	
VF Device ID			X	
Supported Page Size		X		
System Page Size			X	
VF BARs			X	

38.44.4.1 PCIe SR-IOV Header Register (0x160; RO)

Field	Bit(s)	Initial Value	Access	Description
ID	15:0	0x0010	RO	PCIe Extended Capability ID. PCIe extended capability ID for the SR-IOV capability.
Version	19:16	0x1	RO	Capability Version. This field is a PCI-SIG defined version number that indicates the version of the capability structure present. Must be 0x1 for this version of the specification.
Next pointer	31:20	0x0	RO	Next Capability Offset. This field contains the offset to the next PCIe extended capability structure or 0x000 if no other items exist in the linked list of capabilities. See Table 38-627 for possible values of the next capability offset.

38.44.4.2 PCIe SR-IOV Capabilities Register (0x164; RO)

Field	Bit(s)	Initial Value	Access	Description
Reserved	0	0b	RO	Reserved.
ARI CHP	1	1b (lowest SR-IOV-enabled function) / 0b (else)	RO	ARI Capable Hierarchy Preserved. If set, the ARI *Capable Hierarchy* bit is preserved across certain power state transitions. Only present in lowest SR-IOV-enabled function. Read only zero in other PFs.
Reserved	31:2	0x0	RO	Reserved.

38.44.4.3 PCIe SR-IOV Control Register (0x168; RW)

Field	Bit(s)	Initial Value	Access	Description
VFE	0	0b	RW	VF Enable. VF Enable manages the assignment of VFs to the associated PF. If *VF Enable* is set to 1b, VFs must be enabled, associated with the PF, and exists in the PCIe fabric. When enabled, VFs must respond to and can issue PCIe transactions following all other rules for PCIe functions. If set to 0b, VFs must be disabled and not visible in the PCIe fabric; VFs cannot respond to or issue PCIe transactions. In addition, if *VF Enable* is cleared after having been set, all of the VFs must no longer: • Issue PCIe transactions • Respond to configuration space or memory space accesses. The behavior must be as if an FLR was issued to each of the VFs. Specifically, VFs must not retain any context after *VF Enable* has been cleared. Any errors already logged via PF error reporting registers, remain logged. However, no new VF errors must be logged after VF Enable is cleared.
VF ME	1	0b	RO	VF Migration Enable. Enables / Disables VF Migration Support. Not supported in the 10 GbE controller.
VF MIE	2	0b	RO	VF Migration Interrupt Enable. Enables / Disables VF Migration State Change Interrupt Not supported in the 10 GbE controller.
VF MSE	3	0b	RW	Memory Space Enable for Virtual Functions. VF MSE controls memory space enable for all VFs associated with this PF as with the Memory Space Enable bit in a functions PCI command register. The default value for this bit is 0b. When VF Enable is 1, virtual function memory space access is permitted only when VF MSE is Set. VFs shall follow the same error reporting rules as defined in the base specification if an attempt is made to access a virtual functions memory space when VF Enable is 1 and VF MSE is zero. **Implementation Note:** Virtual functions memory space cannot be accessed when VF Enable is zero. Thus, VF MSE is "don't care" when VF Enable is zero, however, software may choose to set VF MSE after programming the VF BARn registers, prior to setting VF Enable to 1.
VF ARI	4	0b	RW (lowest SR-IOV-enabled function) RO (else)	ARI Capable Hierarchy. Device can locate VFs in function numbers 8 to 255 of the captured bus number. If either ARI Capable Hierarchy Preserved is Set or No_Soft_Reset is Set, a power state transition of this PF from $D3_{HOT}$ to D0 does not affect the value of this bit
Reserved	15:5	0x0	RO	Reserved.

Field	Bit(s)	Initial Value	Access	Description
VFMIS	16	0b	RO	VF Migration Status. Indicates a VF Migration In or Migration Out Request has been issued by MR-PCIM. To determine the cause of the event, software may scan the VF State Array. Not implemented in the 10 GbE controller.
Reserved	31:17	0b	RO	Reserved

38.44.4.4 PCIe SR-IOV Initial/Total VFs Register (0x16C; RO)

Field	Bit(s)	Initial Value	Access	Description
InitialVFs	15:0	See Section 3 8.45.1.1	RO	InitialVFs indicates the number of VFs that are initially associated with the PF. If *VF Migration Capable* is cleared, this field must contain the same value as TotalVFs. In the 10 GbE controller this parameter is equal to the TotalVFs in this register.
TotalVFs	31:16	See Section 3 8.45.1.1	RO	TotalVFs defines the maximum number of VFs that can be associated with the PF. This field is derived from the PF_VT_PFALLOC.FIRSTVF and PF_VT_PFALLOC register fields loaded from NVM.

38.44.4.5 PCIe SR-IOV Num VFs Register (0x170; RW)

Field	Bit(s)	Initial Value	Access	Description
NumVFs	15:0	0x0	RW	Num VFs defines the number of VFs software has assigned to the PF. Software sets NumVFs to any value between one and the TotalVFs as part of the process of creating VFs. NumVFs VFs must be visible in the PCIe fabric after both NumVFs is set to a valid value and VF Enable is set to 1b.
FDL	23:16	0x0 (func 0)[1] 0x1 (func 1) ... 0xn (func n) ...	RO	Function Dependency Link. Defines dependencies between physical functions allocation. In the 10 GbE controller there are no constraints.
Reserved	31:24	0	RO	Reserved.

Notes:

1. Applies to dummy function as well.

38.44.4.6 PCIe SR-IOV VF RID Mapping Register (0x174; RO)

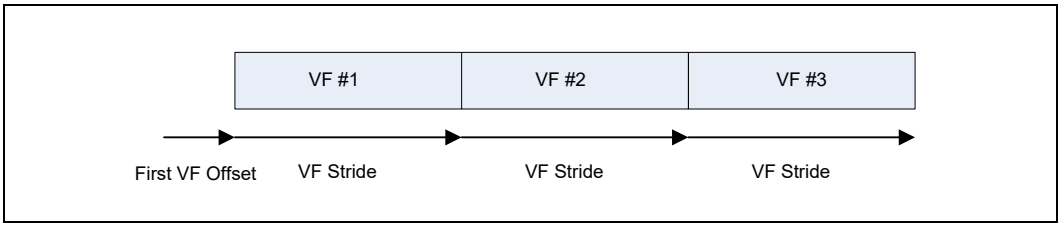
Field	Bit(s)	Initial Value	Access	Description
FVO	15:0	0x100 + 0x10 + FirstVF - PF#	RO	First VF offset defines the requester ID (RID) offset of the first VF that is associated with the PF that contains this capability structure. The first VFs 16-bit RID is calculated by adding the contents of this field to the RID of the PF containing this field. The content of this field is valid only when *VF Enable* is set. If *VF Enable* is 0b, the contents are undefined. If the *VF_ARI Enable* bit is set, this field changes to 0x10 + FirstVF - PF#. This field is derived from the PF_VT_PFALLOC.FIRSTVF register field loaded from NVM

Field	Bit(s)	Initial Value	Access	Description
VFS	31:16	0x1[1]	RO	VF stride defines the requestor ID (RID) offset from one VF to the next one for all VFs associated with the PF that contains this capability structure. The next VFs 16-bit RID is calculated by adding the contents of this field to the RID of the current VF. The contents of this field is valid only when *VF Enable* is set and *NumVFs* is non-zero. If *VF Enable* is 0b or if NumVFs is zero, the contents are undefined.

Notes:

1. See Section 38.45.1.1.

Figure 38-140.VF stride



38.44.4.7 PCIe SR-IOV VF Device ID Register (0x178; RO)

All Virtual functions have the same default value of 0x3A85, and can be auto-loaded from the NVM.

The DevID[15:7] field of a received Set ID Value sideband message is used to change the 9 MSBs of default value, but this value is overridden by the NVM loaded value.

The VF Device ID is loaded from the NVM according to the following rules:

- Device ID is loaded from NVM if the GLPCI_CAPSUP.LOAD_DEV_ID bit is set
- The Device ID value of all VFs associated with a given PF is loaded to the respective *PFPCI_DEVID.VF_DEV_ID* field

38.44.4.8 PCIe SR-IOV Supported Page Size Register (0x17C; RO)

Field	Bit(s)	Initial Value	Access	Description
Supported page Size	31:0	0x553	RO	For PFs that supports the stride-based BAR mechanism, this field defines the supported page sizes. This PF supports a page size of 2^(n+12) if bit n is set. For example, if bit 0 is Set, the Endpoint (EP) supports 4KB page sizes. Endpoints are required to support 4 KB, 8 KB, 64 KB, 256 KB, 1 MB and 4 MB page sizes. All other page sizes are optional.

38.44.4.9 PCIe SR-IOV System Page Size Register (0x180; RW)

Field	Bit(s)	Initial Value	Access	Description
Page size	31:0	0x1	RW	This field defines the page size the system uses to map the VFs' memory addresses. Software must set the value of the *System Page Size* to one of the page sizes set in the *Supported Page Sizes* field. As with *Supported Page Sizes*, if bit n is set in *System Page Size*, the VFs are required to support a page size of 2^(n+12). For example, if bit 1 is set, the system is using an 8 KB page size. The results are undefined if more than one bit is set in *System Page Size*. The results are undefined if a bit is set in *System Page Size* that is not set in *Supported Page Sizes*. When *System Page Size* is set, the VFs are required to align all BAR resources on a *System Page Size* boundary. Each BAR size, including *VF BARn Size* (described later) must be aligned on a *System Page Size* boundary. Each BAR size, including VF *BARn Size* must be sized to consume a multiple of *System Page Size* bytes. All fields requiring page size alignment within a function must be aligned on a *System Page Size* boundary. *VF Enable* must be zero when *System Page Size* is set. The results are undefined if *System Page Size* is set when *VF Enable* is set.

38.44.4.10 PCIe SR-IOV BAR 0 — Low Register (0x184; RW)

Field	Bit(s)	Initial Value	Access	Description
Mem	0	0b	RO	0b indicates memory space.
Mem Type	2:1	10b	RO	Indicates the address space size. 10b = 64-bit. This bit is loaded from the NVM. It is reflected in the GLPCI_VFSUP register
Prefetch Mem	3	0b	RO	0b = Non-prefetchable space. 1b = Prefetchable space. This bit is loaded from the NVM. It is reflected in the GLPCI_VFSUP register
Memory Address Space	31:4	0x0	RW	Which bits are RW bits and which are RO to 0x0 depend on the memory mapping window size.

38.44.4.11 PCIe SR-IOV BAR 0 — High Register (0x188; RW)

Field	Bit(s)	Initial Value	Access	Description
BAR0 — MSB	31:0	0x0	RW	MSB part of BAR0.

38.44.4.12 PCIe SR-IOV BAR 2 Register (0x18C; RO)

Field	Bit(s)	Initial Value	Access	Description
BAR2	31:0	0x0	RO	This BAR is not used.

38.44.4.13 PCIe SR-IOV BAR 3 — Low Register (0x190; RW)

Field	Bit(s)	Initial Value	Access	Description
Mem	0	0b	RO	0b indicates memory space.
Mem Type	2:1	10b	RO	Indicates the address space size. 10b = 64-bit. This bit is loaded from the NVM. It is reflected in the GLPCI_VFSUP register
Prefetch Mem	3	0b*	RO	0b = Non-prefetchable space 1b = Prefetchable space This bit is loaded from the NVM. It is reflected in the GLPCI_VFSUP register
Memory Address Space	31:4	0x0	RW	Which bits are RW bits and which are RO to 0x0 depend on the memory mapping window size. The size is a maximum between 16 KB and page size.

38.44.4.14 PCIe SR-IOV BAR 3 — High Register (0x194; RW)

Field	Bit(s)	Initial Value	Access	Description
BAR3 — MSB	31:0	0x0	RW	MSB part of BAR3.

38.44.4.15 PCIe SR-IOV BAR 5 Register (0x198; RO)

Field	Bit(s)	Initial Value	Access	Description
BAR5	31:0	0x0	RO	This BAR is not used.

38.44.4.16 PCIe SR-IOV VF Migration State Array Offset Register (0x19C; RO)

Field	Bit(s)	Initial Value	Access	Description
BIR	2:0	0x0	RO	Indicates which PF BAR contains the VF Migration State Array. Not implemented in the 10 GbE controller.
Offset	31:0	0x0	RO	Offset, relative to the beginning of the BAR of the start of the migration array. Not implemented in the 10 GbE controller.

38.44.5 TPH Requester Capability

The TPH Requester capability is an optional extended capability to support TLP Processing Hints. The capability is exposed when the GLPCI_CAPSUP.TPH_EN bit is set from NVM.

The following table lists the TPH extended capability structure for PCIe devices.

Byte Offset	Byte 3	Byte 2	Byte 1	Byte 0
0x1A0	PCI Express Extended Capability Header			
0x1A4	TPH Requester Capability Register			
0x1A8	TPH Requester Control Register			

Table 38-632 summarizes configuration sharing of the TPH Requester Capability registers among the different PCI functions.

Table 38-632.Configuration Sharing of the TPH Requester Capability

Field	Sub-field	Shared?	Replicated?	Comments
Enhanced Capability Header Register	Extended Capability ID	X		
	Capability Version	X		
	Next Capability Offset		X	
TPH Requester Capability		X		
TPH Requester Control			X	
TPH ST Table			X	The steering table upper fields are not supported.

38.44.5.1 TPH Requester Extended Capability Header (0x1A0; RO)

Bit Location	Attribute	Default Value	Description
15:0	RO	0x17	Extended Capability ID. PCIe extended capability ID indicating TPH capability.
19:16	RO	0x1	Capability Version. PCIe TPH extended capability version number.
31:20	RO	See Description	Next Capability Offset.This field contains the offset to the next PCIe capability structure. See Table 38-627 for possible values of the next capability offset.

38.44.5.2 TPH Requester Capability Register (0x1A4; RO)

Bit Location	Attribute	Default Value	Description
0	RO	1	No ST Mode Supported. If set indicates that the Function supports the No ST Mode of operation
1	RO	0	Interrupt Vector Mode Supported. Cleared to indicates that the 10 GbE controller does not support Interrupt Vector Mode of operation
2	RO	1	Device Specific Mode. Set to indicate that the 10 GbE controller supports Device Specific Mode of operation
7:3	RsvdP	0	Reserved.
8	RO	0	Extended TPH Requester Supported. Cleared to indicate that the function is not capable of generating requests with Extended TPH TLP Prefix
10:9	RO	00b	11b = Reserved.
15:11	RsvdP	0x0	Reserved.
26:16	RO	0x0	Reserved.
31:27	RsvdP	0x0	Reserved

38.44.5.3 TPH Requester Control Register (0x1A8; R/W)

Bit Location	Attribute	Default Value	Description
2:0	RW	0x0	Reserved.
7:3	RsvdP	0x0	Reserved
9:8	RW	0x0	TPH Requester Enable – Controls the ability to issue Request TLPs using either TPH or Extended TPH. Defined encodings are: 00b = The 10 GbE controller is not permitted to issue transactions with TPH or extended TPH as requester. 01b = The 10 GbE controller is permitted to issue transactions with TPH as requester and is not permitted to issue transactions with extended TPH as requester. 10b – Reserved. 11b = The 10 GbE controller is permitted to issue transactions with TPH and extended TPH as requester (the 10 GbE controller does not issue transactions with extended TPH). The default value of this field is 00b.
31:10	RsvdP	0x0	Reserved

38.44.6 ACS Extended Capability Structure

The ACS extended capability defines a set of control points within a PCIe topology to determine whether a TLP should be routed normally, blocked, or redirected. The capability is exposed when the GLPCI_CAPSUP.ACS_EN bit is set from NVM.

The ACS capability structure is shared and exposed to all PFs.

The following table lists the PCIe ACS extended capability structure for PCIe devices.

Byte Offset	Byte 3	Byte 2	Byte 1	Byte 0
0x1B0	PCI Express Extended Capability Header			
0x1B4	ACS Control Register (0x0)		ACS Capability Register (0x0)	

38.44.6.1 ACS Extended Capability Header (0x1B0; RO)

Bit Location	Attribute	Default Value	Description
15:0	RO	0x0D	PCI Express Extended Capability ID - PCIe extended capability ID indicating ACS capability.
19:16	RO	0x1	Capability Version - PCIe ACS extended capability version number.
31:20	RO	See Description	Next Capability Offset - See Table 38-627 for possible values of the next capability offset.

38.45 Virtual Functions

38.45.1 Overview

38.45.1.1 VF to PF Allocation

The 10 GbE controller supports up to 128 VFs. These VFs can be distributed arbitrarily among the different PFs. The distribution is done by NVM settings as the TotalVFs parameter should be stable at enumeration time.

For each of the potential 16 functions the following parameters are defined in the NVM:

- PCIe* Configuration Space Control 1.SR-IOV enable - should SR-IOV be exposed for this function
- SR-IOV configuration.Next PF First VF - what is the first VF allocated to the next function (out of 128). Can be any number between 0 and 127. VF0 is assumed to be assigned to the first function whose Next PF First VF field is not zero. Next PF First VF for the last PF indicates the last VF exposed.

From these parameters the following parameters are derived in the SR-IOV capability structure (see Section 38.44.4 for details):

- The SR-IOV structure is part of the configuration space of a PF only if the GLPCI_CAPSUP.IOV_EN bit is set in the NVM and PF_VT_PFALLOC.VALID field is set for this function.
- InitialVFs = TotalVFs = PF_VT_PFALLOC.LASTVF[n] - PF_VT_PFALLOC.FIRSTVF [n] +1
- First VF Offset = PF_VT_PFALLOC.FIRSTVF [n]+ 16 - PF# for ARI mode and PF_VT_PFALLOC.FIRSTVF [n]+ 272 - PF# for non ARI mode.

Note: The First VF offset formula is defined so that the RID of a VF is fixed no matter which PF it belongs to.

- VF stride = 1

The First VF and last VF allocated to a PF can be read from the *PF_VT_PFALLOC* registers.

The total number of enabled virtual functions is reflected in the GLGEN_PCIFCNCNT register.

38.45.1.2 Bus-Device-Function Layout

The requester ID allocation of the VF is done using the *First VF Offset* field and the *VF stride* in the IOV structure and is used to do the enumeration of the VFs.

38.45.1.2.1 ARI Mode

The ARI capability allows interpretation of the device part of the Requester ID as part of the function part. Thus a single device can span up to 256 functions.

The allocation of VFs to PF is flexible, there is no relationship between the PF RID and the associated VFs RID.

Table 38-633.RID Per VF - ARI Mode

VF#/PF#	B,D,F	Binary	Notes
PF 0	B,0,0	B,00000,000	PF #0
PF 1	B,0,1	B,00000,001	PF #1
PF 2	B,0,2	B,00000,010	PF #2
...	...	...	
PF 15	B,1,7	B,00001,111	PF #15
VF 0	B,2,0	B,00010,000	
VF 1	B,2,1	B,00010,001	
VF 2	B,2,2	B,00010,010	
...	...	...	
VF 127	B,17,7	B,10001,111	Last

38.45.1.2.2 Non-ARI Mode

When ARI is disabled, a non-zero PCI device number in the first bus can not be used, thus a second bus is needed to provide enough requester IDs. In this mode, we support only up to 8 physical functions and the RID layout is as follow:

Table 38-634.RID Per VF - Non ARI Mode

VF#/PF#	B,D,F	Binary	Notes
PF 0	B,0,0	B,00000,000	PF #0
PF 1	B,0,1	B,00000,001	PF #1
PF 2	B,0,2	B,00000,010	PF #2
...	...	...	
PF 7	B,0,7	B,00000,111	PF #7
VF 0	B+1,2,0	B+1,00010,000	
VF 1	B+1,2,1	B+1,00010,001	
VF 2	B+1,2,2	B+1,00010,010	
...	...	...	
VF 127	B+1,17,7	B+1,10001,111	Last

38.45.1.3 Configuration Space Overview

The configuration space reflected to each of the VF is a sparse version of the physical function configuration space. The following table describes the behavior of each register in the VF configuration space.

Table 38-635.VF PCIe Configuration Space (Sheet 1 of 2)

Section	Offset	Name	VF behavior	Notes
PCI Mandatory Registers	0	Vendor ID	RO — 0xFFFF	
	2	Device ID	RO — 0xFFFF	
	4	Command	Per VF	See Section 38.45.2.3.
	6	Status	Per VF	See Section 38.45.2.4.
	8	RevisionID	RO as PF	
	9	Class Code	RO as PF	
	C	Cache Line Size	RO — 0x0	
	D	Latency Timer	RO — 0x0	
	E	Header Type	RO — 0x0	
	F		RO — 0x0	
	10 — 27	BARs	RO — 0x0	Emulated by VMM.
	28	CardBus CIS	RO — 0x0	Not used.
	2C	Sub Vendor ID	RO as PF	
	2E	Sub System	RO.Same value for all VFs of each PF	See Section 38.45.2.5.
	30	Expansion ROM	RO — 0x0	Emulated by VMM.
	34	Cap Pointer	RO — 0x70	Next = MSI-X capability.
	3C	Int Line	RO — 0x0	
	3D	Int Pin	RO — 0x0	
	3E	Max Lat/Min Gnt	RO — 0x0	
MSI-X Capability	70	MSI-X Header	RO — 0xA011	Next = PCIe capability.
	72	MSI-x Message Control	per VF	See Section 38.45.3.1.1.
	74	MSI-X table Address	RO	See Section 38.45.3.1.2
	78	MSI-X PBA Address	RO	See Section 38.45.3.1.3
PCIe Capability	A0	PCIe Header	RO — 0x0010	Next = Last capability.
	A2	PCIe Capabilities	RO — as PF	
	A4	PCIe Dev Cap	RO — as PF	
	A8	PCIe Dev Ctrl	RW	As PF apart from FLR — See Table 38.45.3.2.1.
	AA	PCIe Dev Status	per VF	See Table 38.45.3.2.2.
	AC	PCIe Link Cap	RO — as PF	
	B0	PCIe Link Ctrl	RO — 0x0	
	B2	PCIe Link Status	RO — 0x0	
	C4	PCIe Dev Cap 2	RO — as PF	
	C8	PCIe Dev Ctrl 2	RO — 0x0	
	D0	PCIe Link Ctrl 2	RO — 0x0	
	D2	PCIe Link Status 2	RO — 0x0	

Table 38-635.VF PCIe Configuration Space (Sheet 2 of 2)

Section	Offset	Name	VF behavior	Notes
AER Capability	100	AER — Header	RO — 0x15010002	Next = ARI structure.
	104	AER — Uncorr Status	per VF	See Section 38.45.3.3.1.
	108	AER — Uncorr Mask	RO — 0x0	
	10C	AER — Uncorr Severity	RO — 0x0	
	110	AER — Corr Status	Per VF	See Section 38.45.3.3.2.
	114	AER — Corr Mask	RO — 0x0	
	118	AER — Cap/Ctrl	RO	See Section 38.45.3.3.3
	11C — 128	AER — Error Log	Shared two logs for all VFs	Same structure as in PF. In case of overflow, the header log is filled with ones.
ARI Capability	150	ARI — Header	0x1A01000E	Next = TPH structure.
	154	ARI — Cap/Ctrl	RO — 0X0	
TPH Requester capability	0x1A0	TPH - Header	0x1D010017	Next = ACS Structure.
	0x1A4	TPH - Capability	RO - 0x00000005	No table reported.
	0x1A8	TPH - Control	per VF	Same structure as in PF
ACS capability	0x1D0	ACS - Header	RO - 0x0001000D	Next = Last extended capability.
	0x1D4	ACS - Capability	RO - 0x00000000	

38.45.2 Mandatory Configuration Space

The IOV specification defines the configuration space of the Virtual functions as a mirror of the Physical function configuration space with the exception of some fields which are implemented per VF.

This section describes the expected handling of the different part of the configuration space for virtual functions. It deals only with the parts relevant to the 10 GbE controller and describes only changes which are not a trivial implementation of the specification.

38.45.2.1 Legacy PCI Configuration Space

The legacy configuration space is allocated to the PF only and emulated for the VFs. A separate set of BARs and one Bus master enable bit is allocated to the whole set of VFs.

All the legacy error reporting bits are emulated for the VF.

38.45.2.2 Memory BARs Assignment

The IOV specification defines a fixed stride for all the VF BARs, so that each VF can be allocated part of the memory BARs at a fixed stride from the a basic set of BARs. In this method only two decoders per replicated BAR per PF are required and the BARs reflected to the VF are emulated by the VMM.

The only BARs that are useful for the VFs are BAR0 and BAR3, thus only those are replicated.

The following table describes the BARs and the stride used for the VFs.

Table 38-636.VF BARs in Product Name

BAR	Type	Usage	Requested Size Per VF
0	Mem	CSR space	
1	Mem	High word of CSR space address	N/A
2	N/A	Not used	N/A
3	Mem	MSI-X	max(16K, page size)
4	Mem	High word of MSI-X space address	N/A
5	N/A	Not used	N/A

38.45.2.3 VF Command Register (0x4; RW)

Bit(s)	Initial Value	Rd/Wr	Description
0	0b	RO	IOAE: I/O Access Enable. RO as zero field.
1	0b	RO	MAE: Memory Access Enable. RO as zero field.
2	0b	RW	BME: Bus Master Enable.Disabling this bit prevents the associated VF from issuing any memory or I/O requests. Note that as MSI/MSI-X interrupt messages are in-band memory writes, disabling the bus master enable bit disables MSI/MSI-X interrupt messages as well. Requests other than memory or I/O requests are not controlled by this bit. ***Note:*** The state of active transactions is not specified when this bit is disabled after being enabled. The device can choose how it behaves when this condition occurs. Software cannot count on the device retaining state and resuming without loss of data when the bit is re-enabled. Transactions for a VF that has its *Bus Master Enable* set must not be blocked by transactions for VFs that have their *Bus Master Enable* cleared.
3	0b	RO	SCM: Special Cycle Enable. Hard wired to 0b
4	0b	RO	MWIE: MWI Enable. Hard wired to 0b.
5	0b	RO	PSE: Palette Snoop Enable. Hard wired to 0b.
6	0b	RO	PER: Parity Error Response. Zero for VFs.
7	0b	RO	WCE: Wait Cycle Enable. Hard wired to 0b.
8	0b	RO	SERRE: SERR# Enable. Zero for VFs.
9	0b	RO	FB2BE: Fast Back-to-Back Enable. Hard wired to 0b.
10	0b	RO	INTD: Interrupt Disable. Hard wired to 0b.
15:11	0b	RO	RSV: Reserved

38.45.2.4 VF Status Register (0x6; RW)

Bits	Initial Value	Rd/Wr	Description
2:0	0x0	RO	RSV: Reserved
3	0b	RO	IS: Interrupt Status. Hard wired to 0b.
4	1b	RO	NC: New Capabilities. Indicates that Product Name VFs implement extended capabilities. The 10 GbE controller VFs implement a capabilities list, to indicate that it supports MSI-X and PCIe extensions.
5	0b	RO	66E: 66 MHz Capable. Hard wired to 0b.
6	0b	RO	RSV: Reserved
7	0b	RO	FB2BC: Fast Back-to-Back Capable. Hard wired to 0b.
8	0b	RW1C	MPERR: Data Parity Reported.
10:9	00b	RO	DEVSEL: DEVSEL Timing. Hard wired to 0b.
11	0b	RW1C	STA: Signaled Target Abort.
12	0b	RW1C	RTA: Received Target Abort.
13	0b	RW1C	RMA: Received Master Abort.
14	0b	RW1C	SSERR: Signaled System Error.
15	0b	RW1C	DSERR: Detected Parity Error.

38.45.2.5 VF Subsystem ID (0x2E; RO)

This value is loaded from NVM if the *GLPCI_CAPSUP.LOAD_SUBSYS_ID* bit is set. Each VF is loaded from the respective PF's NVM *PFPCI_SUBSYSID.VF_SUB_ID* field (i.e., all VFs of a specific PF share the same value)

38.45.3 PCI and PCIe Capabilities

The following capability structures are partially replicated in VFs configuration space:

- PCIe capability structure.
- MSI-X capability structure

The following extended capability structures are partially replicated in VFs config space.

Table 38-637.Extended Capabilities List

Address range	Item	Cases where capability does not exist	Next Pointer
0x100 - 0x128	Advanced Error Reporting (AER)	None (always present)	Any of the below / 0x000
0x150 - 0x154	Alternative RID Interpretation (ARI)	ARI Enabled bit in NVM is set to 0b	Any of the below / 0x000
0x1A0 - 0x1A8	TPH Requester	TPH Enabled bit in NVM is set to 0b	Any of the below / 0x000
0x1B0 - 0x1B4	Access Control Services (ACS)	ACS Enabled bit in NVM is set to 0b	0x000

38.45.3.1 MSI-X Capability

The MSI-X BAR size is max(16K, page size).

The location and size of the MSI-X vector table and the MSI-X Pending Bits table are determined as follows:

- MSI-X vector table
  - The MSI-X table structure (Section 38.43.3.5) typically contains multiple entries, each consisting of several fields: *Message Address*, *Message Upper Address*, *Message Data*, and *Vector Control*. Each entry is capable of specifying a unique vector.
  - Starts at offset 0x0000 from start of BAR
  - Contains the MSI-X vectors for the VF as well as the number of entries in table (N). The maximum value of N is 17 per VF (for the case of up to 32 VFs)
  - The vectors start with the Vector 0 (one per VF), followed by the other vectors allocated to the VF
- MSI-X Pending Bits table
  - The PBA structure [Section 38.43.3.5.2] contains the function's pending bits, one per table entry, organized as a packed array of bits within Qwords. The last Qword is not necessarily fully populated
  - Starts at half the BAR size (default is offset 0x2000 - 8KB from start of BAR).
  - Contains the pending bits for the VF. The VF is allocated one 64-bit register for a maximum of 17 bits
  - The bits start with the Vector 0 bit (one per VF), followed by bits for the other vectors allocated to the VF

38.45.3.1.1 VF MSI-X Control Register (0x72; RW)

Bits	Initial Value	Rd/Wr	Description
10:0	0x004	RO	TS: Table Size (N-1). N varies with the number of virtual functions. This field is loaded from NVM. It is reflected in the GLPCI_CNF2.MSI_X_VF_N CSR field
13:11	0x0	RO	RSV: Reserved.
14	0b	RW	Mask: Function Mask.
15	0b	RW	En: MSI-X Enable.

38.45.3.1.2 MSI-X Address Register (0x74; RO)

Bits	Default	Type	Description
2:0	0x3	RO	Table BIR. Indicates which one of a function’s BARs, beginning at 0x10 in the configuration space, is used to map the function’s MSI-X table into the memory space. while BIR values: 0...5 correspond to BARs 0x10…0x 24 respectively.
31:3	0x000	RO	Table offset. Used as an offset from the address contained by one of the function’s BARs to point to the base of the MSI-X vectors address. The lower three BIR bits are masked off (set to zero) by software to form a 32-bit Qword-aligned offset.

38.45.3.1.3 MSI-X PBA Register (0x78; RO)

Bits	Default	Type	Description
2:0	0x3	RO	PBA BIR. Indicates which one of a function's BARs, located beginning at 0x10 in configuration space, is used to map the function's MSI-X PBA into memory space. A BIR value of three indicates that the PBA is mapped in BAR 3.
31:3	0x400	RO	PBA Offset. Used as an offset from the address contained by one of the function's BARs to point to the base of the MSI-X PBA. The lower three PBA BIR bits are masked off (set to zero) by software to form a 32-bit Qword-aligned offset. This value is changed by hardware to be half of the requested BAR size.

38.45.3.2 PCIe Capability Registers

The device control and device status registers have some fields which are specific per VF.

38.45.3.2.1 VF Device Control Register (0xA8; RW)

Bits	Rd/Wr	Default	Description
0	RO	0b	Correctable Error Reporting Enable. Zero for VFs.
1	RO	0b	Non-Fatal Error Reporting Enable. Zero for VFs.
2	RO	0b	Fatal Error Reporting Enable. Zero for VFs.
3	RO	0b	Unsupported Request Reporting Enable. Zero for VFs.
4	RO	0b	Enable Relaxed Ordering. Zero for VFs.
7:5	RO	0b	Max Payload Size. Zero for VFs.
8	RO	0b	Extended Tag field Enable.
9	RO	0b	Reserved.
10	RO	0b	Aux Power PM Enable. Zero for VFs.
11	RO	0b	Enable No Snoop. Zero for VFs.
14:12	RO	000b	Max Read Request Size. Zero for VFs.
15	RW	0b	Initiate Function Level Reset. Specific to each VF.

38.45.3.2.2 VF Device Status Register (0xAA; RO)

Bits	Rd/Wr	Default	Description
0	R/W1C	0b	Correctable Detected. Indicates status of correctable error detection.
1	R/W1C	0b	Non-Fatal Error Detected. Indicates status of non-fatal error detection.
2	R/W1C	0b	Fatal Error Detected. Indicates status of fatal error detection.
3	R/W1C	0b	Unsupported Request Detected. Indicates that the 10 GbE controller received an unsupported request. This field is separate per VF. However, in case where an error cannot be associated with a VF, this bit is set in all PFs and VFs.
4	RO	0b	Aux Power Detected. Zero for VFs.
5	RO	0b	Transactions Pending. Specific per VF. When set, indicates that a particular function (PF or VF) has issued non-posted requests that have not been completed. A function reports this bit cleared only when all completions for any outstanding non-posted requests have been received.
15:6	RO	0x00	Reserved

38.45.3.3 AER Registers

The following registers in the AER capability have a different behavior in a VF function.

Note that unlike the PF AER registers, these registers are not sticky since the VF is reset on FLR and on in-band reset.

38.45.3.3.1 Uncorrectable Error Status Register (0x104; RW1C)

Bit Location	Attribute	Default Value	Description
3:0	RO	0x0	Reserved
4	RO	0b	Data Link Protocol Error Status. Hardwired to 0b
5	RO	0b	Surprise Down Error Status. Hardwired to 0b
11:6	RO	0x0	Reserved
12	RW1C	0b	Poisoned TLP Status
13	RO	0b	Flow Control Protocol Error Status. Hardwired to 0b
14	RW1C	0b	Completion Timeout Status.
15	RW1C	0b	Completer Abort Status.
16	RW1C	0b	Unexpected Completion Status.
17	RO	0b	Receiver Overflow Status. Hardwired to 0b
18	RO	0b	Malformed TLP Status. Hardwired to 0b
19	RO	0b	ECRC Error Status. Hardwired to 0b
20	RW1C	0b	Unsupported Request Error Status — when caused by a function that claims a TLP.
21	RO	0b	ACS Violation Status. Hardwired to 0b
31:21	RO	0x0	Reserved

38.45.3.3.2 Correctable Error Status Register (0x110; RW1C)

The Correctable Error Status register reports error status of individual correctable error sources on a PCIe device. When an individual error status bit is set to 1b it indicates that a particular error occurred; software can clear an error status by writing a 1b to the respective bit.

Bit Location	Attribute	Default Value	Description
0	RO	0b	Receiver Error Status. Hardwired to 0b
5:1	RO	0x0	Reserved
6	RO	0b	Bad TLP Status. Hardwired to 0b
7	RO	0b	Bad DLLP Status. Hardwired to 0b
8	RO	0b	REPLAY_NUM Rollover Status. Hardwired to 0b
11:9	RO	0x0	Reserved
12	RO	0b	Replay Timer Timeout Status. Hardwired to 0b
13	RW1C	0b	Advisory Non-Fatal Error Status.
31:14	RO	0b	Reserved

38.45.3.3.3 Advanced Error Capabilities and Control Register (0x118; RO)

Bit Location	Attribute	Default Value	Description
4:0	ROS	0b	Vector pointing to the first recorded error in the Uncorrectable Error Status register. This is a read-only field that identifies the bit position of the first uncorrectable error reported in the Uncorrectable Error Status register.
5	RO	0b	ECRC Generation Capable. If set, this bit indicates that the function is capable of generating ECRC. This bit is loaded from NVM. It is reflected in the GLPCI_CAPSUP register
6	RO	0b	ECRC Generation Enable. When set, ECRC generation is enabled. Hardwired to 0b. The PF setting applies to the VF
7	RO	0b	ECRC Check Capable. If set, this bit indicates that the function is capable of checking ECRC. This bit is loaded from NVM. It is reflected in the GLPCI_CAPSUP register.
8	RO	0b	ECRC Check Enable.When set Set, ECRC checking is enabled. Hardwired to 0b. The PF setting applies to the VF
9	RO	0b	Multiple Header Recording Capable. Not supported. hardwired to 0b
10	RO	0b	Multiple Header Recording Enable Not supported. hardwired to 0b
11	RsvdP	0b	TLP Prefix Log Present Not supported. hardwired to 0b
15:12	RO	0x0	Reserved

38.46 Reliability, Diagnostics and Testability

38.46.1 ECC Support and ECC Error Flow

Memories in the 10 GbE controller are protected by ECC (ECC bits are added to the memory on write and compare on read). There are two types of ECC errors:

- Correctable ECC error — When the memory line has a single bit error, the ECC mechanism corrects it. This is done within the memory shell/wrapper and the flow continues as normal.
- Uncorrectable ECC error — When the memory line read has more than a single ECC error, it cannot be recovered by the ECC mechanism. The text that follows describes how the 10 GbE controller reacts to such an event.

Reporting of ECC errors is as follows:

- The GL_CRITERRMODMASK2 register masks the reporting of ECC errors per block.
  - — If enabled, a correctable error is indicates by the ITR_CAUSE_MEM_0_STATUS.ECC_FIX bit and an uncorrectable error is indicated by the ITR_CAUSE_MEM_0_STATUS.ECC_ERR bit.
- If the ECC_ENA.ECC_ENA is set to 1b, an uncorrectable ECC error sets the ECC_ERR interrupt cause (for all PFs).
- The *_ECC_COR_ERR and *_ECC_UNCOR_ERR per-block registers count the occurrence of correctable and uncorrectable errors, respectively (the * symbol stands for the block name).

Uncorrectable errors are handled as follows:

- Device blocks data from going to an external link and blocks any new PCIe master transaction.
- Recovery then depends on the memory where the error happens:
  - An error takes place in the core or global domains
    - If the GLGEN_RSTCTL.ECC_RST_EN bit is set to 1b, the 10 GbE controller generates a GLOBR reset
  - An error takes place in the EMP
    - If the GLMNG_WD_ENA.ECC_RST_ENA bit is set to 1b, the EMP generates an EMPR reset
  - An error takes place in FLEEP (Shadow RAM)
    - An ECC error check in the shadow RAM is enabled via the *Shadow RAM ECC Enable* bit in the NVM
    - The FLEEP reloads the shadow RAM from the NVM.
    - If the GLGEN_RSTCTL.ECC_RST_EN bit is set to 1b, the 10 GbE controller generates a GLOBR reset (which in turn reloads the relevant sections into hardware)
  - An error takes place in the PCIe domain
    - If an error is in data to be sent out, the TLP is sent with an EDB

    Else, the link goes to a link-down state
    - If the GLMNG_WD_ENA.ECC_RST_ENA bit is set to 1b, the EMP generates an EMPR reset
    - Else, if the GLGEN_RSTCTL.ECC_RST_EN bit is set to 1b, the 10 GbE controller generates a GLOBR reset

38.46.2 Link Loopback Operations

Loopback operations are supported by the 10 GbE controller to assist with system and device debug. Loopback operation can be used to test transmit and receive aspects of software device drivers, as well as to verify electrical integrity of the connections between the 10 GbE controller and the system (such as PCIe bus connections, etc.).

38.46.3 NVM Recovery Mode

NVM recovery mode is intended to recover from a misconfiguration, which can be caused by an interrupted firmware update, power failure, host software access or interrupted BMC configuration access. This misconfiguration prevents firmware from executing successful initialization flows. These flows enable the software device driver/ BMC to fix the misconfiguration and cause firmware to initialize the correct data path.

Note: Device firmware implements a recovery mode that requires driver and firmware update tool software assistance to complete the recovery process. For a full discussion of recovery mode, Refer to the *Intel® Ethernet Connection X722 Feature Support Matrix* Application Note.

38.47 Acronyms

The following table lists the acronyms used in this section.

Acronyms	Description
AEQ	Asynchronous Event Queue
AQP	Admin Queue Pair
ARQ	Admin Receive Queue
ASQ	Admin Send Queue
BAR	Base Address Register
BE	Big Endian
BMC	Baseboard Management Controller
CEQ	Completion Event Queue
CQ	Completion Queue
EMP	Embedded Management Processor
HMC	Host Memory Cache
IRD	Inbound RDMA Read Queue Depth
IRRQ	Inbound RDMA Read Queue
iWARP	Internet Wide Area RDMA Protocol
LE	Little Endian
LED	Light Emitting Diode
LQP	LAN Queue Pair
MAC	Media Access Controller
MDC	Mobile Daughter Card
MDIO	Management Data Input/Output
MPA	Marker PDU Aligned
MSI	Message Signaled Interrupt
MSS	Maximum Segment Size
NC-SI	Network Controlled Side-band Interface
NIC	Network Interface Card
ORD	Outbound RDMA Read Queue Depth
PF	Physical Function (in a virtualization context)
PXE	Pre-boot Execution Environment
QP	Queue Pair
RSS	Receive Side Scaling
SDP	Software Definable Pins
SMBus	System Management Bus
TC	Traffic Class
UDA	User-space Direct Access
VF	Virtual Function – A part of a PF assigned to a VI
VI	Virtual Image – A virtual machine to which a part of the I/O resources is assigned. Also known as a VM

Acronyms	Description
VM	Virtual Machine
VMDq	Virtual Machine Device queue
VMM	Virtual Machine Monitor
VSI	Virtual Station Interfaces
WQ	Work Queues
WQE	Work Queue Element
WSP	Weighted Strict Priority

§

39 Intel® QuickAssist Technology (Intel® QAT)

39.1 Acronyms

Acronyms	Description
3DES	Triple Data Encryption Standard (also TDES)
AES	Advanced Encryption Standard
AES-CCM	Counter Mode Cipher Block Chaining Message Authentication Code. Also CBC-MAC.
AES-GCM	Galois/Counter Mode
AES-XCBC	eXtended Ciphertext Block Chaining
AES-XTS	XEX based (**X**) Tweaked CodeBook mode (**T**CB) with ciphertext stealing (CT**S**) [**X**EX **T**CB CT**S**]
DES	Data Encryption Standard
ECDH	Elliptic Curve Diffie-Hellman
ECDSA	Elliptic Curve Digital Signature Algorithm
EPO	EndPoint Only - A PCH mode where the Intel® QuickAssist Technology and 10 GbE Controller can be used as standalone PCIE endpoints on a non-boot PCH
HMAC	Hash Message Authentication Code
KASUMI*	Block cipher used in UMTS, GSM, and GPRS mobile communication systems.
MD5	Message Digest algorithms: a widely used cryptographic hash function
QAT	Intel® QuickAssist Technology
RC4	Rivest Cipher 4: a software stream cipher
SHA	Secure Hash Algorithm. A cryptographic hash function.
SHA-1	See SHA. SHA-1 produces a 160-bit (20-byte) hash value typically 40 digit long hex number
SHA-2	Six digit hash functions that are 224, 256, 384, or 512 bits.
SHA-3	5x5 array of 64-bit words.
Snow3G	Word based synchronous stream cipher
TDES	Triple Data Encryption Standard (also 3DES)
ZUC	Stream cipher proposed for inclusion in the 4G LTE (Long Term Evolution)
SR-IOV	Single Root - I/O Virtualization
IQIA	Intel QuickAssist Integrated Accelerator
PF	Physical Function
VF	Virtual Function
VM	Virtual Machine
VMM	Virtual Machine Manager
ARI	Alternate Routing ID
MSI	Message Signaled Interrupt
AER	Advanced Error Reporting

39.2 References

None

39.3 Overview

Intel® QuickAssist Technology consists of an integrated accelerator for offload look-aside cryptographic and compression/decompression co-processing services. These services are accessible via Intel QuickAssist Technology related APIs that communicate via PCI configuration space access and assisted rings stored in system memory.

Intel QuickAssist Technology traffic only traverses across the PCH PCIe Upstream Ports and does not have DMI access. The PCH Upstream Ports are required to be connected to a PCIe Root Port for this feature to function. Refer to Section 39.6 for additional details on Upstream Port requirements.

Note: Intel QuickAssist Technology may not be supported on all PCH SKUs.

Note: The Intel® C620 Series Chipset SKUs that support Intel QuickAssist Technology require an Adaptive Voltage Identification (AVID) compatible VR solution (Vccprim_Avid of 0.85V to 1.0V) to be used.

39.4 Intel QuickAssist Technology Features

- This product contains Intel QuickAssist Technology hardware version 1.7
  - Intel QuickAssist Technology is supported on the boot PCH in normal mode.
  - Intel QuickAssist Technology is supported on a non-boot PCH in EndPoint Only (EPO) mode.
- Cryptographic Functions
  - Cipher Operations
    - Advanced Encryption Standard
    - Data Encryption Standard/Triple DES (3DES/TDES)
    - RC4
  - Hash Operation
    - SHA-1, MD5
    - SHA-2 (SHA-224, SHA-256, SHA-384, SHA-512)
    - SHA-3
    - ZUC
  - Authentication Operation
    - HMAC, AES-XCBC, AES-CCM, AES-GCM, AES-XTS
  - Cipher-Hash Combined Operation
  - Key Derivation Operation
  - Wireless Cryptography
    - KASUMI*, SNOW 3G*, ZUC
- Public Key Functions
  - RSA Operation
  - Diffie-Helman Operation
  - Digital Signature Standard Operation
  - Key Derivation Operation

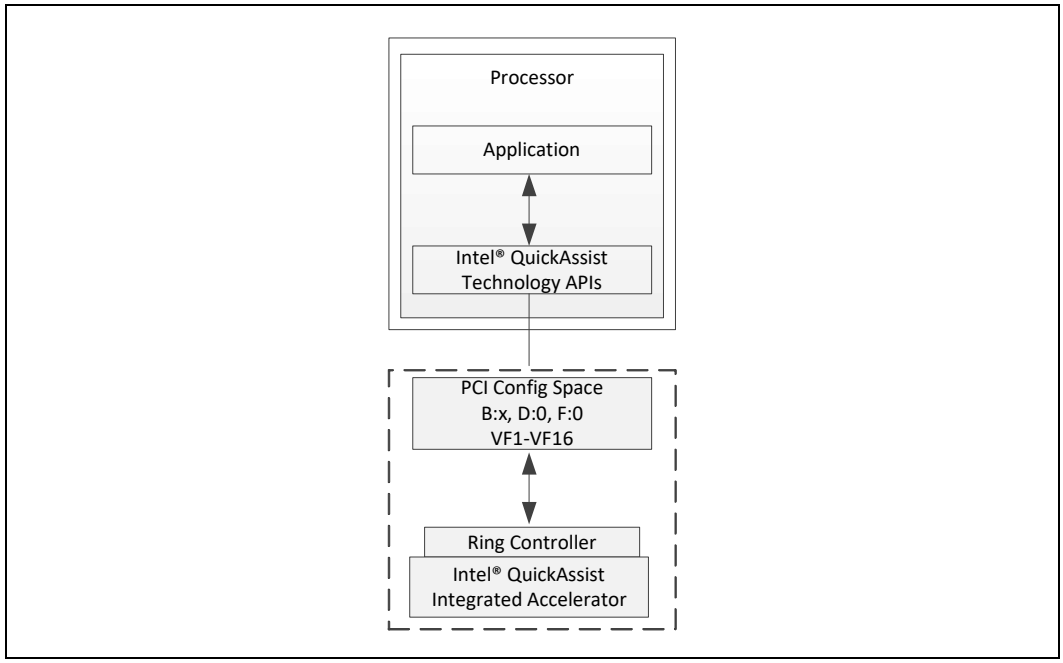
  - — Elliptic Curve Cryptography: ECDSA* and ECDH*
- Compression/Decompression Functions
  - — Deflate

39.5 Intel QuickAssist Integrated Accelerator

The IQIA provides acceleration functions that can be used by the IA cores. The IQIA appears to software as a set of up to three PCIe endpoints and its acceleration services can be accessed by a standard PCIe* driver. To enhance communication between the IA and the IQIA, the IQIA implements rings via a Ring Controller to route messages between them. This allows the IA to efficiently access the Intel QuickAssist Technology services through the Intel QuickAssist Technology APIs. The APIs communicate with the IQIA hardware via PCI configuration space access and assisted rings stored in system memory. Refer to the *Intel® QuickAssist Technology API Programmer's Guide*, incorporated herein by reference, and available at:

https://01.org/packet-processing/intel%C2%AE-quickassist-technology-drivers-and-patches

Figure 39-1. Intel QuickAssist Technology Usage Block Diagram



The following steps list a high-level overview of the usage.

1. The application requires IQIA service (e.g., decrypt a packet using AES). It requests service by calling an API with appropriate parameters.
2. The IQIA device driver formats a request message and places it onto ring-in-host memory, then notifies the IQIA.
3. IQIA reads the request message from the Ring Controller.
4. IQIA carries out the request.
5. When the process is done, write response message to ring-in-host memory.

6. Interrupt the host processor via a Message Signal Interrupt (MSI) or a user application can poll for completions.

Note: See the *Intel® QuickAssist Technology API Programmer's Guide* for information on Intel QuickAssist Technology APIs.

39.5.1 Ring Controller

The Ring Controller provides communication between the IA cores and the Intel QuickAssist Integrated Accelerator.

- Implements 256 rings for communications between the host processor and the IQIA.
- Organized in 16 bundles of 16 rings each.
- Each bundle supports one interrupt to the endpoint.

39.5.2 Single-Root I/O Virtualization

A virtualized platform enables system hardware resources to be shared among multiple guest operating systems. Each guest OS on the virtual platform operates under the assumption that it has complete control of all platform hardware resources. An I/O device that supports I/O virtualization provides the ability for it to be shared simultaneously among multiple guest operating systems.

The Intel QuickAssist Technology endpoint implements support for Single-Root I/O Virtualization for Function 0 and enables a usage model that allows up to 16 guest operating systems simultaneous access to its resources.

SR-IOV is a PCI-SIG* I/O Virtualization specification which in conjunction with system virtualization technologies like Intel® Virtualization Technology (Intel® VT) for Directed I/O (Intel® VT-d) and Intel® Virtualization Technology (Intel® VT) for IA-32, Intel® 64 and Intel® Architecture (Intel® VT-x), allows multiple guest operating systems running simultaneously within a single platform to natively share PCIe devices.

The SR-IOV capability structure is discovered and configured using the well-defined standard PCI enumeration process.

- Implements an SR-IOV extended capability record.
- Up to three Physical Functions available
  - One Physical Function per Intel QuickAssist Technology endpoint
  - Supports INTx, MSI, and MSI-X.
- Up to 48 virtual functions available
  - 16 Virtual Functions per Intel QuickAssist Technology endpoint
  - Supports ARI.
  - Implements MSI only.
  - *Does not* support INTx messages or MSI-X.
- Endpoint *does not* support the following optional IOV features:
  - Multi-root IOV
  - VF migration
- Ring access:
  - Physical Function can access all 16 ring bundles.

— Each Virtual Function can access one bundle.

Note: ARI must be enabled to use all of the Virtual Functions that Intel QuickAssist Technology supports.

39.5.2.1 I/O Virtualization Models

An I/O device such as the Integrated QuickAssist Accelerator can be shared amongst multiple guest operating systems in a virtual platform using several different hardware, software, and a combination of software and hardware techniques.

39.5.2.1.1 Sharing via Virtual Machine Manager

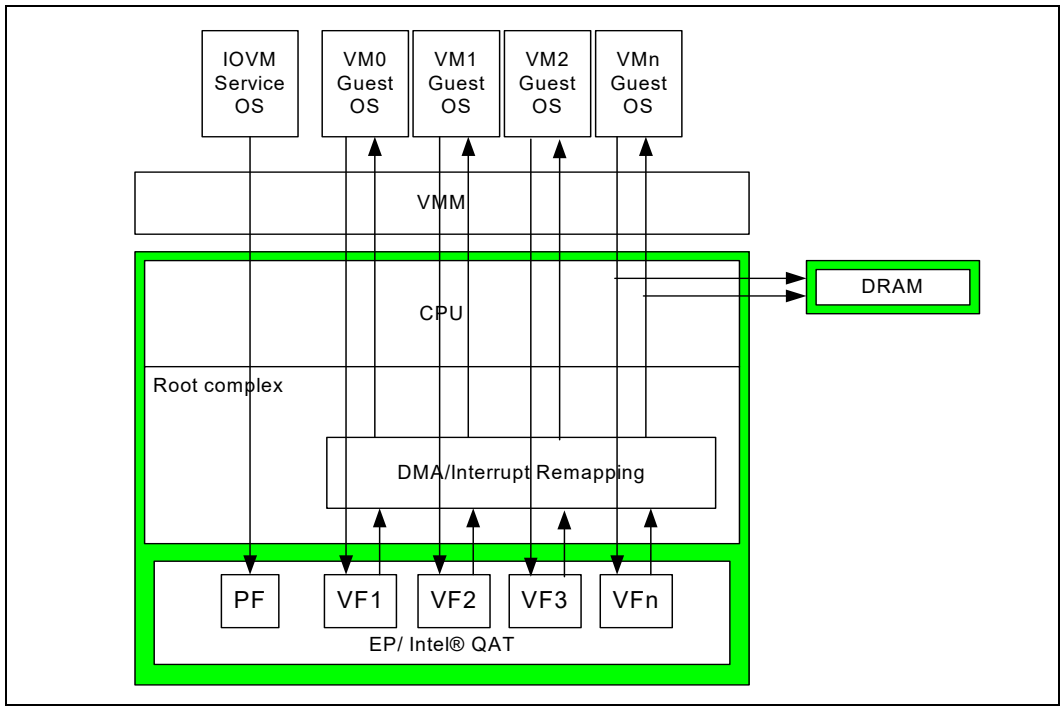
The I/O devices in this mode are assigned to the VMM/hypervisor. The VMM will expose a synthetic device to the Virtual Machines and could provide a paravirtualized driver that will run in the virtual machine. All accesses from the virtual machine using the paravirtualized driver are intercepted by the VMM. The VMM will then use the device driver to access the I/O device on behalf of the virtual machine. In this form of I/O sharing, there is little or no hardware support in the root complex or I/O device for virtualization.

39.5.2.1.2 Direct Assignment of the Endpoint with SR-IOV

The VMM can take advantage of the Intel QuickAssist Technology endpoint with SR-IOV support in the hardware and direct assign up to 16 virtual machines to the 16 virtual functions as shown in Figure 39-2. Each virtual machine can independently access the services of the Intel QuickAssist Technology.

Configuration and initialization of the IQIA will be through the I/O Virtual Machine. All error messages and interrupts not related to the ring status will be directed towards the IOVM (i.e., serviced by the IOVM).

Figure 39-2. Direct I/O Assignment Model with IOV Support



39.5.2.2 Ring Bundle Mapping

Each virtual function will be statically allocated one bundle (16 rings). A guest OS that is directly assigned to a virtual function will have access to only the rings associated with the bundle the virtual function is assigned to as shown in Table 39-1.

All errors and interrupts generated by the rings will use the Ring ID (RID) associated with that bundle.

Table 39-1. Ring Assignment to Intel QuickAssist Technology Virtual Functions

Ring	Virtual Function Assignment	Function Number	Ring CSR PCI Offset Begin
Bundle 0	VF1	8	00000h
Bundle 1	VF2	9	01000h
Bundle 2	VF3	10	02000h
Bundle 3	VF4	11	03000h
Bundle 4	VF5	12	04000h
Bundle 5	VF6	13	05000h
Bundle 6	VF7	14	06000h
Bundle 7	VF8	15	07000h
Bundle 8	VF9	16	08000h
Bundle 9	VF10	17	09000h
Bundle 10	VF11	18	0A000h
Bundle 11	VF12	19	0B000h
Bundle 12	VF13	20	0C000h
Bundle 13	VF14	21	0D000h
Bundle 14	VF15	22	0E000h
Bundle 15	VF16	23	0F000h

Notes:
1. The physical function can view and access all bundles
2. Interrupts and memory accesses generated by each of the bundles via ring commands will use the associated bundle's RID

39.5.2.3 Virtual Function Interrupts

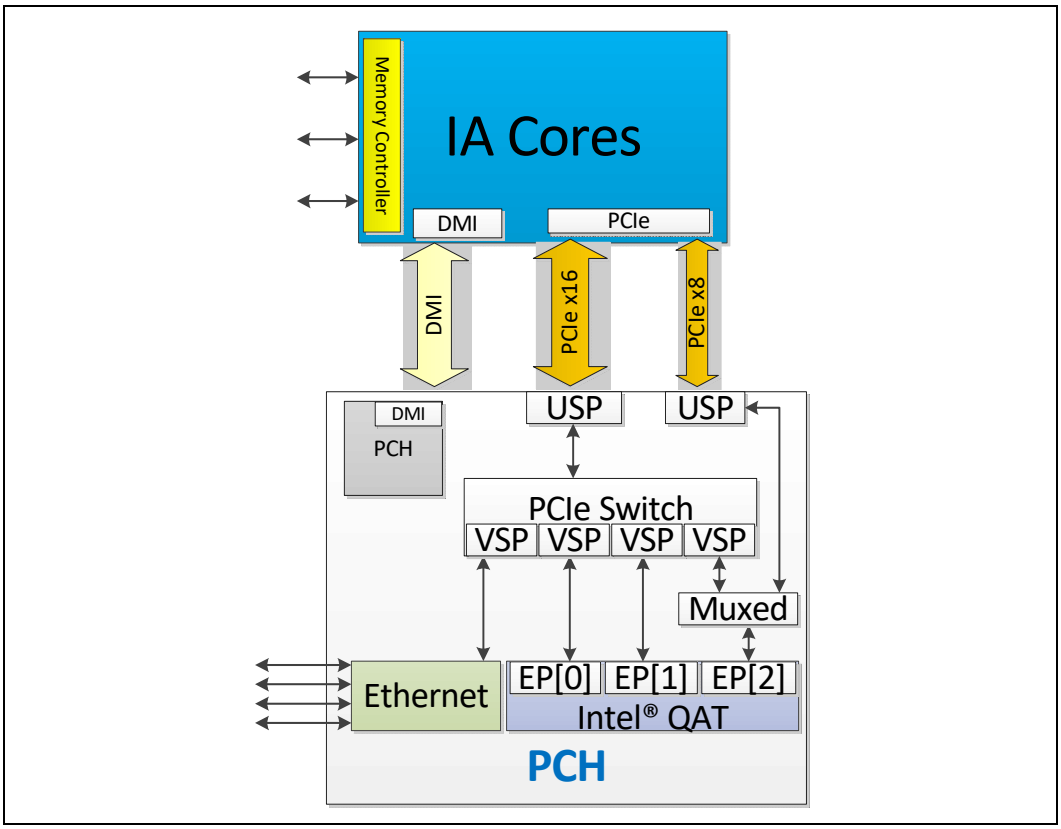
Each virtual function will support only MSI (one vector). A virtual function supports functional interrupts generated by the Ring Controller. INTx is not applicable for a virtual function per the PCI Express* SRIOV specification.

39.6 Intel QuickAssist Technology and PCH Upstream Port Requirements

Intel QuickAssist Technology requires the PCH Upstream Ports (USP) to be connected to a PCIe Root Port. Up to two Upstream Port controllers are supported and each controller consists of multiple lanes. The quantity of controllers and lanes available are SKU dependent. The primary upstream port controller is used by both the Intel QuickAssist Technology and the Integrated 10GbE and will support up to either x16 or x8 lanes depending on the PCH SKU. For PCH SKUs that support a secondary Upstream Port controller, an additional x8 lanes are available that are dedicated to Intel QuickAssist Technology Endpoint 2.

All three Intel QuickAssist Technology endpoints are accessible by the x16 Upstream Port controller via an integrated PCIe Switch and the switch's Virtual Switch Ports (VSP). On PCH SKUs that support the secondary x8 Upstream Port controller, the secondary upstream port controller is muxed with the Intel QuickAssist Technology endpoint 2 (EP[2]) and will have dedicated access to that endpoint (EP[2]) as shown in the following figure.

Figure 39-3. Intel QuickAssist Technology and PCH Upstream Ports



Note: The x8 lanes associated with the second upstream port controller are part of the high speed I/O (HSIO) muxed lanes along with SATA and PCIe Root Ports. When the x8 lanes are used as a second Upstream Port controller, the max number of PCH PCIe RP lanes available will be reduced by eight. Additionally, the PCH SATA ports will not be available. The PCH super speed SATA (sSATA) Ports are muxed separately, and will still be available.

Note: Traditional PCH features and interfaces such as SATA, USB, PCIe RP, and etc. are not accessible via the PCH PCIe Upstream Ports that are connected to a host processor. DMI is required for access to these capabilities.

39.7 Intel QuickAssist Technology Endpoint Only (EPO) Mode

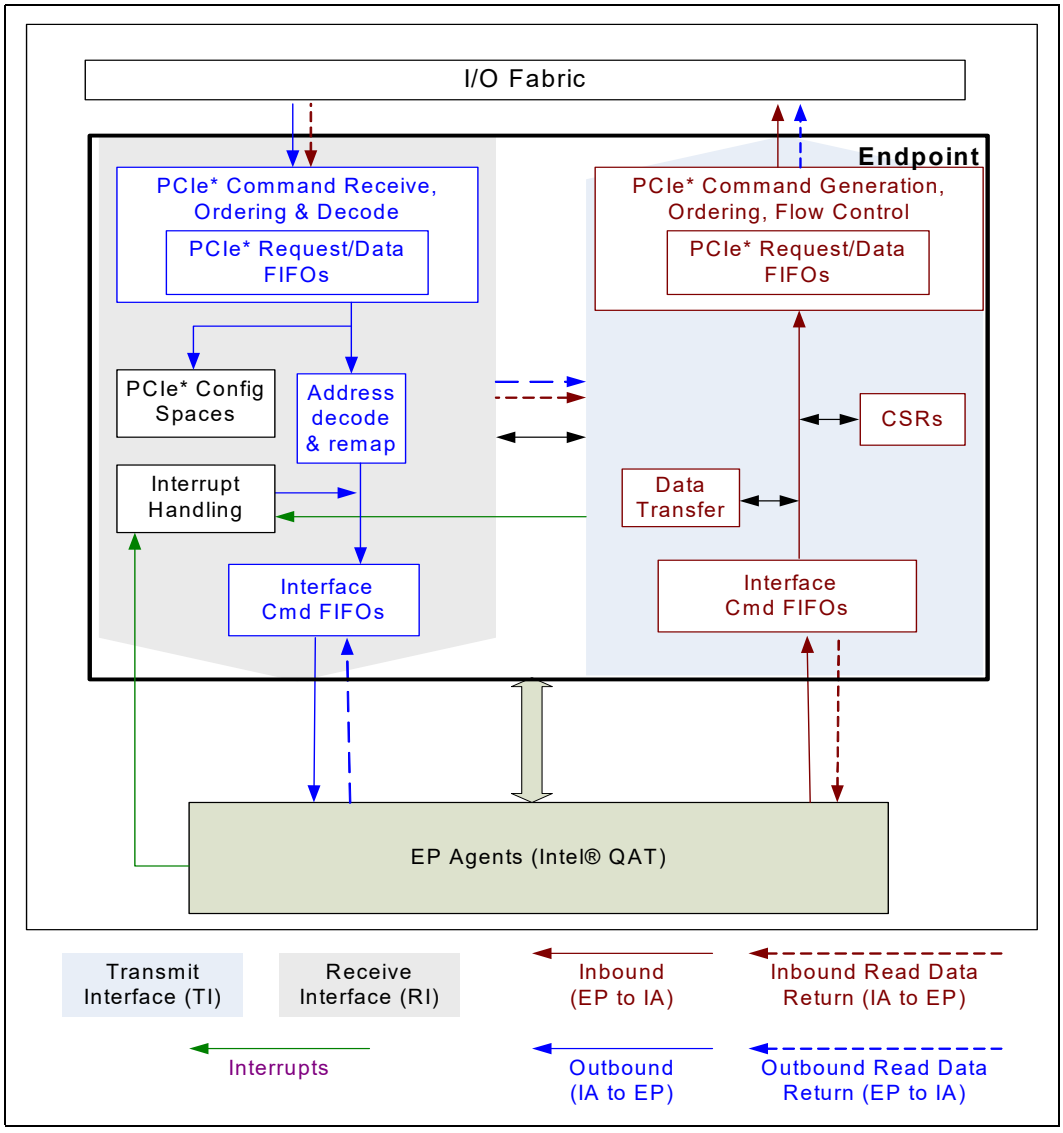
Intel QuickAssist Technology is supported in Endpoint Only Mode. Refer to the Section "PCH EndPoint Only Mode" in this document for additional details.

39.8 Intel QuickAssist Technology PCIe Endpoint Function

Intel QuickAssist Technology will show up as a PCIe device to the software running on the IA cores. The IQIA is integrated/embedded in the Root Complex as the root complex integrated endpoint.

Figure 39-4 shows the functional description block diagram of the PCIe endpoint interface.

Figure 39-4. Endpoint Functional Block Diagram



39.8.1 Transmit Interface

The Transmit Interface is responsible for converting the endpoint agents' commands toward system memory and outbound completions into the PCIe transactions. The Transmit Interface connects to the inbound port of the PCIe root complex.

39.8.2 Receive Interface

The Receive Interface is responsible for converting Outbound PCIe commands from the IA cores into the endpoint agents' commands. Downstream read completions for PCIe commands initiated by the Transmit Interface are also received by the Receive Interface.

39.8.2.1 PCIe Endpoint Functions

The Receive Interface connects to the Outbound port of the PCIe root complex. Table 39-2 lists the Receive Interface supported PCIe transactions. The endpoint only supports requests that are aligned on Dword address boundaries.

Table 39-2. PCIe Commands Supported by the Receive Interface

PCIe* Command	Description	Source	Endpoint Attributes	Comments
MRd32/ MRd64	Memory Read	IA reading EP MMIO region, PCIe* extended configuration space	4B	
MWr32/ MWr64	Memory Write	IA writing to EP MMIO region, PCIe extended configuration space	4-8B	Notes 2
CfgRd0	Type 0 Configuration Read	IA reading EP Configuration register	4B	
CfgWr0	Type 0 Configuration Write	Non-Posted IA writing to EP Configuration register	4B	Note 2
CmplD	Completion with Data	Read completion with data for a read command initiated by TI	4B-256B	Note 3

Notes:
1. Although the endpoint is capable of handling 8B requests, Intel QuickAssist Technology does not support any target that supports 8B requests. Therefore, software should not attempt any 8B requests.
2. This is a non-posted transaction. The Receive Interface will generate a completion without data after the configuration write is completed. The completion will be routed via a RI-TI private bus to the Transmit Interface. The Transmit Interface will initiate the PCIe completion packet.
3. The data will be pushed onto the bus by the Transmit Interface.
4. The Receive Interface will generate an Unsupported Request (UR) completion if it receives PCIe commands from IA that are not listed in this table.

39.8.2.2 Endpoint Function Mapping

The endpoint Receive Interface implements a single PCIe device with SR-IOV, and Table 39-3 shows the mapping of the endpoint functionality. When SR-IOV is enabled, the Receive Interface exposes additional virtual functions.

Table 39-3. PCIe Endpoint Function Mapping

Endpoint	Functional Blocks	PCIe Function Number (Note 1 and 2)	BARs	Interrupt Capability (Note 4)	PCIe Extended / PCI Capabilities (Note 2)
Intel QuickAssist Technology	Intel QuickAssist Technology	0	PMISCBAR, PETRINGCSRBAR	INTA, MSI, MSI-X with 16 vectors	MSI, MSI-X, PM, PCIe, AER, SR-IOV, ARI
VF[1:16]		8-23 Note 1	SRIOVBAR0, SRIOVBAR1	MSI only	MSI, PCIe, AER, ARI, ACS

Notes:
1. ARI must be enabled to view function numbers > 7, which are Virtual Functions.
2. When ARI is disabled, the Bus and Device Number is captured by the Receive Interface on every Type 0 configuration write. When ARI is enabled, the Device Number is implicitly assumed to be 0 and only the bus number is captured by the Receive Interface on every type 0 configuration write.
3. PPETRINGCSRBAR is used for access to registers that configure and access the Ring. PMISCBAR is used for accessing the endpoint and the MSI-X table.

39.8.2.3 Endpoint Mapping of MMIO to BARs

Table 39-4 shows the mapping of the endpoint MMIO regions to the BARs. Host access to the MMIO regions are done via the corresponding BARs. Refer to each PCIe function BAR definition for more details.

Table 39-4. Endpoint Mapping of MMIOs to BARs

Function #	BAR Name (Size)	Region Size	Functionality	Comments
0	PMISCBAR (256 KB)	248 KB	EP, Intel QuickAssist Technology	This region is mapped to the EP and Intel QuickAssist Technology CSRs
0		4 KB	EP MMIO CSRs	The EP CSRs are located in the EP.
0		4 KB	MSI-X Tables	The MSI-X tables are located in the EP
0	PETRINGCSRBAR	256 KB	16 Bundles of Ring CSRs	128 KB for all 16 bundles of CSRs and Intel QuickAssist Technology CSRs. Each bundle requires 4KB. ***Note:*** This space is allocated for 32 bundles. Only the lower 16 bundles are used and the upper 16 are memory holes in this region.
8-23 (VF[1-16])	SRIOVBAR0	MIN (4 KB, System Page Size) (Note 2)	One Ring Bundle per VF[x]	Each bundle requires 512B, but are mapped in a 4 KB window by default or the system page size.
	SRIOVBAR1	MIN (4 KB, System Page Size) (Note 2)	VF Messaging Registers	There is only one register per VF but is mapped in a 4 KB window by default or the system page size.

Notes:
1. Accesses to the MMIO region that are not implemented within the regions defined in this table will return zeros (0s). Accesses to MMIO regions that are not claimed by any of the BARs will return Unsupported Request (UR).
2. The SRIOVBAR0 size is dependent on the value programmed in the system page size.
3. The SRIOVBAR1 size is dependent on the value programmed in the system page size.

39.8.2.4 Alternate Routing-ID

The Receive Interface provides support for the Alternate Routing-ID Interpretation. When enabled, ARI allows the PCIe device to implement more than eight functions.

An ARI device interprets the Routing, Requester, and Completer IDs as having an 8-bit function number instead of the traditional 3-bit function number. An ARI device has its associated device number implied to be 0 rather than specified by an ID field. The Receive Interface supports ARI for Function 0 when SR-IOV is enabled.

Note: ARI must be enabled to use all of the Virtual Functions that Intel QuickAssist Technology supports.

Table 39-5. ARI: Bus, Device and Function Interpretation

ARI Disabled	**Bus Number**	**Device Number**	**Function Number**
	[7:0]	[4:0]	[2:0]
ARI Enabled	**Bus Number**	**Function Number**	
	[7:0]	[7:0]	

39.9 Interrupts

The Receive Interface is responsible for collating interrupts and errors from the endpoint and taking the appropriate actions based on the PCIe configurations status.

Interrupts from the endpoint can be configured in the Receive Interface to be routed to the IA cores. The interrupts can be sent upstream as either an MSI, MSI-X, or by using INTx virtual wire interrupt signaling mechanism (which are messages on PCIe).

39.9.1 INTx

PCIe supports devices that need to use the PCI compatible legacy interrupt signaling: INT[A, B, C, D]. Instead of dedicated pins for this functionality, PCIe supports in-band virtual wires.

The endpoint supports the generation of INTx virtual wire interrupt signaling based on internal generated interrupts. The INTx notification is accomplished using PCIe messages.

The Receive Interface is responsible for collating interrupt sources from the endpoint agents and generating the transaction that is translated to a PCIe interrupt message to the IA. Upon detecting an interrupt from an internal source, the Receive Interface first determines whether to generate an MSI, MSI-X, or INTx message.

39.9.2 Message Signaled Interrupt

Message Signaled Interrupts are an optional feature that enables a device function to request service by writing a system-specified data value to a system specified address (using a PCI Dword memory write transaction). System software initializes the message address and message data during device configuration. Each function in the endpoint supports the MSI capability. All the interrupts per function share one vector.

39.9.3 MSI-X

MSI-X defines a separate optional extension to the basic MSI functionality. Compared to MSI, MSI-X supports a larger number of vectors per function, the ability for the OS to program independent address and data values for each vector. The address and data values for each vector is specified by a table that resides in the memory space. MSI-X is supported only for the physical function.

Once an MSI has been generated using MSI-X, its characteristics are similar to an MSI that was generated using an MSI capability record. However, MSI-X supports per-vector masking that is optional for MSI. MSI-X also supports a Function Mask bit, which when set, masks all of the vectors associated with a function. Per-vector masking is managed through a Mask and Pending bit pair per the MSI vector or MSI-X table entry. An MSI vector is masked when its associated Mask bit is set. An MSI-X vector is masked when its associated MSI-X Table Entry Mask bit or the MSI-X Function Mask bit is set. While a vector is masked, the function is prohibited from sending the associated message, and the function must set the associated Pending bit whenever the function would otherwise send the message. When software unmasks a vector whose associated Pending bit is set, the function must schedule sending the associated message, and clear the Pending bit as soon as the message has been sent.

When MSI-X is enabled, the endpoint generates an interrupt using a PCIe posted memory write transaction. The address and data of that transaction are determined by the system and programmed in the PCI MSI-X table entry.

39.10 Endpoint Errors

There are two classes of errors that the endpoint will encounter:

- Errors that occur due to PCIe transactions
  - PCIe error management defines the scope of these kinds of errors including how they are reported and handled.
- Other errors not directly related to PCIe transactions
  - This class of errors are reported via interrupts and do not fall under the scope of PCIe error management mechanisms.

39.10.1 PCIe Error Management

PCIe error management focuses on errors associated with the PCIe interface and the transactions between the transaction layers of the transmitting and the receiving PCIe ports.

39.10.2 PCIe Error Reporting Mechanisms

The PCIe specification provides three mechanisms for reporting errors that occur while servicing to PCIe transactions.

1. PCI
   a. This mechanism provides backward compatibility with legacy PCI compatible software and is required per the PCIe specification. Legacy compatible PCI software will use this mechanism and control this feature via the PCI configuration Command register. When this mechanism is enabled, the endpoint will also log error status information in the PCI Configuration Status register.
   b. The endpoint will support the legacy PCI compatible error mechanism.

2. PCIe
   a. This is the baseline PCIe mechanism for software that understands PCIe devices. The mechanism can be enabled via the PCIe Device Control register. Error status is logged in the PCIe Device Status register.
   b. Supported error classifications: unsupported request type, fatal error, non-fatal error, and correctable error.
   c. The endpoint will support the baseline PCIe error reporting mechanism.
3. PCIe Advanced Error Reporting
   a. The optional advanced error reporting registers can be implemented by PCIe devices. This mechanism is defined using a PCIe extended capability structure.
   b. All endpoint Functions will support AER.

39.10.3 PCIe Error Handling and Signaling

The endpoint will classify errors in the following three PCIe specification-defined buckets:

1. Correctable errors: These errors are handled by the Hardware.
2. Uncorrectable errors (non-fatal): These errors are handled by device-specific software.
3. Uncorrectable errors (fatal): These errors are handled by system software.

The endpoint will use three different mechanisms to signal errors that occur when processing PCIe commands.

1. Completion status
2. Error forwarding or data poisoning
3. Error messages

39.10.4 PCIe Error Sources

Table 39-6 defines the standard PCIe defined error sources. The following sections describe each of the PCIe error types, and describe how the endpoint handles the PCIe errors.

Table 39-6. PCIe Error Sources

	Type of Error	Report Using	Comment
TL	ECRC Check	Uncorrectable Non-Fatal Message	
	Malformed TLP	Uncorrectable Fatal Message	Receive Interface will perform checks on the PCIe transactions received from IA.
	Completion Timeout	Uncorrectable Non-Fatal Message	Receive Interface will generate these split transaction errors. Completer abort is supported in the Receive Interface.
	Unsupported Requests	Completion Status	
	Completer Abort (optional)		
	Unexpected Completion	Uncorrectable Non-Fatal Message	
	Data Corruption / Poisoning	PCIe Header	Transmit Interface will set the endpoint bit in the3 header based on the data error. Writes that terminate in the Receive Interface get dropped and the status logged. Writes that target an endpoint function will be sent with a data error and logged in the Receive Interface.

39.10.4.1 Role-Based Error Reporting

In earlier versions of the PCI Express Specification, errors were reported by the agent that detected the error. The *PCI Express Base Specification,* Rev. 1.1 implements a role-based error reporting where the response to the errors is based on the components role in the transaction. In general, errors detected in non-posted transactions are handled by the initial requester and the completer may optionally send an advisory message to the root complex as an ERR_COR message. Errors in posted transactions are still logged and reported by the target device.

Note: If the severity for the error is programmed to fatal in the PCI Express Uncorrectable Error Severity register, then it is not an "advisory non-fatal error" and is signaled with an ERR_FATAL message. A fatal severity overrides all other Advisory Error Control bits.

The following errors are considered "advisory non-fatal error" cases and have different handling depending on the transaction type.

- ECRC check failed
- Unexpected completion
- Unsupported request
- Poisoned TLP received
- Completer abort
- Completion timeout

39.10.4.2 Malformed Packets

The following checks are made to detect malformed TLPs.

- Data payload exceeds the length specified by the value in the Max_Payload_Size field of the Device Control register.

When a malformed packet is detected, the packet is dropped and the error is logged. No flow control information is updated for malformed packets.

39.10.4.3 ECRC Check Failed

The endpoint does not calculate for ECRC but will report the ecrc_error signal that it receives. The endpoint reports by sending an ERR_NONFATAL message to the root complex.

39.10.4.4 Unsupported Request

Unsupported requests are detected by the address decode and translation logic. A TLP is treated as unsupported in the following cases:

- The TLP fails to match any of the active memory or I/O windows.
- A configuration TLP that targets an invalid function number.
- A receipt of a Vendor_Defined Type 0 message.
- A message request with an undefined or unsupported message code.
- A poisoned I/O or configuration request.
- A receipt of a memory or I/O transaction while in a non-D0 power state.
- A receipt of a Memory Read Lock (MRdLk).

No checks are made for the address plus the length crossing a window boundary.

For posted transactions, this is **not** an advisory error, and an ERR_NONFATAL is sent to the root complex.

For non-posted transactions, this is considered an advisory error. An ERR_COR is sent to the root complex and a completion with Unsupported Request status is returned to the requester.

Note: If the severity setting in the PCI Express Uncorrectable Error Severity register is fatal, this is not an advisory error, and an ERR_FATAL will be sent to the root complex.

39.10.4.5 Completer Abort

Requests that target abort or master abort on the internal bus are treated as a completer abort.

These requests must have passed the malformed TLP checks as well as the unsupported request checks before they are issued on the internal bus.

For posted transactions, an ERR_NONFATAL is sent to the root complex.

For non-posted transactions, an ERR_COR is sent to the root complex and a completion with Completer Abort status is returned to the requester.

Note: If the severity setting in the PCI Express Uncorrectable Error Severity register is fatal this is not an advisory error and an ERR_FATAL will be sent to the root complex.

39.10.4.6 Unexpected Completions

Unexpected completions occur when a completion transaction ID does not match an outstanding request. If the Requester ID of the completion matches a valid function, the error will get logged in that function. Otherwise, the error will get logged against all functions.

This is an advisory non-fatal error and an ERR_COR will be sent to the root complex.

Note: If the severity setting in the PCI Express Uncorrectable Error Severity register is fatal, this is not an advisory error and an ERR_FATAL will be sent to the root complex.

39.10.4.7 Poisoned TLP Received

Poisoned TLPs can be received for both inbound posted (write/message) and inbound completions. The two TLP types can be handled differently.

Poisoned completions are passed through to the target agent with the error bit set. The error is logged in the corresponding function. This is an advisory error and an ERR_COR will be issued.

Poisoned memory writes that are MMIO-mapped are passed through to the intended target. Depending on the register target, the MMIO-mapped register may get updated with the poisoned data, whereas in some cases the register will not be updated as the byte-enable signals will be disabled. The error is logged in the corresponding function. This is a non-fatal error and an ERR_NONFATAL will be issued.

Writes to configuration registers will not complete to the target. Since configuration writes are non-posted writes, a completion with Unsupported Request is returned to the host.

Note: If the severity setting in the PCI Express Uncorrectable Error Severity register is fatal, this is not a non-fatal error, and an ERR_FATAL will be sent to the root complex. Auto-recovery is discouraged as the ERR_FATAL message will likely bring down the hierarchy.

39.10.4.8 Completion Timeout

When an out-bound, non-posted request results in a completion timeout, the advanced error registers are updated in the corresponding function. This will be treated as an advisory error, and an ERR_COR will be sent to the root complex.

Note: If the severity setting in the PCI Express Uncorrectable Error Severity register is fatal, this is not an advisory error, and an ERR_FATAL will be sent to the root complex. Auto-recovery is discouraged as the ERR_FATAL message will likely bring down the hierarchy.

39.10.4.9 Non-Function Specific Errors

The PCI Express Specification lists the following errors as non-function specific:

- Transaction layer errors
  - — ECRC fail
  - — Unsupported Request, when caused by no function claiming a TLP
  - — Receiver overflow
  - — Flow control protocol error
  - — Malformed TLP
  - — Unexpected completion, when caused by no function claiming a completion

On the detection of one of these errors, a multi-function device should generate at most one error reporting message of a given severity, where the message must report the Requester ID of a function of the device that is enabled to report that specific type of error. If no function is enabled to send a reporting message, the device does not send a reporting message. If all reporting-enabled functions have the same severity level set for the error, only one error message is sent. If all reporting-enabled functions do not have the same severity level set for the error, one error message for each severity level is sent. Software is responsible for scanning all functions in a multifunction device when it detects one of those errors.

39.10.5 Device-Specific Error Management

The endpoint will report device-specific errors using interrupts—INTx or MSI based on the PCIe configuration settings.

Table 39-7. Device Specific Errors

Function	Description
Function 0	The endpoint implements error source and mask registers. When detecting an assertion on an unmasked error, the Receive Interface will generate either an INTA or MSI.

39.10.5.1 Memory Error Poisoning

The Transmit Interface also tracks data errors for all transactions that flow through the Transaction Interface, including accesses to PCIe and CSRs.

39.10.5.2 Memory Write Poisoning

For a PCIe memory write (including memory writes triggered by the Ring Controller), the Transmit Interface pulls data from the endpoint master. If the internal bus indicates an error, then the Transmit Interface logs the error, but it does not abort the write transaction. Instead, the Transmit Interface passes along the data error notification along with the data. On PCIe, the Transmit Interface will set the appropriate bit in the header of the packet.

39.11 Error Handling with SR-IOV

The endpoint ring-related errors that are associated with a given Virtual Function are handled by the given VF itself. For example, the VF would report and log the errors. Each VF supports the legacy PCI (PCI Command and Status registers) and PCIe reporting mechanism (ERR_COR, ERR_NONFATAL, ERR_FATAL messages). The VF also supports PCIe extended capability - AER. Errors that are not directly related to the ring associated with a given VF are handled by the PF. For example, errors such as endpoint parity errors, etc.

39.12 PCIe Endpoint Function 0 Registers (Intel QuickAssist Technology) - EndPoint (EP) Physical Function (PF) PCI Configuration Space

Table 39-8. Register Attributes / Modifiers

Attribute	Description
RO	**Read Only:** These bits can be read by Software, but writes have no effect. The value of the bit is determined by hardware only.
RW	**Read/Write:** These register bits can be read by software and modified by software
RW/1C	**Read/Write 1Clear**. These bits can be read from and written to by software. Writing a "0" to these bits has no effect. Writing a "1" clears the bit.
RV	**Reserved**: These bits are reserved. Their value must be maintained by software. When writing these bits, software must preserve the value read.
Modifier	**Description**
-O	**Once:** After reset, these bits can only be written to once. After that they are read only.
-V	**Variant:** The value of the bit can be updated by hardware.

The EP Physical Function implement Type 0 PCIe configuration space that is compliant with PCIe 3.0 Specification.

The Endpoint is a single-function device designated as: Bus (M) = 0xM, Device (D) = 0xD, and Function = 0x0.

Figure 39-5. PF EP PCI Configuration Registers (Sheet 1 of 3)

	3	2	1	0	Offset[1]
Required PCI Compatible Configuration Space	Device ID		Vendor ID		00h
	Status		Command		04h
	Class Code			Revision ID	08h
	BIST	Header Type	Master Latency Timer	Cache Line Size	0Ch
	PeQatBar				10h
					14h
	Base Address Register 3 2 (PMISCBAR)				18h
					1Ch
	Base Address Register 5 4 (PETRINGCSRBAR)				20h
					24h
	Cardbus CIS Pointer				28h
	Subsystem ID		Subsystem Vendor ID		2Ch
	Expansion ROM Base Address				30h
	Reserved			Capabilities Pointer	34h
	Reserved				38h
	Max_Lat	Min_Gnt	Interrupt Pin	Interrupt Line	3Ch
	Reserved				40h
	Reserved				44 4Bh
	Reserved				4Ch

Figure 39-5. PF EP PCI Configuration Registers (Sheet 2 of 3)

	3	2	1	0	Offset[1]
	Reserved				50-5Ch
MSI-X Cap	Message Control		Next Cap Pointer	MSI-X CAP_ID	60h
	MSI-X Table Offset and BIR				64h
	MSI-X PBA Offset and BIR				68h
PM Cap	Power Management Capabilities		Next Cap Pointer	PM CAP_ID	6Ch
	Power Management Control and Status				70h
PCI Cap	PCI Express Capabilities Register		Next Cap Pointer	PCIe CAP_ID	74h
	Device Capabilities				78h
	Device Status		Device Control		7Ch
	Link Capability (Reserved)				80h
	Link Status (Reserved)		Link Control (Reserved)		84h
	Slot Capabilities (Reserved)				88h
	Slot Status (Reserved)		Slot Control (Reserved)		8Ch
	Root Capabilities (Reserved)		Root Control (Reserved)		90h
	Root Status (Reserved)				94h
	Device Capabilities 2				98h
	Device Status 2		Device Control 2		9Ch
	Link Capabilities 2 (Reserved)				A0h
	Link Status 2 (Reserved)		Link Control 2 (Reserved)		A4h
	Slot Capabilities 2 (Reserved)				A8h
	Slot Status 2 (Reserved)		Slot Control 2 (Reserved)		ACh
MSI Cap	Message Control		Next Cap Pointer	MSI CAP_ID	B0h
	MSI Addr				B4h - BBh
	Reserved		MSI Data		BCh
	MSI Mask				C0h
	MSI pending				C4h
Reserved	Reserved				C8h - FCh
AER Configuration Space	Next Capability Pointer/Capability Version		AER Capability ID		100h
	Uncorrectable Error Status				104h
	Uncorrectable Error Mask				108h
	Uncorrectable Error Severity				10Ch
	Correctable Error Status				110h
	Correctable Error Mask				114h
	Control and Capability				118h
	Header Log				11Ch
	Header Log				120h
	Header Log				124h
	Header Log				128h
ARI Configuration Space	Next Capability Pointer/Capability Version		ARI Capability ID		138h
	ARI Control		ARI Capability		13Ch

Figure 39-5. PF EP PCI Configuration Registers (Sheet 3 of 3)

	3	2	1	0	Offset[1]
SR-IOV Configuration Space	Next Capability Pointer/Capability Version		SR-IOV Capability ID		140h
	SR-IOV Capabilities				144h
	SR-IOV Status		SR-IOV Control		148h
	Total VF's (RO)		Initial VF's (RO)		14Ch
	Reserved	Function Dependency	Num VF's (RW)		150h
	VF Stride (RO)		First VF Offset (RO)		154h
	VF Device ID (RO)		Reserved		158h
	Supported Page Sizes (RO)				15Ch
	System Page Size (RW)				160h
	VF BAR0 (RW)				164h
					168h
	VF BAR 1 (RW)				16Ch
					170h
	Reserved				174h
	Reserved				178h
	VF Migration Offset (N/A)				17Ch
Reserved	Reserved				180h -1ACh
ACS Configuration Space	Next Capability Pointer/Capability Version		ACS Capability ID		1B0h
	ACS Control Register		ACS Capability Register		1B4h
Reserved	Reserved				1B8h - FFCh

Notes:
1. Any addresses not shown are Read-only 0.

39.12.1 PCI Physical Function (PF) Views

Table 39-9. Bus M, Function 0: Summary of PCIe Intel QuickAssist Technology Configuration Registers (Sheet 1 of 3)

Offset Start	Offset End	Register ID - Description	Default Value
00h	01h	"PVID - PF Vendor Identification Register" on page 3783	8086h
02h	03h	"PDID - PF Device Identification Register" on page 3783	37C8h
04h	05h	"PPCICMD - PF Device Command Register" on page 3784	0000h
06h	07h	"PPCISTS - PF PCI Device Status Register" on page 3785	0010h
08h	0Bh	"PRID - PF Revision ID and Class Code Register" on page 3786	0B400000h
0Ch	0Ch	"PCLS - PF Cache Line Size Register" on page 3786	00h
0Eh	0Eh	"PHDR - PF Header Type Register" on page 3787	00h
18h	1Bh	"PMISCLBAR - PF Miscellaneous Lower Base Address Register" on page 3787	00000004h
1Ch	1Fh	"PMISCUBAR - PF Miscellaneous Upper Base Address Register" on page 3788	00000000h
20h	23h	"PETRINGCSRLBAR - PF Ring CSR Lower Base Address Register" on page 3788	00000004h
24h	27h	"PETRINGCSRUBAR - PF Ring CSR Upper Base Address Register" on page 3789	00000000h
2Ch	2Dh	"PSVID - PF Subsystem Vendor ID Register" on page 3789	8086h

Table 39-9. Bus M, Function 0: Summary of PCIe Intel QuickAssist Technology Configuration Registers (Sheet 2 of 3)

Offset Start	Offset End	Register ID - Description	Default Value
2Eh	2Fh	"PSID - PF Subsystem ID Register" on page 3790	0000h
34h	34h	"PCP - PF Capabilities Pointer Register" on page 3790	B0h
3Ch	3Ch	"PIRQL - PF Interrupt Line Register" on page 3791	00h
3Dh	3Dh	"PIRQP - PF Interrupt Pin Register" on page 3791	01h
60h	60h	"PMSIX - PF Message Signaled Interrupt X Capability ID Register" on page 3792	11h
61h	61h	"PMSIXNCP - PF MSIX Next Capability Pointer Register" on page 3792	6Ch
62h	63h	"PMSIXCNTL - PF Message Signaled Interrupt X Control Register" on page 3793	0010h
64h	67h	"PMSIXTBIR - PF MSI-X Table Offset and Table BIR Register" on page 3793	0003B002h
68h	6Bh	"PMSIXPBABIR - PF MSI-X Pending Bit Array and BIR Offset Register" on page 3794	0003B802h
6Ch	6Ch	"PPMCAP - PF Power Management Capabilities ID Register" on page 3794	01h
6Dh	6Dh	"PPMCP - PF Power Management Next Capability Pointer Register" on page 3795	74h
6Eh	6Fh	"PPMC - PF Power Management Capabilities Register" on page 3795	0023h
70h	73h	"PPMCSR - PF Power Management Control and Status Register" on page 3796	00000000h
74h	74h	"PPCID - PF PCI Express Capability Register" on page 3798	10h
75h	75h	"PPCP - PF PCI Express Next Capability Pointer Register" on page 3798	0h
76h	77h	"PPCR - PF PCI Express Capabilities Register" on page 3799	0002h
78h	7Bh	"PPDCAP - PF PCI Express Device Capabilities Register" on page 3799	10008061h
7Ch	7Dh	"PPDCNTL - PF PCI Express Device Control Register" on page 3801	3830h
7Eh	7Fh	"PPDSTAT - PF PCI Express Device Status Register" on page 3802	0000h
80h	83h	"PLCAPR - PF Link Capabilities Register" on page 3803	00400D02h
84h	85h	"PLCNTLR - PF Link Control Register" on page 3805	0h
86h	87h	"PLSR - PF Link Status Register" on page 3807	0h
98h	9Bh	"PDCAPR2 - PF Device Capabilities 2 Register" on page 3808	00000012h
9Ch	9Dh	"PDCNTR2 - PF Device Control 2 Register" on page 3809	0000h
A0h	A3h	"PDCNTR2 - PF Device Control 2 Register" on page 3809	4h
A4h	A5h	"PLCNTLR2 - PF Link Control 2 Register" on page 3810	0h
A6h	A7h	"PLSR2 - PF Link Status 2 Register" on page 3811	0h
B0h	B0h	"PMSICID - PF Message Signaled Interrupt Capability ID Register" on page 3812	05h
B1h	B1h	"PMSINCP - PF Message Signaled Interrupt Next Capability Pointer Register" on page 3812	60h
B2h	B3h	"PMSICTL - PF Message Signaled Interrupt Control Register" on page 3813	0180h
B4h	B7h	"PMSILADDR - PF Message Signaled Interrupt Lower Address Register" on page 3813	00000000h
B8h	BBh	"PMSIUADDR - PF Message Signaled Interrupt Upper Address Register" on page 3814	00000000h
BCh	BDh	"PMSIDATA - PF Message Signaled Interrupt Data Register" on page 3814	0000h
C0h	C3h	"PMSIMSK - PF Message Signaled Interrupt Mask Register" on page 3815	00000000h
C4h	C7h	"PMSIPND - PF Message Signaled Interrupt Pending Register" on page 3815	00000000h
100h	103h	"PPCIEAERCAPID - PF PCI Express AER Capability ID Register" on page 3816	13810001h
104h	107h	"PPAERUCS - PF PCI Express AER Uncorrectable Error Status Register" on page 3816	0h

Table 39-9. Bus M, Function 0: Summary of PCIe Intel QuickAssist Technology Configuration Registers (Sheet 3 of 3)

Offset Start	Offset End	Register ID - Description	Default Value
108h	10Bh	"PPAERUCM - PF PCI Express AER Uncorrectable Error Mask Register" on page 3818	0h
10Ch	10Fh	"PPAERUCSEV - PF PCI Express AER Uncorrectable Error Severity Register" on page 3819	00440000h
110h	113h	"PPAERCS - PF PCI Express AER Correctable Error Register" on page 3820	00h
114h	117h	"PPAERCM - PF PCI Express AER Correctable Error Mask Register" on page 3821	2000h
118h	11Bh	"PPAERCTLCAP - PF PCI Express AER Control and Capability Register" on page 3822	0h
11Ch	11Fh	"PPAERHDRLOG0 - PF PCI Express AER Header Log 0 Register" on page 3822	0h
120h	123h	"PPAERHDRLOG1 - PF PCI Express AER Header Log 1 Register" on page 3823	0h
124h	127h	"PPAERHDRLOG2 - PF PCI Express AER Header Log 2 Register" on page 3823	0h
128h	12Bh	"PPAERHDRLOG3 - PF PCI Express AER Header Log 3 Register" on page 3824	0h
138h	13Bh	"PARIDHDR - PF Alternative Routing ID Capability Header" on page 3824	1401000Eh
13Ch	13Dh	"PFARICAP - PF ARI Capabilities Register" on page 3825	0000h
13Eh	13Fh	"PARIDCTL - PF Alternative Routing ID Control Register" on page 3825	00000000h
140h	141h	"PSRIOVCAPID - PF SR IOV Capability ID Register" on page 3826	10h
142h	143h	"PSRIOVCVNC - PF SRIOV Capability Version and Next Capability Pointer Register" on page 3826	1B01h
144h	147h	"PSRIOVCAP - PF SRIOV Capabilities Register" on page 3827	00000000h
148h	14Bh	"PSRIOVCS - PF SRIOV Control and Status Register" on page 3827	00000000h
14Ch	14Fh	"PSRIOVMTOTINI - PF SRIOV Initial and Total VF's Register" on page 3828	00100010h
150h	153h	"PSRIOVNUMVF - PF SRIOV Number of VF's Register" on page 3828	00000000h
154h	155h	"PSRIOVFVFO - PF SRIOV First VF Offset Register" on page 3829	0008h
156h	157h	"PSRIOVVFS - PF SRIOV VF Stride Register" on page 3829	0001h
158h	15Bh	"PSRIOVFDID - PF SRIOV VF Device ID Register" on page 3830	000037C9h
15Ch	15Fh	"PSRIOVPAGESIZE - PF SRIOV Supported Page Size Register" on page 3830	00000553h
160h	163h	"PSRIOVSYSPS - PF SRIOV System Page Size Register" on page 3831	00000001h
164h	167h	"PSRIOVLBAR0 - SRIOV Lower BAR0 Register" on page 3832	00000004h
168h	16Bh	"PSRIOVUBAR0 - SRIOV Upper BAR0 Register" on page 3833	00000000h
16Ch	16Fh	"PSRIOVLBAR1 - SRIOV Lower BAR1 Register" on page 3833	00000004h
170h	173h	"PSRIOVUBAR1 - SRIOV Upper BAR1 Register" on page 3834	00000000h
17Ch	17Fh	"PSRIOVVFMA - PF SRIOV VF Migration Array Register" on page 3834	00000000h
1B0h	1B1h	"PACSCAPID - PF ACS Capability ID Register" on page 3835	00Dh
1B2h	1B3h	"PACSCVNC - PF ACS Capability Version and Next Capability Pointer Register" on page 3835	1h
1B4h	1B7h	"PACSCAP - PF ACS Capabilities Register" on page 3836	00000000h

39.12.2 PCI Standard Header Registers

39.12.2.1 PVID - PF Vendor Identification Register

The VID Register contains the vendor identification number. This 16-bit register combined with the Device Identification Register uniquely identifies any PCI device. Writes to this register have no effect.

Table 39-10. PVID - PF Vendor Identification Register

Description: PF Vendor Identification Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** 00h **Offset End:** 01h	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 00h **Offset End:** 01h	
Size: 16 bit	**Default:** 8086h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	VID	**Vendor Identification:** This register field contains the PCI standard identification for Intel, 8086h.		8086h	RO

39.12.2.2 PDID - PF Device Identification Register

This 16-bit register combined with the Vendor Identification register uniquely identifies any PCI device. Writes to this register have no effect.

Table 39-11. PDID - PF Device Identification Register

Description: PF Device Identification Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** 02h **Offset End:** 03h	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 02h **Offset End:** 03h	
Size: 16 bit	**Default:** 37C8h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	DIDLOW	**Device Identification Number:** DID = 37C8h		37C8h	RO

39.12.2.3 PPCICMD - PF Device Command Register

Table 39-12. PPCICMD - PF Device Command Register

Description: PF Device Command Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** 04h **Offset End:** 05h	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 04h **Offset End:** 05h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :11	Reserved	Reserved		0h	RO
10	INTD	**Interrupt Disable:** Setting this bit disables generation of INTX messages by the EP. Default value is 0 which enables the INTX message generation.		0h	RW
09	FBTB	**Fast Back-to-Back Enable:** EP does not implement this functionality and it is not applicable to PCIe devices. The bit is hardwired to 0.		0h	RO
08	SER	**SERR# Enable:** When set, this bit enables the non-fatal and fatal errors detected by the EP to be reported to the RC. Note that the error reporting can also be enabled via the PCIe specific bits in the PCIe device control register (Section 39.12.5.5, "PPDCNTL - PF PCI Express Device Control Register") The default value of this bit is 0.		0h	RW
07	Reserved1	Reserved/Does not apply to PCIe.		0h	RO
06	PER	**Parity Error Enable:** Controls the setting of the Master Data Parity Error bit in the Device Status Register (Section 39.12.2.4, "PPCISTS - PF Device Status Register") The Master Data Parity Error bit is set by the EP if its Parity Error Enable bit is set and either of the following two conditions occurs: • If the EP receives a poisoned Completion from the RC • If the EP poisons a write request. If the Parity Error Enable bit is cleared, the Master Data Parity Error status bit is never set The default value of this bit is 0.		0h	RW
05	VPS	**VGA Palette Snoop Enable:** The device does not implement this functionality/Does not apply to PCIe. The bit is hardwired to 0.		0h	RO
04	MWE	**Memory Write and Invalidate Enable:** The device does not implement this functionality/Does not apply to PCIe. The bit is hardwired to 0.		0h	RO
03	SS	**Special Cycle Enable:** The device does not implement this functionality/Does not apply to PCIe. The bit is hardwired to 0.		0h	RO
02	BM	**Bus Master Enable:** Controls the ability of EP to issue Memory Read/Write Requests. Clearing (0) this bit prevents EP from issuing any Memory Requests. Note that as MSI's are in-band memory writes, disabling the bus master enable bit disables MSI as well. PCIe messages are not affected by this bit.		0h	RW
01	MEM	**Memory Space Enable**: Setting this bit enables access to the memory regions the device claims through its BARs. EP will return "unsupported request" completion status and error message in response to memory transactions it receives when this bit is clear.		0h	RW
00	IO	**I/O Space Enable:** The device does not implement this functionality since it claims no I/O regions. The bit is hardwired to 0.		0h	RO

39.12.2.4 PPCISTS - PF Device Status Register

Table 39-13. PPCISTS - PF PCI Device Status Register

Description: PF device status register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** 06h **Offset End:** 07h	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 06h **Offset End:** 07h	
Size: 16 bit	**Default:** 0010h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	DPE	**Detected Parity Error:** This bit is set by EP whenever it receives a Poisoned TLP, regardless of the state the Parity Error Enable bit in the Command register. Default value of this field is 0.		0h	RW/1C/V
14	SSE	**Signaled System Error:** This bit is set by the EP when it sends a ERR_FATAL or ERR_NONFATAL message and the SERR bit in the Device Command register bit is set. Default value of this field is 0.		0h	RW/1C/V
13	RMA	**Received Master Abort Status:** This bit is set when EP, as a Requester receives a Completion with Unsupported Request Completion Status. Default value of this field is 0.		0h	RW/1C/V
12	RTA	**Received Target Abort Status:** This bit is set when EP, as a Requester receives a Completion with Completer Abort Completion Status. Default value of this field is 0.		0h	RW/1C/V
11	STA	**Signaled Target Abort Status:** This bit is set when EP completes a Request using Completer Abort Completion Status. Default value of this field is 0.		0h	RW/1C/V
10 :09	DST	**DEVSEL Timing:** Does not apply to PCI Express. These bits are hardwired to 0.		00b	RO
08	MDPE	**Master Data Parity Error Detected:** This bit is set by EP, as a Requester if the Parity Error Enable bit in the Command register is 1b and either of the following two conditions occurs: • Requester receives a Completion marked poisoned • Requester poisons a write Request If the Parity Error Enable bit is 0b, this bit is never set. Default value of this field is 0.		0h	RW/1C/V
07	FB2B	**Fast Back-to-Back Capable:** Does not apply to PCI Express. The bit is hardwired to 0.		0h	RO
06	Reserved1	Reserved1		0h	RO
05	MC66	**66 MHz Capable:** Does not apply to PCI Express. The bit is hardwired to 0.		0h	RO
04	CL	**Capabilities List:** This bit is hardwired to 1 to indicate that EP has a capabilities list.		1h	RO
03	INTSTS	**Interrupt Status:** Indicates that the EP has transmitted a INTX message and is awaiting servicing. This bit does not includes MSI's generated by the EP.		0h	RO/V
02 :00	Reserved2	Reserved2		0h	RO

39.12.2.5 PRID - PF Revision ID Register

The value of this register indicates the chip stepping. It is hardwired on chip and reflects the latest revision.

Table 39-14. PRID - PF Revision ID and Class Code Register

Description: PF revision ID register and PF class code Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** 08h **Offset End:** 0Bh	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 08h **Offset End:** 0Bh	
Size: 32 bit	**Default:** 0B400000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :8	CC	**Class Code:** This value indicates the base class, subclass, and interface. 0B4000h = Base class: Processor, Sub-class Co-processor, no specific register level programming interfaces are defined.		0B4000h	RW/O
07 :4	RIDU	Major Revision: Steppings which require all masks to be regenerated. 00b: A stepping (Current Stepping) 01b: B stepping 10b: C stepping 11b: D stepping		0000b	RO
03 :0	RIDL	Minor Revision: Incremented for each stepping which does not modify all masks. Reset for each major revision. 00b: x0 stepping (Current Stepping) 01b: x1 stepping 10b: x2 stepping 11b: x3 stepping		0000b	RO

39.12.2.6 PCLS - PF Cache Line Size Register

Table 39-15. PCLS - PF Cache Line Size Register

Description: PF Cache Line size Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** 0Ch **Offset End:** 0Ch	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 0Ch **Offset End:** 0Ch	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	CLS	**Cache Line Size:** Not used by PCI Express Endpoint. Left for compatibility reason only.		00h	RW

39.12.2.7 PHDR - PF Header Type Register

Table 39-16. PHDR - PF Header Type Register

Description: PF Header type Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 0Eh **Offset End:** 0Eh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 0Eh **Offset End:** 0Eh	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	HDR	**PCI Header Type:** The header type of the EP device. 00h (Bit 7 = 0b) indicates a single-function device with standard header layout.		00h	RO

39.12.2.8 PMISCLBAR - PF Miscellaneous Lower Base Address Register

The BAR is used to access EP and MSI-X registers.

Table 39-17. PMISCLBAR - PF Miscellaneous Lower Base Address Register

Description: PF Miscellaneous Lower Base Address Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 18h **Offset End:** 1Bh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 18h **Offset End:** 1Bh	
Size: 32 bit	**Default:** 00000004h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :18	ADDR	**Lower Programmable Base Address:** These bits are set by BIOS to locate the base address of the region.		0h	RW
17 :04	ZERO	**Lower Bits:** Hardwired to 0 (256KB region).		0h	RO
03	PREF	**Prefetchable:** Hardwired to 0 to indicate that the region is non-prefetchable.		0b	RO
02 :01	TYP	**Addressing Type:** Hardwired to indicate a 64-bit region.		10b	RO
00	MEM	**Memory Space Indicator:** Hardwired to 0 to identify the region as in memory space.		0b	RO

39.12.2.9 PMISCUBAR - PF Miscellaneous Upper Base Address Register

This BAR is used to access EP MMIO space.

Table 39-18. PMISCUBAR - PF Miscellaneous Upper Base Address Register

Description: PF Miscellaneous Upper Base Address Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 1Ch **Offset End:** 1Fh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 1Ch **Offset End:** 1Fh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	ADDR	**Upper Programmable Base Address:** These bits are set by BIOS to locate the base address of the region.		0h	RW

39.12.2.10 PETRINGCSRLBAR - PF Ring CSR Lower Base Address Register

This BAR points to the Ring Controller rings.

Table 39-19. PETRINGCSRLBAR - PF Ring CSR Lower Base Address Register

Description: PF Ring CSR Lower Base Address Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 20h **Offset End:** 23h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 20h **Offset End:** 23h	
Size: 32 bit	**Default:** 00000004h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :18	ADDR	**Lower Programmable Base Address:** These bits are set by BIOS to locate the base address of the region.		0h	RW
17 :04	ZERO	**Lower Bits:** Hardwired to 0 (256KB region).		0h	RO
03	PREF	**Prefetchable:** Hardwired to 0 to indicate that the region is non-prefetchable.		0b	RO
02 :01	TYP	**Addressing Type:** Hardwired to indicate a 64-bit region.		10b	RO
00	MEM	**Memory Space Indicator:** Hardwired to 0 to identify the region as in memory space.		0b	RO

39.12.2.11 PETRINGCSRUBAR - PF Ring CSR Upper Base Address Register

This BAR points to the Ring controller rings.

Table 39-20. PETRINGCSRUBAR - PF Ring CSR Upper Base Address Register

Description: PF Ring CSR Upper Base Address Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 24h **Offset End:** 27h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 24h **Offset End:** 27h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	ADDR	**Upper Programmable Base Address:** These bits are set by BIOS to locate the base address of the region.		0h	RW

39.12.2.12 PSVID - PF Subsystem Vendor ID Register

This register is a write-once register. Once any byte in the register has been written, the register locks against further writes until reset.

Table 39-21. PSVID - PF Subsystem Vendor ID Register

Description: PF Subsystem Vendor ID Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 2Ch **Offset End:** 2Dh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 2Ch **Offset End:** 2Dh	
Size: 16 bit	**Default:** 8086h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	SVID	**Subsystem Vendor ID:** This field is hardwired to ID assigned to Intel.		8086h	RO

39.12.2.13 PSID - PF Subsystem ID Register

This register is a write-once register. Once any byte in the register has been written, the register locks against further writes until reset.

Table 39-22. PSID - PF Subsystem ID Register

Description: PF Subsystem ID Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** 2Eh **Offset End:** 2Fh	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 2Eh **Offset End:** 2Fh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	SID	**Subsystem ID:** Vendor supplied device ID. Default is 0h.		0h	RW/O

39.12.2.14 PCP - PF Capabilities Pointer Register

The Capabilities Pointer Register provides the offset in configuration space to the location where the first set of capabilities registers is located.

Table 39-23. PCP - PF Capabilities Pointer Register

Description: PF Capabilities Pointer Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** 34h **Offset End:** 34h	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 34h **Offset End:** 34h	
Size: 8 bit	**Default:** B0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	CP	**Pointer to First Capability Structure:** Value points to the configuration space offset of the first capability structure (MSI).		B0h	RO

39.12.2.15 PIRQL - PF Interrupt Line Register

Table 39-24. PIRQL - PF Interrupt Line Register

Description: PF Interrupt Line Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 3Ch **Offset End:** 3Ch	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 3Ch **Offset End:** 3Ch	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	IRQL	**Interrupt Line:** BIOS writes the interrupt routing information to this register to indicate which input of the interrupt controller this device is connected to. The device itself does not use this information.		0h	RW

39.12.2.16 PIRQP - PF Interrupt Pin Register

Table 39-25. PIRQP - PF Interrupt Pin Register

Description: PF Interrupt Pin Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 3Dh **Offset End:** 3Dh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 3Dh **Offset End:** 3Dh	
Size: 8 bit	**Default:** 01h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	IRQP	**Interrupt Pin:** Set to 01h to indicate that EP uses INTA# as its interrupt pin.		01h	RO

39.12.3 MSI-X Capability Structure

39.12.3.1 PMSIX - PF Message Signaled Interrupt X Capability ID Register

MSI-X Capability ID Register indicates that the device is capable of generating MSI-X Interrupts.

Table 39-26. PMSIX - PF Message Signaled Interrupt X Capability ID Register

Description: PF Message Signaled Interrupt X Capability ID Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** 60h **Offset End:** 60h	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 60h **Offset End:** 60h	
Size: 8 bit	**Default:** 11h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	MSIX	**Capability ID:** PCI SIG assigned capability record ID (11h, MSI-X capability). 11h identifies the EP as a device that is MSI-X capable.		11h	RO

39.12.3.2 PMSIXNCP - PF MSIX Next Capability Pointer Register

Table 39-27. PMSIXNCP - PF MSIX Next Capability Pointer Register

Description: PF MSIX Next Capability Pointer Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** 61h **Offset End:** 61h	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 61h **Offset End:** 61h	
Size: 8 bit	**Default:** 6Ch			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	MCP	**Next Capability Pointer:** Power Management Capability		6Ch	RO

39.12.3.3 PMSIXCNTL - PF Message Signaled Interrupt X Control Register

Table 39-28. PMSIXCNTL - PF Message Signaled Interrupt X Control Register

Description: PF Message Signaled Interrupt X Control Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 62h **Offset End:** 63h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 62h **Offset End:** 63h	
Size: 16 bit	**Default:** 0010h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	MSIXEN	MSI-X Enable: This bit enables the EP to generates interrupts using the MSI-X tables instead of the legacy INTx messages. When this bit is set to 1, the EP will not generate INTx messages and must use the MSI-X to signal interrupts. The device driver should not clear this bit to mask interrupts. This bit will be set by the system PCI device manager.		0h	RW
14	FM	Function Mask: This bit controls the masking of all vectors implemented in the EP. When this bit is 0, each vectors mask bit determines whether the vector is masked or not. When this bit is 1, all vectors are masked regardless of the per vector masking bit.		0h	RW
13 :11	Reserved	Always returns 0's when read.		0h	RO
10 :00	TS	MSI-X Table Size: Number of vectors (encoded as N-1) supported by the EP. The EP supports 17 vectors.		10h	RO

39.12.3.4 PMSIXTBIR - PF MSI-X Table Offset and Table BIR Register

Table 39-29. PMSIXTBIR - PF MSI-X Table Offset and Table BIR Register

Description: PF MSI-X Table Offset and Table BIR Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 64h **Offset End:** 67h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 64h **Offset End:** 67h	
Size: 32 bit	**Default:** 0003B002h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :3	TO	Table Offset: Offset to a location in one of the EP's BAR's (indicated by TBIR) that points to the location of the base of the MSI-X Table. The EP MSI-X Table maps to an offset of 236KB in PMISCBAR.		00000000000001110110000000000b	RO
2 :0	TBIR	Table BAR Indicator Register: The BIR points to the EP PMISCBAR register (18h)		010b	RO

39.12.3.5 PMSIXPBABIR - PF MSI-X Pending Bit Array and BIR Offset Register

Table 39-30. PMSIXPBABIR - PF MSI-X Pending Bit Array and BIR Offset Register

Description: PF MSI-X Pending Bit Array and BIR Offset Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 68h **Offset End:** 6Bh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 68h **Offset End:** 6Bh	
Size: 32 bit	**Default:** 0003B802h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :3	PBAO	Pending Bit Array Offset: Offset to a location in one of the EP BAR's (indicated by PBABIR) that points to the location of the base of the MSI-X PBA. The EP MSI-X PBA maps to an offset of 238KB in PMISCBAR.		000000000000001110111000 00000b	RO
2 :00	PBABIR	Pending Bit Array BAR Indicator Register: The PBABIR points to the EP PMISCBAR register (18h)		010b	RO

39.12.4 Power Management Capability Structure

39.12.4.1 PPMCAP - PF Power Management Capabilities ID Register

EP supports the PCI bus Power Management Interface Specification Rev 1.2 based PCIe Power Management. The specification requires the implementation of the PCI Power Management Capabilities Register.

Table 39-31. PPMCAP - PF Power Management Capabilities ID Register

Description: PF Power Management Capabilities ID Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 6Ch **Offset End:** 6Ch	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 6Ch **Offset End:** 6Ch	
Size: 8 bit	**Default:** 01h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	PMID	**Capability ID:** PCI SIG assigned capability record ID (01h, Power Management capability)		01h	RO

39.12.4.2 PPMCP - PF Power Management Next Capability Pointer Register

Table 39-32. PPMCP - PF Power Management Next Capability Pointer Register

Description: PF Power Management Next Capability Pointer Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 6Dh **Offset End:** 6Dh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 6Dh **Offset End:** 6Dh	
Size: 8 bit	**Default:** 74h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	PMCP	**Next Capability Pointer:** PCI Express Capability.		74h	RO

39.12.4.3 PPMC - PF Power Management Capabilities Register

EP supports the PCI bus Power Management Interface Specification Rev 1.2 based PCIe Power Management. The specification requires the implementation of the PCI Power Management Capabilities Register.

Table 39-33. PPMC - PF Power Management Capabilities Register

Description: PF Power Management Capabilities Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 6Eh **Offset End:** 6Fh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 6Eh **Offset End:** 6Fh	
Size: 16 bit	**Default:** 0023h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :11	PME	**PME Support**: Set to indicate that the EP will not assert PME# in any state.		00000b	RO
10	D2	**D2 Support**. EP does not support D2 state.		0b	RO
9	D1	**D1 Support**. EP does not support D1 state.		0b	RO
8 :6	AC	**Aux current**: Not relevant for EP.		000b	RO
5	DSI	**Device specific initialization**: A one in this bit indicates that immediately after entry into the D0 Uninitialized state, the function requires additional configuration above and beyond setup of its PCI configuration Header registers before the Class driver can use the function. For the EP this bit is set to 1.		1b	RO
4	Reserved	Reserved		0b	RO
3	PMC	**PME clock**. Does not apply to PCIe.		0b	RO
2 :0	PMV	**Version**. This field is set to 3 to indicate that the EP is compliant with the PCI bus Power Management Interface Specification Rev 1.2.		011b	RO

39.12.4.4 PPMCSR - PF Power Management Control and Status Register

EP supports the PCI bus Power Management Interface Specification Rev 1.2 based PCIe Power Management. The specification requires the implementation of the PCI Power Management Capabilities Register.

Table 39-34. PPMCSR - PF Power Management Control and Status Register (Sheet 1 of 2)

Description: PF Power Management Control and Status Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 70h **Offset End:** 73h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 70h **Offset End:** 73h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	Reserved	Reserved: Not Supported for EP.		0000h	RO
15	PMS	**PME Status**: Hardwired to 0 since EP will not generate PME.		0b	RO
14 :13	DSC	**Data Scale**: Set to 0 because the EP does not implement the Data register. If the Data register has not been implemented, this field must return "00b" when the PMCSR is read.		00b	RO
12 :9	DSEL	**Data Select**: Set to 0 for EP because the EP does not implement the Data register (The Data register is an optional, 8-bit read-only register that provides a mechanism for the function to report state dependent operating data such as power consumed or heat dissipation) If the Data register is not implemented, this field should be read only and return "0000b" when the PMCSR is read.		0000b	RO
8	PME	**PME Enable**: Hardwired to 0 since EP does not generate a PME.		0b	RO
7 :4	Reserved1	Reserved1		0000b	RO

Table 39-34. PPMCSR - PF Power Management Control and Status Register (Sheet 2 of 2)

Description: PF Power Management Control and Status Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 70h **Offset End:** 73h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 70h **Offset End:** 73h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
3	NSR	**No_Soft_Reset**: Set to 0. The EP will not maintain functional context in the D3_hot state. So when the EP is transitioned from D3_hot to D0 it will be in the D0_uniniitialized state and therefore would require software to initialize it before it can transition to the D0_initialized state. When set (“1”), this bit indicates that devices transitioning from $D3_{HOT}$ to D0 because of Power State commands do not perform an internal reset. Configuration Context is preserved. Upon transition from the D3_hot to the D0_Initialized state, no additional operating system intervention is required to preserve Configuration Context beyond writing the PowerState bits. When clear (“0”), devices do perform an internal reset upon transitioning from D3_hot to D0 via software control of the Power State bits. Configuration Context is lost when performing the soft reset. Upon transition from the D3_hot to the D0 state, full reinitialization sequence is needed to return the device to D0_initialized state		0b	RO
2	Reserved2	Reserved2		0b	RO
1 :0	PS	**Power State**. This field is used to determine the current power state of EP and to set a new power state. • 00: D0 • 01: D1 (Not supported by EP, ignore writes with this value) • 10: D2 (Not supported by EP, ignore writes with this value) • 11: D3_hot If software selects a Power state that is not supported by the EP (D2/D1), the writes must complete normally on PCIe, but the write data is discarded and no state change occurs.		00b	RW

39.12.5 PCI Express Capability Structure

39.12.5.1 PPCID - PF PCI Express Capability ID Register

The PCI Express Capability List register enumerates the PCI Express Capability structure in the PCI 3.0 configuration space capability list.

Table 39-35. PPCID - PF PCI Express Capability Register

Description: PF PCI Express Capability Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** 74h **Offset End:** 74h	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 74h **Offset End:** 74h	
Size: 8 bit	**Default:** 10h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	PCIECID	**Capability ID:** PCI SIG assigned capability record ID (10h, PCI Express capability)		10h	RO

39.12.5.2 PPCP - PF PCI Express Next Capability Pointer Register

Table 39-36. PPCP - PF PCI Express Next Capability Pointer Register

Description: PF PCI Express Next Capability Pointer Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** 75h **Offset End:** 75h	
View: PCI PF	**BAR:** Configuration	**Bus:Device:Function:** M:0		**Offset Start:** 75h **Offset End:** 75h	
Size: 8 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	PCIENP	**Next Capability Pointer:** Last Capability.		00h	RO

39.12.5.3 PPCR - PF PCI Express Capabilities Register

Table 39-37. PPCR - PF PCI Express Capabilities Register

Description: PF PCI Express Capabilities Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 76h **Offset End:** 77h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 76h **Offset End:** 77h	
Size: 16 bit	**Default:** 0002h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :14	Reserved	Reserved		00b	RO
13 :9	IMN	**Interrupt Message Number:** This field indicates which MSI vector is used for the interrupt message generated in association with the status bits in either the Slot Status field of this capability structure (applies to only RC or Switch) Not applicable to EP.		00000b	RO
8 8	SI	**Slot Implemented:** This bit when set indicates that the PCI Express Link associated with this port is connected to a slot. Hardwired to 0 for EP.		0b	RO
7 :4	DPT	**Device/Port Type:** Indicates the type of PCI Express logical device. Hardwired to 0000b (PCIe Endpoint)		0000b	RO
3 :0	CV	**Capability Version:** Indicates PCI-SIG defined PCI Express capability structure version number. EP is PCIe 3.0 Specification Compliant.		0010b	RO

39.12.5.4 PPDCAP - PF PCI Express Device Capabilities Register

Table 39-38. PPDCAP - PF PCI Express Device Capabilities Register (Sheet 1 of 2)

Description: PF PCI Express Device Capabilities Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 78h **Offset End:** 7Bh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 78h **Offset End:** 7Bh	
Size: 32 bit	**Default:** 10008061h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :29	Reserved0	Reserved0		0h	RO
28	FLR	**Function level reset**: When set indicates that the device supports the FLR feature.		1b	RO
27 :26	CSPS	**Captured Slot Power Limit Scale:** Does not apply to EP.		00b	RO
25 :18	CSPV	**Captured Slot Power Limit Value:** Does not apply to EP.		0h	RO
17 :16	Reserved1	Reserved1		0h	RO
15	RBEP	**Role-Based Error Reporting:** Indicates that EP conforms to Role based error reporting ECN for PCIe 1.0a and which was subsequently rolled in PCIe 1.1 and future revisions.		1b	RO
14 :12	ATTN	**Attention Button/Indicator Present and Power Indicator Present**. None of these are implemented in the EP.		000b	RO

Table 39-38. PPDCAP - PF PCI Express Device Capabilities Register (Sheet 2 of 2)

Description: PF PCI Express Device Capabilities Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 78h **Offset End:** 7Bh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 78h **Offset End:** 7Bh	
Size: 32 bit	**Default:** 10008061h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
11 :9	EL1L	**Endpoint L1 Acceptable Latency:** This field indicates the acceptable latency that an Endpoint can withstand due to the transition from L1 state to the L0 state. Hardcoded to the lowest value of 1us. Since EP does not have a physical layer, this functionality is not implemented.		000b	RO
8 :6	EL0L	**Endpoint L0s Acceptable Latency:** This field indicates the acceptable latency that an Endpoint can withstand due to the transition from L1 state to the L0 state. This value is hardcoded to the latency of 64-128ns. Since EP does not have a physical layer, this functionality is not implemented.		001b	RO
5	ETFS	**Extended Tag Field Supported:** This field indicates the maximum supported size of the Tag field as a Requester. When Clear indicates 5-bit Tag field is supported. When set 8-bit Tag field is supported. Supports 96 outstanding requests.		1b	RO
4 :3	PFS	**Phantom Functions Supported:** This field indicates the support for use of unclaimed function numbers to extend the number of outstanding transactions allowed. EP does not use this capability.		00b	RO
2 :0	MPS	**Max_Payload_Size Supported:** This field indicates the maximum payload size that EP can support for TLPs. This value is set to indicate 256B. The defined encodings are: • 000b = 128B max payload size • 001b = 256B max payload size (Max supported) • 010b - 111b = Reserved		001b	RO

39.12.5.5 PPDCNTL - PF PCI Express Device Control Register

Table 39-39. PPDCNTL - PF PCI Express Device Control Register (Sheet 1 of 2)

Description: PF PCI Express Device Control Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 7Ch **Offset End:** 7Dh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 7Ch **Offset End:** 7Dh	
Size: 16 bit	**Default:** 3830h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	STARTFLR	**Initiate FLR** - Used to initiate FLR transition. A write of 1 initiates FLR transition. Since hardware must not respond to any cycles till FLR completion, the value read by software from this bit is 0.		0b	RW
14 :12	MRS	**Max_Read_Request_Size:** This field sets the maximum Read Request size for the EP as a Requester. The EP is capable for generating up to 2kB read requests. However requests generated by the EP will be limited by the programmed value in this field. Defined encodings for this field are: 000b = 128B max read request size 001b = 256B max read request size 010b = 512B max read request size (Default) 011b = 1024B max read request size 100b - 111b = Reserved		010b	RW
11	ENS	**Enable No Snoop (NS):** If this bit is set to 1, EP is permitted to set the No Snoop bit in the Requester Attributes of transactions it initiates. When clear all transactions will have the No Snoop bit clear. Note that setting this bit will not cause the EP to set the No Snoop attribute on every memory requests that it initiates.		1b	RW
10	APME	**Auxiliary (AUX) Power PM Enable:** This bit when set enables a device to draw AUX power independent of PME AUX power. Does not apply to EP		0b	RO
9	PFE	**Phantom Functions Enable:** When set, this bit enables a device to use unclaimed functions as Phantom Functions. Does not apply to EP.		0b	RO
8	ETFE	**Extended Tag Field Enable:** When set, this bit enables a device to use an 8-bit Tag field as a requester. Supports 96 outstanding requests.		0b	RW
7 :5	MPS	**Max_Payload_Size (MPS):** This field sets maximum TLP payload for EP. As a Receiver, the EP must handle TLPs as large as the set value; as a Transmitter, the EP must not generate TLPs exceeding the set value. 000b = 128B(Default) 001b = 256B Others values not supported by EP		000b	RW
4	ERO	**Enable Relaxed Ordering:** If this bit is set, EP is permitted to set the Relaxed Ordering bit in the Attributes field of transactions it initiates. Note that setting this bit does not cause the EP to set the RO on every transaction it issues.		1b	RW
3	URRO	**Unsupported Request Reporting Enable**: This bit, in conjunction with other bits, controls the signaling of Unsupported Requests by sending Error Messages to the root port. When clear it disables sending of error messages.		0b	RW

Table 39-39. PPDCNTL - PF PCI Express Device Control Register (Sheet 2 of 2)

Description: PF PCI Express Device Control Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 7Ch **Offset End:** 7Dh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 7Ch **Offset End:** 7Dh	
Size: 16 bit	**Default:** 3830h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
2	FERE	**Fatal Error Reporting Enable**: This bit, in conjunction with other bits, controls sending ERR_FATAL Messages to the root port. When clear disables sending of error messages.		0h	RW
1	NERE	**Non-Fatal Error Reporting Enable**: This bit, in conjunction with other bits, controls sending ERR_NONFATAL Messages to the root port. When clear disables sending of error messages.		0h	RW
0	CERE	**Correctable Error Reporting Enable:** This bit, in conjunction with other bits, controls sending ERR_COR Messages to the root port. When clear disables sending of error messages.		0h	RW

39.12.5.6 PPDSTAT - PF PCI Express Device Status Register

Table 39-40. PPDSTAT - PF PCI Express Device Status Register (Sheet 1 of 2)

Description: PF PCI Express Device Status Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 7Eh **Offset End:** 7Fh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 7Eh **Offset End:** 7Fh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :6	Reserved	Reserved		0h	RO
5	TP	**Transactions Pending:** This bit when set indicates that EP has issued Non-Posted Requests which have not been completed either with a completion packet or completion timeout mechanism.		0h	RO/V
4	APD	**AUX Power Detected:** Devices that require AUX power report this bit as set if AUX power is detected by the device. ***Note:*** Does not apply to this EP function.		0h	RO
3	URD	**Unsupported Request Detected:** This bit indicates that EP received an Unsupported Request. A one indicates that an error was detected since the last time this bit was cleared. Errors are updated in this field regardless of whether the error reporting bit is set in the Device Control Register.		0h	RW/1C/V

Table 39-40. PPDSTAT - PF PCI Express Device Status Register (Sheet 2 of 2)

Description: PF PCI Express Device Status Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 7Eh **Offset End:** 7Fh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 7Eh **Offset End:** 7Fh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
2	FED	**Fatal Error Detected:** This bit indicates status of Fatal errors detected. A one indicates that an error was detected since the last time this bit was cleared. Errors are updated in this field regardless of whether the error reporting bit is set in the Device Control Register.		0h	RW/1C/V
1	NED	**Non-Fatal Error Detected:** This bit indicates status of Nonfatal errors detected. A one indicates that an error was detected since the last time this bit was cleared. Errors are updated in this field regardless of whether the error reporting bit is set in the Device Control Register.		0h	RW/1C/V
0	CED	**Correctable Error Detected:** This bit indicates status of correctable errors detected. A one indicates that an error was detected since the last time this bit was cleared. Errors are updated in this field regardless of whether the error reporting bit is set in the Device Control Register.		0h	RW/1C/V

39.12.5.7 PLCAPR– PF Link Capabilities Register

Table 39-41. PLCAPR - PF Link Capabilities Register (Sheet 1 of 2)

Description: Link Capabilities Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 80h **Offset End:** 83h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 80h **Offset End:** 83h	
Size: 32 bit	**Default:** 00400D02h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	PORTNUM	**Port Number**: Assigned by EP after link training phase.		0h	RW/O
23 :23	Reserved	Reserved		0b	RO
22 :22	ASPM	ASPM Optionality Compliance ***Note:*** Defined for PCIe compliance.		1b	RO
21 :21	LBN	**Link Bandwidth Notification** Capability: ***Note:*** Does not apply to PCIe Endpoints/EP.		0b	RO
20 :18	Reserved	Reserved		000b	RO

Table 39-41. PLCAPR - PF Link Capabilities Register (Sheet 2 of 2)

Description: Link Capabilities Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 80h **Offset End:** 83h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 80h **Offset End:** 83h	
Size: 32 bit	**Default:** 00400D02h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
17 :15	L1EL	**L1 Exit Latency**- Indicates the exit latency from L1 to L0 state. 000b - Less than 1 ìs 001b - 1 ìs - 2 ìs 010b - 2 ìs - 4 ìs 011b - 4 ìs - 8 ìs 100b - 8 ìs - 16 ìs 101b - 16 ìs - 32 ìs 110b - 32 ìs - 64 ìs 111b - L1 transition not supported ***Note:*** EP does not support ASPM L1 transition		000b	RO
14 :12	L0EL	**L0s Exit Latency**- Indicates the exit latency from L0s to L0 state. 000b - Less than 64ns 001b - 64ns - 128ns 010b - 128ns - 256ns 011b - 256ns - 512ns 100b - 512ns - 1 ìs 101b - 1 ìs - 2 ìs 110b - 2 ìs - 4 ìs 111b - Reserved		000b	RO
11 :10	ASLPM	**Active State Link PM Support** - Indicates the level of active state power management supported in EP. Defined encodings are: 00b - Reserved 01b - L0s Entry Supported 10b - Reserved 11b - L0s and L1 Supported		11b	RO
9 :4	LINKW	Max Link Width- Indicates the max link width. Relevant encoding: 000000b - Reserved 000001b - x1 000010b - x2 000100b - x4 001000b - x8 001100b - x12 010000b - x16 100000b - x32 EP value depends on SKU. However the max link width is x16.		010000b	RO
3 :0	MAXSPEED	Max Link Speed - Indicates Maximum supported Link Speed. Defined encodings are: 0001b - 2.5Gbs Link speed supported (Gen 1) 0010b - 5.0Gbs Link speed supported (Gen 2) EP indicates a max Link Speed of 5.0 Gbs.		0010b	RO

39.12.5.8 PLCNTLR - PF Link Control Register

Not supported in EP.

Table 39-42. PLCNTLR - PF Link Control Register (Sheet 1 of 2)

Description: Link Control Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 84h **Offset End:** 85h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 84h **Offset End:** 85h	
Size: 16 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :12	Reserved	Reserved		0000b	RO
11	LBWINTE	**Link Autonomous Bandwidth Interrupt Enable** - When set to 1b this bit enables the generation of an interrupt to indicate that the Link Autonomous Bandwidth Status bit has been set. ***Note:*** Defined for PCIe Compliance		0b	RO
10	LBWMINTE	**Link Bandwidth Management Interrupt Enable** - When set to 1b this bit enables the generation of an interrupt to indicate that the Link Bandwidth Management Status bit has been set. ***Note:*** Defined for PCIe Compliance		0b	RO
9	WD	**Hardware Autonomous Width Disable** - When set to 1b this bit disables hardware from changing the link width for reasons other than attempting to correct unreliable link operation by reducing link width. ***Note:*** Defined for PCIe Compliance		0b	RO
8	ECLKPM	**Enable Clock Power Management** - Defined for PCIe Compliance		0b	RO
7	EXTSYNC	**Extended Sync**. When set to one, this bit forces the transmission of: - 4096 FTS ordered sets during the L0s state - followed by a single SKP ordered set prior to entering the L0 state, - as well as the transmission of 1024 TS1 ordered sets in the L1 state prior to entering the Recovery state. This mode gives external devices (e.g., logic analyzers) that may be monitoring Link activity time to achieve bit and symbol lock before the Link enters the L0 or Recovery state and resumes communication. Defined for PCIe Compliance		0b	RW
6	CCLKCFG	**Common Clock Configuration** - when set indicates that EP and the root port at the other end of the link are operating with a common reference clock. A value of 0 indicates that they operating with an asynchronous clock. This parameter affects the L0s Exit Latencies. After changing the value in this bit in both components on a Link, software must trigger the Link to retrain by writing a 1b to the Retrain Link bit. Defined for PCIe Compliance		0b	RW
5	RETRAIN	**Retrain Link**: A write of 1b to this bit initiates Link retraining by directing the Physical Layer LTSSM to the Recovery state. ***Note:*** Defined for PCIe Compliance. Does not apply to endpoint		0b	RO

Table 39-42. PLCNTLR - PF Link Control Register (Sheet 2 of 2)

Description: Link Control Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 84h **Offset End:** 85h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 84h **Offset End:** 85h	
Size: 16 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
4	LINKDIS	**Link Disable**: This bit disables the Link by directing the LTSSM to the Disabled state when set to 1b. ***Note:*** Defined for PCIe Compliance. Does not apply to endpoint		0b	RO
3	RCB	**Read Completion Boundary (RCB)**: For PCIe Endpoints this field is set optionally by configuration software to indicate the RCB value of the Root Port upstream from the Endpoint. – 0b = 64B – 1b = 128 byte ***Note:*** Defined for PCIe Compliance		0b	RW
2	Reserved2	Reserved		0b	RO
1 :0	ASPMC	**Active State Link PM Control** - this field controls the active state PM supported on the link. Link PM functionality is determined by the lowest common denominator of all functions. Defined encodings are: 00b - PM Disabled 01b - L0s Entry Supported 10b - Reserved 11b - L0s and L1 Supported Defined for PCIe Compliance		00b	RW

39.12.5.9 PLSR– PF Link Status Register

Not supported in EP

Table 39-43. PLSR - PF Link Status Register

Description: Link Status Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 86h **Offset End:** 87h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 86h **Offset End:** 87h	
Size: 16 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	Reserved	Reserved		0b	RO
14	LBWMS	**Link Bandwidth Management Status** ***Note:*** Defined for PCIe Compliance		0b	RO
13	DLACT	**Data Link Layer Link Active** ***Note:*** Defined for PCIe Compliance		0b	RO
12	SCLKCFG	**Slot Clock Configuration** - When set indicates that EP uses the physical reference clock that the platform provides on the connector. This bit must be cleared if EP uses an independent clock. Defined for PCIe Compliance		0b	RW/O
11	LTINPROG	**Link Training** - Indicates that link training is in progress. ***Note:*** Defined for PCIe Compliance, Does not apply to PCIe Endpoints		0b	RO
10	LTE	**Link Training Error** - Indicates that a link training error has occurred. ***Note:*** Defined for PCIe Compliance		0b	RO
9 :4	NLW	**Negotiated Link Width**: Negotiated Link Width - Indicates the negotiated width of the link. Relevant encoding for EP are: 000001b - x1 000010b - x2 000100b - x4 001000b - x8 010000b - x16 Defined for PCIe Compliance		010000b	RO
3 :0	NLS	**Negotiated Link Speed**: The negotiated Link Speed. Defined encodings are: 0001b - 2.5Gbs 0010b - 5.0Gbs Defined for PCIe Compliance		0010b	RO

39.12.5.10 PDCAPR2 - PF Device Capabilities 2 Register

Table 39-44. PDCAPR2 - PF Device Capabilities 2 Register

Description: Device Capabilities 2 Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** 98h **Offset End:** 9Bh	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 98h **Offset End:** 9Bh	
Size: 32 bit	**Default:** 00000012h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :5	Reserved	**Reserved**		0h	RO
4	CTODS	Completion Timeout Disable Supported. A value of 1b indicates support for the completion timeout disable mechanism.		1b	RO
3 :0	CTORS	Completion Timeout Ranges Supported: This field indicates support for the optional completion timeout programmability mechanism. This mechanism enables system software to modify the completion timeout value. Four time value ranges are defined: • Range A = 50 us to 10 ms • Range B = 10 ms to 250 ms • Range C = 250 ms to 4 s • Range D = 4 s to 64 s Bits are set according to the following table to show the timeout value ranges that are supported. • 0000b = Completion timeout programming not supported. • 0001b = Range A. • 0010b = Range B. • 0011b = Ranges A and B. • 0110b = Ranges B and C. • 0111b = Ranges A, B and C. • 1110b = Ranges B, C and D. • 1111b = Ranges A, B, C and D. • All other values are reserved. It is strongly recommended that the completion timeout mechanism not expire in less than 10 ms		0010b	RO

39.12.5.11 PDCNTR2 - PF Device Control 2 Register

Table 39-45. PDCNTR2 - PF Device Control 2 Register

Description: Device Control 2 Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** 9Ch **Offset End:** 9Dh	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 9Ch **Offset End:** 9Dh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 5	Reserved0	Reserved0		0h	RO
4	CTODIS	Completion Timeout Disable: When set to 1b, this bit disables the completion timeout mechanism. Software is permitted to set or clear this bit at any time. When set, the completion timeout detection mechanism is disabled. If there are outstanding requests when the bit is cleared, it is permitted but not required for hardware to apply the completion timeout mechanism to the outstanding requests. If this is done, it is permitted to base the start time for each request on either the time this bit was cleared or the time each request was issued. The default value for this bit is 0b.		0b	RW
3 :0	CTOV	Completion Timeout Value: In devices that support completion timeout programmability, this field enables system software to modify the completion timeout value. Encoding: • 0000b = Default range: 50 us to 50 ms. It is strongly recommended that the completion timeout mechanism not expire in less than 10 ms. Values available if Range A (50 us to 10 ms) programmability range is supported: • 0001b = 50us to 100 us. • 0010b = 1 ms to 10 ms. Values available if Range B (10 ms to 250 ms) programmability range is supported: • 0101b = 16 ms to 55 ms. • 0110b = 65 ms to 210 ms. Values available if Range C (250 ms to 4 s) programmability range is supported: • 1001b = 260 ms to 900 ms. • 1010b = 1 s to 3.5 s. Values available if the Range D (4 s to 64 s) programmability range is supported: • 1101b = 4 s to 13 s. • 1110b = 17 s to 64 s. Values not defined are reserved. Software is permitted to change the value in this field at any time. For requests already pending when the completion timeout value is changed, hardware is permitted to use either the new or the old value for the outstanding requests and is permitted to base the start time for each request either when this value was changed or when each request was issued. The default value for this field is 0000b.		0000b	RW

39.12.5.12 PLCAPR2 - PF Link Capabilities 2 Register

Table 39-46. PLCAPR2 - PF Link Capabilities 2 Register

Description: Link Capabilities 2 Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** A0h **Offset End:** A3h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** A0h **Offset End:** A3h	
Size: 32 bit	**Default:** 4h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :8	Reserved	Reserved		0h	RO
7 :0	Reserved	Reserved		4h	RO

39.12.5.13 PLCNTLR2 - PF Link Control2 Register

Table 39-47. PLCNTLR2 - PF Link Control 2 Register (Sheet 1 of 2)

Description: Link Control2 Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** A4h **Offset End:** A5h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** A4h **Offset End:** A5h	
Size: 16 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :13	Reserved	Reserved		0h	RW
12	CDE	Compliance De-emphasis – This bit sets the de-emphasis level in Polling.Compliance state if the entry occurred due to the Enter Compliance bit being 1b. Encodings: 1b -3.5 dB 0b -6 dB	Y	0b	RW
11	CSOS	Compliance SOS – When set to 1b, the LTSSM is required to send SOS periodically in between the (modified) compliance patterns.	Y	0b	RW
10	EMC	Enter Modified Compliance – When this bit is set to 1b, the device transmits modified compliance pattern if the LTSSM enters Polling.Compliance state.	Y	0b	RW
9 7	TMARG	Transmit Margin – This field controls the value of the non deemphasized voltage level at the Transmitter pins. Encodings: 000b - Normal operating range 001b - 800-1200 mV for full swing and 400-700 mV for half-swing. 010b - (n-1) - Values must be monotonic with a non-zero slope. The value of n must be greater than 3 and less than 7. At least two of these must be below the normal operating range of n: 200-400 mV for full-swing and 100-200 mV for half-swing. other values - reserved	Y	0b	RW
6	SDEM	Selectable De-emphasis This bit is not applicable and reserved for Endpoints.		0b	RO

Table 39-47. PLCNTLR2 - PF Link Control 2 Register (Sheet 2 of 2)

Description: Link Control2 Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** A4h **Offset End:** A5h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** A4h **Offset End:** A5h	
Size: 16 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
5	HWAUTOSD	Hardware Autonomous Speed Disable. When set to 1b, this bit disables hardware from changing the link speed for reasons other than attempting to correct unreliable link operation by reducing link speed. Hard wire to 0b.		0b	RO
4	ENCOMP	Enter Compliance. Software is permitted to force a link to enter compliance mode at the speed indicated in the Target Link Speed field by setting this bit to 1b in both components on a link and then initiating a hot reset on the link. The default value of this field following a fundamental reset is 0b.	Y	0b	RW
3 :0	TLNKS	Target Link Speed. This field is used to set the target compliance mode speed when software is using the Enter Compliance bit to force a link into compliance mode. Defined encodings are: 0001b = 2.5 Gb/s Target Link Speed. 0010b = 5 Gb/s Target Link Speed. All other encodings are reserved. If a value is written to this field that does not correspond to a speed included in the Supported Link Speeds field, the result is undefined. The default value of this field is the highest link speed supported by the EP (as reported in the Supported Link Speeds field of the Link Capabilities register).	Y	0010b	RW

39.12.5.14 PLSR2– PF Link Status 2 Register

Table 39-48. PLSR2 - PF Link Status 2 Register

Description: Link Status Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** A6h **Offset End:** A7h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** A6h **Offset End:** A7h	
Size: 16 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :1	Reserved	**Reserved**		0h	RO
:0	CDEL	Current De-emphasis Level – When the Link is operating at 5 GT/s speed, this bit reflects the level of de-emphasis. it is undefined when the Link is operating at 2.5 GT/s speed Encodings: 1b -3.5 dB 0b -6 dB		0b	RO

39.12.6 MSI Capability Structure

39.12.6.1 PMSICID - PF Message Signaled Interrupt Capability ID Register

The Message Signaled Interrupt Capability record defines how the device generates PCI MSI messages. It is a PCI SIG-defined capability record and includes the MCID, MCP, MCTL, MADR, and MDATA fields of the configuration header.

Table 39-49. PMSICID - PF Message Signaled Interrupt Capability ID Register

Description: Message Signaled Interrupt Capability ID Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** B0h **Offset End:** B0h	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** B0h **Offset End:** B0h	
Size: 8 bit	**Default:** 05h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	MCID	**Capability ID:** PCI SIG assigned capability record ID (05h, MSI capability)		05h	RO

39.12.6.2 PMSINCP - PF Message Signaled Interrupt Next Capability Pointer Register

Table 39-50. PMSINCP - PF Message Signaled Interrupt Next Capability Pointer Register

Description: Message Signaled Interrupt Next Capability Pointer Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** B1h **Offset End:** B1h	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** B1h **Offset End:** B1h	
Size: 8 bit	**Default:** 60h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	MCP	**Next Capability Pointer:** MSI-X capability.		60h	RO

39.12.6.3 PMSICTL - PF Message Signaled Interrupt Control Register

Table 39-51. PMSICTL - PF Message Signaled Interrupt Control Register

Description: Message Signaled Interrupt Control Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** B2h **Offset End:** B3h	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** B2h **Offset End:** B3h	
Size: 16 bit	**Default:** 0180h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :09	Reserved	Reserved		0h	RO
08	MC	**Per-Vector Masking Capable:** Per-vector masking capable.		1b	RO
07	C64	**64 bit Address Capable:** Hardwired to 1 to indicate the device generate 64b message addresses.		1b	RO
06 :04	MME	**Multiple Message Enable:** System software writes to this field to indicate the number of allocated messages (less than or equal to the number of requested messages in MMC). A value of 0 corresponds to one message.		000h	RW
03 :01	MMC	**Multiple Message Capable:** System software reads this field to determine the number of requested messages. Hardwired to 0 to request one message.		000h	RO
00	MSIE	MSI Enable: System software sets this bit to enable MSI signaling. A device driver is prohibited from writing this bit to mask a device's service request. If 1, the device can use an MSI to request service. If 0, the device cannot use an MSI to request service.		0h	RW

39.12.6.4 PMSILADDR - PF Message Signaled Interrupt Lower Address Register

Table 39-52. PMSILADDR - PF Message Signaled Interrupt Lower Address Register

Description: Message Signaled Interrupt Lower Address Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** B4h **Offset End:** B7h	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** B4h **Offset End:** B7h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :02	ADDR	**Lower Message Address:** Written by the system to indicate the lower 30-bits of the address to use for the MSI memory write transaction.		0h	RW
01 :00	Reserved	**Reserved**		00b	RO

39.12.6.5 PMSIUADDR - PF Message Signaled Interrupt Upper Address Register

Table 39-53. PMSIUADDR - PF Message Signaled Interrupt Upper Address Register

Description: Message Signaled Interrupt Upper Address Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** B8h **Offset End:** BBh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** B8h **Offset End:** BBh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	ADDR	**Upper Message Address:** Written by the system to indicate the lower 62-bits of the address to use for the MSI memory write transaction. The lower two bits will always be written as 0.		0h	RW

39.12.6.6 PMSIDATA - PF Message Signaled Interrupt Data Register

Table 39-54. PMSIDATA - PF Message Signaled Interrupt Data Register

Description: Message Signaled Interrupt Data Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** BCh **Offset End:** BDh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** BCh **Offset End:** BDh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	DATA	**Message Data:** Written by the system to indicate the lower 16 bits of the data written in the MSI memory write DWORD transaction. The upper 16 bits of the transaction are written as 0.		0h	RW

39.12.6.7 PMSIMSK - PF Message Signaled Interrupt Mask Register

Table 39-55. PMSIMSK - PF Message Signaled Interrupt Mask Register

Description: Message Signaled Interrupt Mask Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** C0h **Offset End:** C3h	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** C0h **Offset End:** C3h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :01	Reserved	Reserved.		0h	RO
00 :00	MASK0	**Mask Bits:** Only on bit defined. Refer to the interrupt section.		0b	RW

39.12.6.8 PMSIPND - PF Message Signaled Interrupt Pending Register

Table 39-56. PMSIPND - PF Message Signaled Interrupt Pending Register

Description: Message Signaled Interrupt Pending Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** C4h **Offset End:** C7h	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** C4h **Offset End:** C7h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :01	Reserved	Reserved.		0h	RO
00 :00	PNDB	**Pending Bits.**		0b	RO/V

39.12.7 PF Advanced Error Reporting Capability Structure

39.12.7.1 PPCIEAERCAPID - PF PCI Express AER Capability ID Register

The PCI Express Capability List register enumerates the PCI Express AER Capability structure in the PCI 3.0 configuration space capability list. The AER Capability can be the last capability in the list. Refer to the "PPCIEAERCAPID - PF PCI Express AER Capability ID Register" which describes the next capability pointer value.

Table 39-57. PPCIEAERCAPID - PF PCI Express AER Capability ID Register

Description: PCI Express AER Capability ID Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 100h **Offset End:** 103h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 100h **Offset End:** 103h	
Size: 32 bit	**Default:** 13810001h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :20	PCIEAERNCP	Next PCI Express Extended Capability Pointer		138h	RO
19 :16	PCIEAERCVN	Advanced Error Capability Version Number: PCI Express Advanced Error Reporting Extended Capability Version Number.		1h	RO
15 :00	PCIEAERCID	Advanced Error Capability ID: PCI Express Extended Capability ID indicating Advanced Error Reporting Capability.		1h	RO

39.12.7.2 PPAERUCS - PF PCI Express AER Uncorrectable Error Status Register

Table 39-58. PPAERUCS - PF PCI Express AER Uncorrectable Error Status Register

Description: PCI Express AER Uncorrectable Error Status Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 104h **Offset End:** 107h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 104h **Offset End:** 107h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :23	Reserved0	Reserved0:		00b	RO
:22	IEUNC	**Uncorrectable Internal Error Status**		00b	RW/1C/V
:21	Reserved3	Reserved3		00b	RO
20	UR	**Unsupported Request Error Status:** As a receiver, set whenever an unsupported request is detected. The Header is logged.		0b	RW/1C/V
19	ECRCC	**ECRC Check:** As a receiver, set when ECRC check fails. The Header is logged.		0b	RW/1C/V
18	MTLP	**Malformed TLP**: As a receiver, set whenever a malformed TLP is detected. The Header is logged.		0b	RW/1C/V

Table 39-58. PPAERUCS - PF PCI Express AER Uncorrectable Error Status Register

Description: PCI Express AER Uncorrectable Error Status Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 104h **Offset End:** 107h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 104h **Offset End:** 107h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
17	RO	**Receiver Overflow:** Set if PCI Express receive buffers overflow.		0b	RO
16	EC	**Unexpected Completion:** As a receiver, set whenever a completion is received that does not match the requester ID or outstanding Tag. The Header is logged.		0b	RW/1C/V
15	CA	**Completer Abort:** As a completer, set whenever an internal agent signals a data abort. The header is logged.		0b	RW/1C/V
14	CT	**Completion Timeout**: As a requester, set whenever an outbound Non Posted Request does not receive a completion within 16-32ms.		0b	RW/1C/V
13	FCPES	**Flow Control Protocol Error Status:** Set whenever a flow control protocol error is detected. Not supported.		0b	RO
12	PTLPR	**Poisoned TLP Received:** As a receiver, set whenever a poisoned TLP is received from PCI Express. The header is logged. Note that internal queue errors are not covered by this bit, they are logged by the Configuration target of the transaction.		0h	RW/1C/V
11 :6	Reserved1	Reserved1		0000b	RO
5	SDES	**Surprise Down Error:** Not supported.		0b	RO
4	DLPE	**Data Link Protocol Error:** Set whenever a data link protocol error is detected.		0b	RO
03 :00	Reserved2	Reserved		0h	RO

39.12.7.3 PPAERUCM - PF PCI Express AER Uncorrectable Error Mask Register

Table 39-59. PPAERUCM - PF PCI Express AER Uncorrectable Error Mask Register

Description: PCI Express AER Uncorrectable Error Mask Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 108h **Offset End:** 10Bh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 108h **Offset End:** 10Bh	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :23	Reserved0	Reserved0		0h	RO
22	IEUNC	**Uncorrectable Internal Error Mask**: When '1' error reporting is masked.		0b	RW
21	Reserved3	Reserved3		0b	RO
20	UR	**Unsupported Request Error Mask**: When '1' error reporting is masked.		0b	RW
19	ECRCC	**ECRC Check Error Mask**: When '1' error reporting is masked.		0b	RW
18	MTLP	**Malformed TLP Error Mask**: When '1' error reporting is masked.		0b	RW
17	RO	**Receiver Overflow Error Mask**: When '1' error reporting is masked.		0b	RO
16	EC	**Unexpected Completion Error Mask**: When '1' error reporting is masked.		0b	RW
15	CA	**Completer Abort Error Mask**: When '1' error reporting is masked.		0b	RW
14	CT	**Completion Time Out Error Mask**: When '1' error reporting is masked.		0b	RW
13	FCPES	**Flow Control Protocol Error Mask**: When '1' error reporting is masked. Not supported.		0b	RO
12	PTLPR	**Poisoned TLP Received Error Mask**: When '1' error reporting is masked.		0h	RW
11 :6	Reserved1	Reserved1		0h	RO
5	SDES	**Surprise Down Error:** Not supported.		0b	RO
4	DLPE	**Data Link Protocol Error Mask**: When '1' error reporting is masked.		0b	RO
03 :00	Reserved2	Reserved2		0h	RO

39.12.7.4 PPAERUCSEV - PF PCI Express AER Uncorrectable Error Severity Register

The Uncorrectable Error Severity register controls whether an individual uncorrectable error is reported as a non-fatal or fatal error. An error is reported as fatal when the corresponding error bit in the severity register is set. If the bit is cleared, the corresponding error is considered non-fatal.

Table 39-60. PPAERUCSEV - PF PCI Express AER Uncorrectable Error Severity Register

Description: PCI Express AER Uncorrectable Error Severity Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 10Ch **Offset End:** 10Fh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 10Ch **Offset End:** 10Fh	
Size: 32 bit	**Default:** 00440000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :23	Reserved0	reserved0		0h	RO
22	IEUNC	Uncorrectable Internal Error status severity		1b	RW
21	Reserved3	Reserved3		0b	RO
20	UR	Unsupported request error status severity (URESS):		0b	RW
19	ECRCC	ECRC check severity:		0b	RW
18	MTLP	malformed tlp severity:		1b	RW
17	RO	receiver overflow severity:		0b	RW
16	EC	unexpected completion severity:		0b	RW
15	CA	completer abort severity:		0b	RW
14	CT	completion time out severity:		0b	RW
13	FCPES	flow control protocol error severity: not supported.		0b	RO
12	PTLPR	poisoned tlp received severity:		0h	RW
11 :6	Reserved1	reserved1		0h	RO
5	SDES	surprise down error severity: not supported.		0h	RO
4	DLPE	data link protocol error severity:		0h	RW
03 :00	Reserved2	reserved2		0h	RO

39.12.7.5 PPAERCS - PF PCI Express AER Correctable Error Register

Table 39-61. PPAERCS - PF PCI Express AER Correctable Error Register

Description: PCI Express AER Correctable Error Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** 110h **Offset End:** 113h	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 110h **Offset End:** 113h	
Size: 32 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :15	Reserved0	Reserved0		0h	RO
14	IECOR	Corrected Internal Error Status		0b	RW/1C/V
13	ANFES	Advisory Non Fatal Error status		0b	RW/1C/V
12	RTTS	Replay Timer Timeout Status: Set whenever a replay timer timeout occurs. Not supported		0b	RO
11 :09	Reserved1	Reserved1		0b	RO
8	RNRS	REPLAY_NUM Rollover Status: Set whenever the replay number rolls over from 11 to 00. Not supported		0h	RO
7	BDLLPS	Bad DLLP Status: Sets this bit on CRC errors on DLLP. Not supported		0b	RO
6	DLPE	Bad TLP Status: Sets this bit on CRC errors or sequence number out of range on TLP. Not supported		0b	RO
05 :01	Reserved2	Reserved2		0h	RO
00	RES	Receiver Error Status: Set whenever the physical layer detects a receiver error. Not supported		0b	RO

39.12.7.6 PPAERCM - PF PCI Express AER Correctable Error Mask Register

Table 39-62. PPAERCM - PF PCI Express AER Correctable Error Mask Register

Description: PCI Express AER Correctable Error Mask Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 114h **Offset End:** 117h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 114h **Offset End:** 117h	
Size: 32 bit	**Default:** 2000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :15	Reserved0	Reserved0		0h	RO
14	IECOR	Corrected Internal Error Status		1b	RW
13	ANFES	**Advisory Non-Fatal Error Mask**: this bit is set by default to enable compatibility with software that does not comprehend Role-Based Error Reporting.		1b	RW
12	RTTS	**Replay Timer Timeout Mask** Not supported		0b	RO
11 :09	Reserved1	**Reserved1**		0h	RO
8	RNRS	**REPLAY NUM Rollover Mask:** Not supported		0b	RO
7	BDLLPS	**Bad DLLP Mask:** Not supported		0b	RO
6	DLPE	**Bad TLP Mask:** Not supported		0b	RO
05 :01	Reserved2	Reserved2		0h	RO
00	RES	**Receiver Error Mask:** Not supported		0b	RO

39.12.7.7 PPAERCTLCAP - PF PCI Express AER Control and Capability Register

Table 39-63.PPAERCTLCAP - PF PCI Express AER Control and Capability Register

Description: PCI Express AER Control and Capability Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** 118h **Offset End:** 11Bh	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 118h **Offset End:** 11Bh	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :09	Reserved	Reserved0		0h	RO
8	ECRCCE	**ECRC Check Enable**: When set enables ECRC checking.		0b	RW
7	ECRCCC	**ECRC Check Capable**: Indicates EP is capable of checking ECRC.		1b	RO
6	ECRCGE	**ECRC Generation Enable**: When set enables ECRC generation.		0b	RW
5	ECRCGC	**ECRC Generation Capable**: Indicates the EP is not capable of generating ECRC.		1b	RO
04 :00	TFEP	**The First Error Pointer:** Identifies the bit position of the first error reported in the Section 39-58, "PPAERUCS - PF PCI Express AER Uncorrectable Error Status Register" This register will not update until all bits in the ERRUNC STS register are cleared.		00000b	RO/V

39.12.7.8 PPAERHDRLOG0 - PF PCI Express AER Header Log 0 Register

Table 39-64.PPAERHDRLOG0 - PF PCI Express AER Header Log 0 Register

Description: PCI Express AER Header Log 0 Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** 11Ch **Offset End:** 11Fh	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 11Ch **Offset End:** 11Fh	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HDRLOGDW0	1st DWord of the Header for the PCI Express packet in error (HDRLOGDW0): Once an error is logged in this register, it remains locked for further error logging until the time the software clears the status bit that cause the header log i.e., the error pointer is rearmed to log again.		0h	RO/V

39.12.7.9 PPAERHDRLOG1 - PF PCI Express AER Header Log 1 Register

Table 39-65.PPAERHDRLOG1 - PF PCI Express AER Header Log 1 Register

Description: PCI Express AER Header Log 1 Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 120h **Offset End:** 123h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 120h **Offset End:** 123h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HDRLOGDW1	2nd DWord of the Header for the PCI Express packet in error (HDRLOGDW1): Once an error is logged in this register, it remains locked for further error logging until the time the software clears the status bit that cause the header log i.e., the error pointer is rearmed to log again.		0h	RO/V

39.12.7.10 PPAERHDRLOG2 - PF PCI Express AER Header Log 2 Register

Table 39-66.PPAERHDRLOG2 - PF PCI Express AER Header Log 2 Register

Description: PCI Express AER Header Log 2 Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 124h **Offset End:** 127h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 124h **Offset End:** 127h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HDRLOGDW2	3rd DWord of the Header for the PCI Express packet in error (HDRLOGDW2): Once an error is logged in this register, it remains locked for further error logging until the time the software clears the status bit that cause the header log i.e., the error pointer is rearmed to log again.		0h	RO/V

39.12.7.11 PPAERHDRLOG3 - PF PCI Express AER Header Log 3 Register

Table 39-67.PPAERHDRLOG3 - PF PCI Express AER Header Log 3 Register

Description: PCI Express AER Header Log 3 Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 128h **Offset End:** 12Bh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 128h **Offset End:** 12Bh	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HDRDWLOG3	4th DWord of the Header for the PCI Express packet in error (HDRDWLOG3): Once an error is logged in this register, it remains locked for further error logging until the time the software clears the status bit that cause the header log i.e., the error pointer is rearmed to log again.		0h	RO/V

39.12.8 PF Alternative Routing-ID Extended Capability Structure

This section describes the PCI Express Extended Configuration Space registers that make up the Alternative Routing ID Extended Capability Structure.

Some information from the specification is repeated here as an aid to the reader or to describe implementation choice. Refer to the *PCI Express Base Specification 2.0* for the full register descriptions and additional information regarding their operation.

39.12.8.1 PARIDHDR - PF Alternative Routing ID Capability Header

This register contains information associated with the Alternative Routing ID capability. This is compliant with the *PCI-SIG ECN: Alternatifve Routing-ID Interpretation (ARI)*, Updated June4,2007.

Table 39-68. PARIDHDR - PF Alternative Routing ID Capability Header

Description: Alternative Routing ID Capability Header					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 138h **Offset End:** 13Bh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 138h **Offset End:** 13Bh	
Size: 32 bit	**Default:** 1401000Eh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :20	ARINCO	Next Capability Offset - This field contains 140h which points to the next item in the extended capabilities list, the SR-IOV extended capability.		140h	RO
19 :16	ARICV	Capability Version - This is set to 1h for the most current version of the specification.		1h	RO
15 :0	ARIECI	PCI Express Extended Capability ID - The PCI SIG has assigned 000Eh to the ARI extended capability.		000EH	RO

39.12.8.2 PFARICAP - PF ARI Capabilities Register

Table 39-69. PFARICAP - PF ARI Capabilities Register

Description: ARI Capabilities Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 13Ch **Offset End:** 13Dh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 13Ch **Offset End:** 13Dh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :8	VNFN	**Next Function Number:** The function number of the next highest numbered PF in a multi-function device. Only one physical function exists. Therefore, value = 0h.		0h	RO
7 :2	Reserved	Reserved.		0h	RO
1	ACS	**ACS Functional Groups Capability**: Hardwired to Zero.		0b	RO
0	MFVC	**MFVC Functional Groups Capability**: Hardwired to Zero		0b	RO

39.12.8.3 PARIDCTL - PF Alternative Routing ID Control Register

This register contains information associated with the Alternative Routing ID capability.

Table 39-70. PARIDCTL - PF Alternative Routing ID Control Register

Description: Alternative Routing ID Control Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 13Eh **Offset End:** 13Fh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 13Eh **Offset End:** 13Fh	
Size: 16 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :7	Reserved0	Reserved0		0h	RO
6 :4	FG	**Function Group:** Hardwired to Zero as EP does not support Function Groups.		0b	RO
3 :2	Reserved1	Reserved1		0b	RO
1	ACS	**ACS Functional Groups Enable**: Hardwired to Zero as EP does not support.		0b	RO
0	MFVC	**MFVC Functional Groups Enable**: Hardwired to Zero as EP does not support.		0b	RO

39.12.9 PF SR-IOV Extended Capability Structure

39.12.9.1 PSRIOVCAPID - PF SR-IOV Capability ID Register

Table 39-71. PSRIOVCAPID - PF SR IOV Capability ID Register

Description: SR IOV Capability ID Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 140h **Offset End:** 141h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 140h **Offset End:** 141h	
Size: 16 bit	**Default:** 10h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	SRIOVOD	**Capability ID:** PCI SIG assigned a capability record ID for SR-IOV per the 1.1 revision		0010h	RO

39.12.9.2 PSRIOVCVNC - PF SRIOV Capability Version and Next Capability Pointer Register

Table 39-72. PSRIOVCVNC - PF SRIOV Capability Version and Next Capability Pointer Register

Description: SRIOV Capability Version and Next Capability Pointer Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 142h **Offset End:** 143h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 142h **Offset End:** 143h	
Size: 16 bit	**Default:** 1B01h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :4	SRIOVNCO	Next Capability Offset.		1B0h	RO
3 :0	SRIOVCV	Capability Version - This is set to 1h for the Single Root I/O Virtualization and Sharing Specification, Revision 1.1.		1h	RO

39.12.9.3 PSRIOVCAP - PF SRIOV Capabilities Register

Table 39-73. PSRIOVCAP - PF SRIOV Capabilities Register

Description: SRIOV Capabilities Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 144h **Offset End:** 147h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 144h **Offset End:** 147h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :21	VFMINT	VF Migration Interrupt Message Number - Indicates the MSI vector used for migration interrupts. The value in this field has no significance if bit 0 of this capability is Clear. ***Note:*** N/A for EP		0h	RO
20 :1	Reserved	Reserved.		0h	RO
:0	VFMCAP	VF Migration Capable - If Set, the PF is Migration Capable and operating under a Migration Capable MR-PCIM. ***Note:*** EP does not support VF migration		0b	RO

39.12.9.4 PSRIOVCS - PF SRIOV Control and Status Register

Table 39-74. PSRIOVCS - PF SRIOV Control and Status Register

Description: SRIOV Control and Status Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 148h **Offset End:** 14Bh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 148h **Offset End:** 14Bh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :5	Reserved	Reserved		0h	RO
4	VFARI	ARI Capable Hierarchy - Device may locate VFs in Function numbers 8 to 255 of the captured Bus number. EP supports ARI and will locate the VF1-VF16 in Function 8 through Function 23.		0b	RW
3	VFMSE	VF MSE - Memory Space Enable for Virtual Functions.		0b	RW
2	VFMIE	VF Migration Interrupt Enable - Enables / Disables VF Migration State Change Interrupt. ***Note:*** EP does not support VF Migration		0b	RO
1	VFME	VF Migration Enable - Enables / Disables VF Migration Support. ***Note:*** EP does not support VF Migration		0b	RO
0	VFE	VF Enable - Enables / Disables VFs. Default value is 0b		0b	RW

39.12.9.5 PSRIOVMTOTINI - PF SRIOV Initial and Total VF's Register

Table 39-75. PSRIOVMTOTINI - PF SRIOV Initial and Total VF's Register

Description: SRIOV Initial and Total VF's Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 14Ch **Offset End:** 14Fh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 14Ch **Offset End:** 14Fh	
Size: 32 bit	**Default:** 00100010h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :16	TOTVF	TotalVFs indicates the maximum number of VFs that could be associated with the PF. Since EP does not support VF migration this field must be equal to the INITVF field		0010h	RO
15 :0	INITVF	InitialVFs indicates the number of VFs that are initially associated with the PF. Since EP does not support VF migration this field must be equal to the TOTVF field		0010h	RO

39.12.9.6 PSRIOVNUMVF - PF SRIOV Number of VF's Register

Table 39-76. PSRIOVNUMVF - PF SRIOV Number of VF's Register

Description: SRIOV Number of VF's Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 150h **Offset End:** 153h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 150h **Offset End:** 153h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	Reserved	Reserved		00h	RO
23 :16	FUNDEP	This field describes dependencies between PFs. VF dependencies are the same as the dependencies of their associated PFs. If a PF is independent from other PFs of a Device, this field shall contain the Function Number of the PF. ***Note:*** EP implements a single PF and therefore this field is N/A to EP.		00h	RO
15 :0	NUMVF	Num VFs defines the number of VFs software has assigned to the PF. Software sets NumVFs to any value between 1 and TotalVFs (16 for EP) as part of the process of creating VFs. NumVFs VFs shall be visible in the PCI-Express fabric after both NumVFs is set to a valid value and VF Enable is set to one. 'Visible in the PCI-Express fabric' means that the VF shall respond to PCI Express transactions targeting the VF, following all other rules defined by this specification and the base specification.		0h	RW

39.12.9.7 PSRIOVFVFO - PF SRIOV First VF Offset Register

Table 39-77. PSRIOVFVFO - PF SRIOV First VF Offset Register

Description: SRIOV First VF Offset Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** 154h **Offset End:** 155h	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 154h **Offset End:** 155h	
Size: 16 bit	**Default:** 0008h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :0	VFOFFSET	First VF Offset is a constant and defines the Routing ID (RID) offset of the first VF that is associated with the PF that contains this Capability structure. The first VFs 16-bit RID is calculated by adding the contents of this field to the RID of the PF. ***Note:*** For the EP, the RID of the first VF is at an offset of eight from the RID of the Physical Function (PF).		0008h	RO

39.12.9.8 PSRIOVVFS - PF SRIOV VF Stride Register

Table 39-78. PSRIOVVFS - PF SRIOV VF Stride Register

Description: SRIOV VF Stride Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** 156h **Offset End:** 157h	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 156h **Offset End:** 157h	
Size: 16 bit	**Default:** 0001h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :0	VFSTRIDE	VF Stride is a constant and defines the Routing ID (RID) offset from one VF to the next one for all VFs associated with the PF that contains this Capability structure. The next VFs 16-bit RID is calculated by adding the contents of this field to the RID of the current VF. ***Note:*** For the EP, the stride between the RIDs of subsequent VFs is one.		0001h	RO

39.12.9.9 PSRIOVFDID - PF SRIOV VF Device ID Register

Table 39-79. PSRIOVFDID - PF SRIOV VF Device ID Register

Description: SRIOV VF Device ID Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 158h **Offset End:** 15Bh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 158h **Offset End:** 15Bh	
Size: 32 bit	**Default:** 37C90000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :23	VFDID151HIGH	VF Device ID - The Device ID that is presented to the OS for every VF. 37C9h for EP		001101111b	RO/V
22 :16	VFDID151LOW	VF Device ID - The Device ID that is presented to the OS for every VF.		1001001b	RO
15 :00	Reserved	Reserved		0h	RO

39.12.9.10 PSRIOVPAGESIZE - PF SRIOV Supported Page size Register

Table 39-80. PSRIOVPAGESIZE - PF SRIOV Supported Page Size Register

Description: SRIOV Supported Page Size Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 15Ch **Offset End:** 15Fh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 15Ch **Offset End:** 15Fh	
Size: 32 bit	**Default:** 00000553h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	FIRSTVFO	This field must define the supported page sizes. This PF supports a page size of 2^(n+12) if bit n is Set. For example, if bit 0 is Set, the PF supports 4k byte page sizes. PFs are required to support 4k, 8k, 64k, 256k, 1M and 4M page sizes. All other page sizes are optional. A page size describes the minimum alignment requirements for VF BAR resources as described in "System Page Size".		00000553h	RO

39.12.9.11 PSRIOVSYSPS - PF SRIOV System Page Size Register

Table 39-81. PSRIOVSYSPS - PF SRIOV System Page Size Register

Description: SRIOV System Page Size Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 160h **Offset End:** 163h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 160h **Offset End:** 163h	
Size: 32 bit	**Default:** 00000001h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	SYSPAGSIZ	This field must define the page size the system will use to map the VFs' memory addresses. Software must set the value of the System Page Size to one of the page sizes set in the Supported Page Sizes field. As with Supported Page Sizes, if bit n is Set in System Page Size, the VFs associated with this PF are required to support a page size of 2^(n+12). For example, if bit 1 is Set, the system is using an 8k byte page size. The results are undefined if more than one bit is set in System Page Size. The results are undefined if a bit is Set in System Page Size that is not Set in Supported Page Sizes. When System Page Size is set, the VF associated with this PF is required to align all BAR resources on a System Page Size boundary. Each VF BARn or VF BARn pair (described below) shall be aligned on a System Page Size boundary. Each VF BARn or VF BARn pair defining a non-zero address space shall be sized to consume an integer multiple of System Page Size bytes. All data structures requiring page size alignment within a VF shall be aligned on a System Page Size boundary.		00000001h	RW

39.12.9.12 PSRIOVLBAR0 - SRIOV Lower BAR0 Register

The PSRIOVLBAR0 points to the Ring Controller rings. The base window size of this BAR is dependent on the System Page Size. However, the resulting window size is based on the System Page Size and the Number of VFs supported.

Table 39-82. PSRIOVLBAR0 - SRIOV Lower BAR0 Register

Description: SRIOV Lower BAR0 Register					
View: PCI	**BAR:** Configuration	**Bus:Device:Function:** B:D:0		**Offset Start:** 164h **Offset End:** 167h	
View: PCI PF	**BAR:** Configuration	**Bus:Function:** M:0		**Offset Start:** 164h **Offset End:** 167h	
Size: 32 bit	**Default:** 00000004h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :12	VFBAR0	**Lower SRIOV Base Address 0:** These bits are used to define the memory window for Virtual Functions. This window is used by Virtual Functions to address the Ring bundles. The memory base window size is dependent on the System Page Size as described in Section 39.12.9.11, "PSRIOVSYSPS - PF SRIOV System Page Size Register". As an example, for a System Page Size set at 8KB, and 16 Virtual Functions, this register will be hard-wired to request an 8KB memory window. Since there are 16 Virtual Functions, the resulting window size will be 16*8KB=128KB. Bits Above bit 11 Page Size — Window Size that respond as Reserved Memory 4-KB — None — 128-KB 8-KB — 12:12 — 256-KB 64-KB — 15:12 — 2048-KB 256-KB — 17:12 — 8192-KB 1-MB — 19:12 — 32-MB 4-MB — 21:12 — 128-MB		0h	RW
11 :04	VFZERO	**Lower Bits:** Hardwired to 0 (4KB region).		0h	RO
3	VFPREF	**Prefetchable:** Hardwired to 0 to indicate that the region is not prefetchable.		0h	RO
02 :01	VFTYP	**Addressing Type:** Hardwired to indicate a 64-bit region.		10b	RO
00	VFMEM	**Memory Space Indicator:** Hardwired to 0 to identify the region as in memory space.		0h	RO

39.12.9.13 PSRIOVUBAR0 - SRIOV Upper BAR0 Register

Table 39-83. PSRIOVUBAR0 - SRIOV Upper BAR0 Register

Description: SRIOV Upper BAR0 Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 168h **Offset End:** 16Bh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 168h **Offset End:** 16Bh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	VFBAR0	Upper SRIOV Base Address 0.		0h	RW

39.12.9.14 PSRIOVLBAR1 - SRIOV Lower BAR1 Register

The base window size of this BAR is dependent on the System Page Size. However, the resulting window size is based on the System Page Size and the Number of VFs supported.

Table 39-84. PSRIOVLBAR1 - SRIOV Lower BAR1 Register (Sheet 1 of 2)

Description: SRIOV Lower BAR1 Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 16Ch **Offset End:** 16Fh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 16Ch **Offset End:** 16Fh	
Size: 32 bit	**Default:** 00000004h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :12	VFBAR1	**Lower SRIOV Base Address 1:** These bits are used to define the memory window for Virtual Functions. The memory base window size is dependent on the System Page Size as described in Section 39.12.9.11, "PSRIOVSYSPS - PF SRIOV System Page Size Register". As an example, for a System Page Size set at 8KB, and 32 Virtual Functions, this register will be hardwired to request an 8KB memory window. Since there are 32 Virtual Functions, the resulting window size will be 32*8KB=256KB. Page Size — Bits Above bit 11 that respond as Reserved Memory — Window Size 4-KB — None — 128-KB 8-KB — 12:12 — 256-KB 64-KB — 15:12 — 2048-KB 256-KB — 17:12 — 8192-KB 1-MB — 19:12 — 32-MB 4-MB — 21:12 — 128-MB		0h	RW
11 :04	VFZERO	**Lower Bits:** Hardwired to 0 (4KB region)		0h	RO

Table 39-84. PSRIOVLBAR1 - SRIOV Lower BAR1 Register (Sheet 2 of 2)

Description: SRIOV Lower BAR1 Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 16Ch **Offset End:** 16Fh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 16Ch **Offset End:** 16Fh	
Size: 32 bit	**Default:** 00000004h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
3	VFPREF	**Prefetchable:** Hardwired to 0 to indicate that the region is not prefetchable.		0h	RO
02 :01	VFTYP	**Addressing Type:** Hardwired to indicate a 64-bit region.		10b	RO
00	VFMEM	**Memory Space Indicator:** Hardwired to 0 to identify the region as in memory space.		0h	RO

39.12.9.15 PSRIOVUBAR1 - SRIOV Upper BAR1 Register

Table 39-85. PSRIOVUBAR1 - SRIOV Upper BAR1 Register

Description: SRIOV Upper BAR1 Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 170h **Offset End:** 173h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 170h **Offset End:** 173h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	VFBAR1	**Upper SRIOV Base Address 1.**		0h	RW

39.12.9.16 PSRIOVVFMA - PF SRIOV VF Migration Array Register

Table 39-86. PSRIOVVFMA - PF SRIOV VF Migration Array Register

Description: SRIOV VF Migration Array Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 17Ch **Offset End:** 17Fh	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 17Ch **Offset End:** 17Fh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	Reserved	Reserved. ***Note:*** VF Migration not supported in EP.		00000000h	RO

39.12.10 Access Control Services (ACS) Capability

The PCIe ACS defines a set of control points within a PCIe topology to determine whether a TLP should be routed normally, blocked, or redirected. ACS is applicable to RCs, switches, and multifunction devices. The ACS Capability structure is shared and exposed to all PFs.

This section describes the registers for ACS Capability.

39.12.10.1 PACSCAPID - PF - ACS Capability ID Register

Table 39-87. PACSCAPID - PF ACS Capability ID Register

Description: ACS capability					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 1B0h **Offset End:** 1B1h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 1B0h **Offset End:** 1B1h	
Size: 16 bit	**Default:** 00Dh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	ACSCID	**Capability ID:** PCI SIG assigned a capability record ID for CAS per the 1.0 revision		000Dh	RO

39.12.10.2 PACSCVNC - PF ACS Capability Version and Next Capability Pointer Register

Table 39-88. PACSCVNC - PF ACS Capability Version and Next Capability Pointer Register

Description: ACS Capability Version					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 1B2h **Offset End:** 1B3h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 1B2h **Offset End:** 1B3h	
Size: 16 bit	**Default:** 1h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :4	ACSCO	Next Capability Offset - last extended capability.		00h	RO
3 :0	ACSCV	Capability Version - This is set to 1h for the Single Root I/O Virtualization and Sharing Specification, Revision 1.1.		1h	RO

39.12.10.3 PACSCAP - PF ACS Capabilities Register

Table 39-89. PACSCAP - PF ACS Capabilities Register

Description: ACS Capabilities Register					
View: PCI	**BAR:** Configuration		**Bus:Device:Function:** B:D:0	**Offset Start:** 1B4h **Offset End:** 1B7h	
View: PCI PF	**BAR:** Configuration		**Bus:Function:** M:0	**Offset Start:** 1B4h **Offset End:** 1B7h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :23	Reserved	Reserved0		0h	RO
:22	ACSDTP2PE	ACS Direct Translated P2P Enable (T) – Hard-wired to zero, not supported.		0h	RO
:21	ACSP2PECE	ACS P2P Egress Control Enable (E) – Hard-wired to zero, not supported.		0h	RO
:20	ACSUFE	ACS Upstream Forwarding Enable (U) – Hard-wired to zero, not supported.		0h	RO
:19	ACSP2PCRE	ACS P2P Completion Redirect Enable (C) – Hard-wired to zero, not supported.		0h	RO
:18	ACSP2PRRE	ACS P2P Request Redirect Enable (R) – Hard-wired to zero, not supported.		0h	RO
:17	ACSTBE	ACS Translation Blocking Enable (B) – Hard-wired to zero, not supported.		0h	RO
:16	ACSSVE	ACS Source Validation Enable (V) – Hard-wired to zero, not supported.		0h	RO
15 :8	ACSECVS	Egress Control Vector Size – Hard-wired to zero, not supported.		0h	RO
:7	Reserved1	Reserved.1		0h	RO
:6	ACSDTP2P	ACS Direct Translated P2P (T) – Hard-wired to zero, not supported.		0h	RO
:5	ACSP2PEC	ACS P2P Egress Control (E) – Hard-wired to zero, not supported.		0h	RO
:4	ACSUF	ACS Upstream Forwarding (U) – Hard-wired to zero, not supported.		0h	RO
:3	ACSP2PCR	ACS P2P Completion Redirect (C) – Hard-wired to zero, not supported.		0h	RO
:2	ACSP2PRR	ACS P2P Request Redirect (R) – Hard-wired to zero, not supported.		0h	RO
:1	ACSTB	ACS Translation Blocking (B) – Hard-wired to zero, not supported.		0h	RO
:0	ACSSV	ACS Source Validation (V) – Hard-wired to zero, not supported.		0b	RO

39.13 PCIe Endpoint Function 0 Registers (Intel QuickAssist Technology) - EndPoint (EP) Virtual Function (VF) PCI Configuration Space

High-lighted regions of the configurations space are RO for the VF configuration spaces and have the same value as the PF.

Table 39-90. VF PCIe Configuration Space (Sheet 1 of 2)

	3	2	1	0	Offset Note 1
Required PCI Compatible Configuration Space	Device ID		Vendor ID		00h
	Status		Command		04h
	Class Code			Revision ID	08h
	BIST	Header Type	Master Latency Timer	Cache Line Size	0Ch
	Read Only Zero				10h
					14h
	Read Only Zero				18h
					1Ch
	Read Only Zero				20h
					24h
	Cardbus CIS Pointer				28h
	Subsystem ID		Subsystem Vendor ID		2Ch
	Read Only Zero				30h
	Reserved			Capabilities Pointer	34h
	Reserved				38h
	Max_Lat	Min_Gnt	Interrupt Pin	Interrupt Line	3Ch
Reserved	Reserved				40h-4Fh
PCIE Cap	PCI Express Capabilities Register		Next Cap Pointer	PCIe CAP_ID	50h
	Device Capabilities				54h
	Device Status		Device Control		58h
	Link Capabilities				5Ch
	Link Status		Link Control		60h
	Reserved				64h - 73h
	Device Capabilities 2				74h
	Device Status 2 (reserved)		Device Control 2 (reserved)		78h
	Link Capabilities 2 (reserved)				7Ch
	Link Status 2 (reserved)		Link Control 2 (reserved)		80h
Reserved	Reserved				84h - 8Fh
MSI Cap	Message Control		Next Cap Pointer	MSI CAP_ID	90h
	MSI Addr				94h - 9Bh
	Reserved		MSI Data		9Ch
	MSI Mask				A0h
	MSI Pending				A4h
Reserved	Reserved				A8h - FFh

Table 39-90. VF PCIe Configuration Space (Sheet 2 of 2)

	3	2	1	0	Offset Note 1
AER Configuration Space	Next Capability Pointer/Capability Version		AER Capability ID		100h
	Uncorrectable Error Status				104h
	Uncorrectable Error Mask				108h
	Uncorrectable Error Severity				10Ch
	Correctable Error Status				110h
	Correctable Error Mask				114h
	Control and Capability				118h
	Header Log				11Ch
	Header Log				120h
	Header Log				124h
	Header Log				128h
ARI Configuration Space	Next Capability Pointer/Capability Version		ARI Capability ID		138h
	ARI Control		ARI Capability		13Ch
Reserved	Reserved				140h - 1ACh
ACS Configuration Space	Next Capability Pointer/Capability Version		ACS Capability ID		1B0h
	ACS Control Register		ACS Capability Register		1B4h
Reserved	Reserved				1B8h - FFCh

Notes:
1. Any addresses not shown are Read-only 0.

39.13.1 PCI Virtual Function (VF) Views

Table 39-91. Bus M, Function 8+Index 1: Summary of PCIe Intel QuickAssist Technology VF Configuration Registers (Sheet 1 of 2)

Offset Start	Offset End	Register ID - Description	Default Value
00h	01h	"VVID[0:15] - VF Vendor Identification Register" on page 3841	FFFFh
02h	03h	"VDID[0:15] - VF Device Identification Register" on page 3841	FFFFh
04h	05h	"VPCICMD[0:15] - VF Device Command Register" on page 3842	0000h
06h	07h	"VPCISTS[0:15] - VF PCI Device Status Register" on page 3843	0010h
08h	0Bh	"VRID[0:15] - VF Revision ID Register" on page 3844	0B400000h
0Eh	0Eh	"VHDR[0:15] - VF Header Type Register" on page 3844	00h
2Ch	2Dh	"VSVID[0:15] - VF Subsystem Vendor ID Register" on page 3844	8086h
2Eh	2Fh	"VSID[0:15] - VF Subsystem ID Register" on page 3845	0000h
34h	34h	"VCP[0:15] - VF Capabilities Pointer Register" on page 3845	90h
3Ch	3Ch	"VIRQL[0:15] - VF Interrupt Line Register" on page 3845	00h
3Dh	3Dh	"VIRQP[0:15] - VF Interrupt Pin Register" on page 3846	00h
50h	50h	"VPCID[0:15] - VF PCI Express Capability ID Register" on page 3846	10h
51h	51h	"VPCP[0:15] - VF PCI Express Next Capability Pointer Register" on page 3846	0h
52h	53h	"VPCR[0:15] - VF PCI Express Capabilities Register" on page 3847	002h
54h	57h	"VPDCAP[0:15] - VF PCI Express Device Capabilities Register" on page 3847	10008061h
58h	59h	"VPDC[0:15] - VF PCI Express Device Control Register" on page 3848	00h
5Ah	5Bh	"VPDS[0:15] - VF PCI Express Device Status Register" on page 3849	0000h
5Ch	5Fh	"VLCR[0:15] - VF Link Capabilities Register" on page 3850	00000011h
60h	61h	"VLCNTLR[0:15] - VF Link Control Register" on page 3852	0000h
62h	63h	"VLSR[0:15] - VF Link Status Register" on page 3852	0000h
74h	77h	"VDCAPR2[0:15] - VF Device Capabilities 2 Register" on page 3853	00000000h
7Ch	7Fh	"VLCR2[0:15] - VF Link Capabilities 2 Register" on page 3853	00000000h
80h	81h	"VLCNTLR2 [0:15] - VF Link Control 2 Register" on page 3854	0000h
82h	83h	"VLSR2[0:15] - VF Link Status Register" on page 3854	0000h
90h	90h	"VMSICID[0:15] - Message Signaled Interrupt Capability ID Register" on page 3854	05h
91h	91h	"VMSINCP[0:15] - Message Signaled Interrupt Next Capability Pointer Register" on page 3855	50h
92h	93h	"VMSICTL[0:15] - Message Signaled Interrupt Control Register" on page 3855	0180h
94h	97h	"VMSILADDR[0:15] - Message Signaled Interrupt Lower Address Register" on page 3856	00000000h
98h	9Bh	"VMSIUADDR[0:15] - Message Signaled Interrupt Upper Address Register" on page 3856	00000000h
9Ch	9Dh	"VMSIDATA[0:15] - Message Signaled Interrupt Data Register" on page 3856	0000h
A0h	A3h	"VMSIMSK[0:15] - VF Message Signaled Interrupt Mask Register" on page 3857	00000000h
A4h	A7h	"VMSIPND[0:15] - VF Message Signaled Interrupt Pending Register" on page 3857	00000000h
100h	103h	"VPCIEAERCAPID[0:15] - VF PCI Express AER Capability ID Register" on page 3858	13810001h
104h	107h	"VPAERUCS[0:15] - VF PCI Express AER Uncorrectable Error Status Register" on page 3858	0h

Table 39-91. Bus M, Function 8+Index 1: Summary of PCIe Intel QuickAssist Technology VF Configuration Registers (Sheet 2 of 2)

Offset Start	Offset End	Register ID - Description	Default Value
108h	10Bh	"VPAERUCM[0:15] - VF PCI Express AER Uncorrectable Error Mask Register" on page 3859	0h
10Ch	10Fh	"VPAERUCSEV[0:15] - VF PCI Express AER Uncorrectable Error Severity Register" on page 3861	00h
110h	113h	"VPAERCS[0:15] - VF PCI Express AER Correctable Error Status Register" on page 3862	0h
114h	117h	"VPAERCM[0:15] - VF PCI Express AER Correctable Error Mask Register" on page 3863	0h
118h	11Bh	"VPAERCTLCAP[0:15] - VF PCI Express AER Control and Capability Register" on page 3864	0h
11Ch	11Fh	"VPAERHDRLOG0[0:15] - VF PCI Express AER Header Log 0 Register" on page 3865	0h
120h	123h	"VPAERHDRLOG1[0:15] - VF PCI Express AER Header Log 1 Register" on page 3865	0h
124h	127h	"VPAERHDRLOG2[0:15] - VF PCI Express AER Header Log 2 Register" on page 3865	0h
128h	12Bh	"VPAERHDRLOG3[0:15] - VF PCI Express AER Header Log 3 Register" on page 3866	0h
138h	13Bh	"VARIDHDR[0:15] - VF Alternative Routing ID Capability Header" on page 3866	0001000Eh
13Ch	13Dh	"VFARICAP[0:15] - VF ARI Capabilities Register" on page 3867	00000000h
13Eh	13Fh	"VARIDCTL[0:15] - VF Alternative Routing ID Control Register" on page 3867	00000000h
1B0h	1B1h	"VACSCAPID[0:15] - VF ACS Capability ID Register" on page 3868	000Dh
1B2h	1B3h	"VACSCVNC[0:15] - VF ACS Capability Version and Next Capability Pointer Register" on page 3868	1h
1B4h	1B7h	"VACSCAP[0:15] - VF ACS Capabilities Register" on page 3868	00000000h

39.13.2 PCI Standard Header Registers

Reflects 16 Virtual Functions.

This section describes the PCI Configuration Space registers that make up the standard Type 0 header. Some information from the specification is repeated here as an aid to the reader or to describe implementation choice. Refer to the *PCI Express Base Specification 2.0* and *PCI Local Bus Specification* for the full register descriptions and additional information regarding their operation.

39.13.2.1 VVID[0:15] - VF Vendor Identification Register

The VID Register contains the vendor identification number. This 16-bit register combined with the Device Identification Register uniquely identifies any PCI device. Writes to this register have no effect.

Table 39-92. VVID[0:15] - VF Vendor Identification Register

Description: VF Vendor Identification Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 00h **Offset End:** 01h	
Size: 16 bit	**Default:** FFFFh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	VID	Vendor ID: The Single Root I/O Virtualization and Sharing Specification, Revision 1.1 requires that this field return FFFFH.		FFFFh	RO

39.13.2.2 VDID[0:15] - VF Device Identification Register

This 16-bit register combined with the Vendor Identification register uniquely identifies any PCI device. Writes to this register have no effect.

Table 39-93. VDID[0:15] - VF Device Identification Register

Description: VF Device Identification Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 02h **Offset End:** 03h	
Size: 16 bit	**Default:** FFFFh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	DID	Device ID: The Single Root I/O Virtualization and Sharing Specification, Revision 1.1 requires that this field return FFFFH. Software should return the VF Device ID value from the associated PF as the Device ID for the VF.		FFFFh	RO

39.13.2.3 VPCICMD[0:15] - VF Device Command Register

Table 39-94. VPCICMD[0:15] - VF Device Command Register

Description: VF Device Command Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 04h **Offset End:** 05h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :11	Reserved	Reserved0		0h	RO
10	INTD	Interrupt Disable: The Single Root I/O Virtualization and Sharing Specification, Revision 1.1 requires that this field is hardwired to 0b for all VFs. This bit does not apply to VFs.		0h	RO
9	FBTB	Fast Back to Back Enable: Does not apply to PCI Express. Hardwired to 0		0h	RO
8	SER	**SERR# Enable:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field is hardwired to 0b for all VFs. In addition the functionality associated with the setting of this bit in the Section 39.12.2.3, "PPCICMD - PF Device Command Register" will apply to all VFs.		0h	RO
7	Reserved0	Address/Data Stepping Control: Does not apply to PCI Express. Hard-wired to 0.		0h	RO
6	PER	**Parity Error Response**: The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field is hardwired to 0b for all VFs. In addition the functionality associated with the setting of this bit in the Section 39.12.2.3, "PPCICMD - PF Device Command Register" will apply to all VFs.		0h	RO
5	VPS	**VGA Palette Snoop Enable:** Does not apply to PCI Express. Hard-wired to 0.		0h	RO
4	MWE	**Memory Write and Invalidate Enable**: Does not apply to PCI Express. Hard-wired to 0.		0h	RO
3	SS	**Special Cycle Enable:** Does not apply to PCI Express. Hard-wired to 0.		0h	RO
2	BM	**Bus Master Enable:** When cleared, the EP is prevented from issuing any memory or I/O read/write requests. Requests other than memory or I/O requests are not controlled by this bit. The EP will initiate a completion transaction regardless of the setting.		0h	RW
1	MEM	**Memory Enable:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field is hardwired to 0b for all VFs.		0h	RO
0	IO	**I/O Space Enable:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field is hardwired to 0b for all VFs.		0h	RO

39.13.2.4 VPCISTS[0:15] - VF Device Status Register

Table 39-95. VPCISTS[0:15] - VF PCI Device Status Register

Description: VF PCI Device Status Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 06h **Offset End:** 07h	
Size: 16 bit	**Default:** 0010h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	DPE	**Detected Parity Error:** set when the EP receives a poisoned TLP regardless of the state of the Parity Error Response in the VPCICMD register.		0h	RW/1C/V
14	SSE	**SERR# Asserted:** set when the EP sends an ERR_FATAL or ERR_NONFATAL message, and the SERR Enable bit in the VPCICMD register is '1'.		0h	RW/1C/V
13	RMA	**Received Master Abort:** set when the EP receives a completion with Unsupported Request Completion Status.		0h	RW/1C/V
12	RTA	**Received Target Abort:** set when the EP receives a completion with Completer Abort Completion Status.		0h	RW/1C/V
11	STA	**Signaled Target Abort:** set when the EP completes a Request using Completer Abort Completion Status		0h	RW/1C/V
10 :09	DST	DEVSEL# **Timing:** Does not apply to PCI Express. Hard-wired to 0.		00b	RO
8	MDPE	**Master Data Parity Error:** This bit is set by the EP if its Parity Error Enable bit is set and either of the following two conditions occurs: This bit is set under the following conditions. EP receives a Poisoned Completion for an Outbound Read Request EP transmits a Poisoned TLP for an Outbound Write Request. If the Parity Error Response bit is cleared in the Section 39.12.2.3, "PPCICMD - PF Device Command Register", this bit is never set.		0h	RW/1C/V
7	FB2B	**Fast Back-to-Back**: Does not apply to PCI Express.Hard-wired to 0.		0h	RO
6	Reserved1	Reserved1		0h	RO
5	MC66	**66 MHz Capable (C66):** Does not apply to PCI Express. Hard-wired to 0		0h	RO
4	CL	**Capabilities List:** All PCI Express devices are required to implement the PCI Express capability structure. Hard-wired to 1.		1h	RO
3	INTSTS	**Interrupt Status:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field is hardwired to 0b for all VFs. This bit does not apply to VFs.		0h	RO
02 : 00	Reserved2	Reserved2		0h	RO

39.13.2.5 VRID[0:15] - Revision ID Register

The value of this register indicates the chip stepping. It is hardwired on chip and reflects the latest revision.

Table 39-96. VRID[0:15] - VF Revision ID Register

Description: VF Revision ID Register and VF Class Code Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 08h **Offset End:** 0Bh	
Size: 32 bit	**Default:** 0B400000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :8	CC	**Class Code:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field return the same value as the Section 39.12.2.6, "PCLS - PF Cache Line Size Register" for all VFs.		0B4000h	RW/O
07 :0	RID	**EP Revision:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 states that this field should be viewed as a Vendor Defined Extension to the Device ID. This may be different than the PF's Revision ID, but the same value must be reported by all VFs. Refer to the PF RID for a description.		00h	RO/V

39.13.2.6 VHDR[0:15] - VF Header Type Register

Table 39-97. VHDR[0:15] - VF Header Type Register

Description: VF Header Type Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 0Eh **Offset End:** 0Eh	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	HDR	**Header Type:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 states that this field must be set to 00H for VFs.		00h	RO

39.13.2.7 VSVID[0:15] - VF Subsystem Vendor ID Register

Table 39-98. VSVID[0:15] - VF Subsystem Vendor ID Register

Description: VF Subsystem Vendor ID Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 2Ch **Offset End:** 2Dh	
Size: 16 bit	**Default:** 8086h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	SVID	**Subsystem Vendor ID:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that when read, this read only register must return the same value as the Section 39.12.2.12, "PSVID - PF Subsystem Vendor ID Register" for all VFs.		8086h	RO/V

39.13.2.8 VSID[0:15] - VF Subsystem ID Register

This register is a write-once register. Once any byte in the register has been written, the register locks against further writes until reset.

Table 39-99. VSID[0:15] - VF Subsystem ID Register

Description: VF Subsystem ID Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 2Eh **Offset End:** 2Fh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	SID	**Subsystem ID:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this read only register return the same value for all VFs. This may be a different value than that contained in the PF (e.g.,Section 39.12.2.13, "PSID - PF Subsystem ID Register").		0000h	RW/O

39.13.2.9 VCP[0:15] - VF Capabilities Pointer Register

The Capabilities Pointer Register provides the offset in configuration space to the location where the first set of capabilities registers is located.

Table 39-100. VCP[0:15] - VF Capabilities Pointer Register

Description: VF Capabilities Pointer Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 34h **Offset End:** 34h	
Size: 8 bit	**Default:** 90h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	CP	**Capability List Pointer:** This field provides an offset into the configuration space pointing to the first item in the capability list which in the VF is the MSI extended capabilities header.		50h	RO

39.13.2.10 VIRQL[0:15] - VF Interrupt Line Register

Table 39-101. VIRQL[0:15] - VF Interrupt Line Register

Description: VF Interrupt Line Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 3Ch **Offset End:** 3Ch	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	IRQL	**Interrupt Assigned:** This field does not apply to VFs and is hardwired to Zero.		0h	RO

39.13.2.11 VIRQP[0:15] - VF Interrupt Pin Register

Table 39-102.VIRQP[0:15] - VF Interrupt Pin Register

Description: VF Interrupt Pin Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 3Dh **Offset End:** 3Dh	
Size: 8 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	IRQP	**Interrupt Assigned:** This field does not apply to VFs and is hardwired to Zero.		0h	RO

39.13.3 VF PCI Express Capability Structure

39.13.3.1 VPCID[0:15] - VF PCI Express Capability ID Register

The PCI Express Capability List register enumerates the PCI Express Capability structure in the PCI 3.0 configuration space capability list.

Table 39-103.VPCID[0:15] - VF PCI Express Capability ID Register

Description: VF PCI Express Capability ID Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 50h **Offset End:** 50h	
Size: 8 bit	**Default:** 10h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	PCIECID	**Cap Id:** This field identifies this item in the linked list of Extended Capability Headers as being the PCI Express capability registers.		10h	RO

39.13.3.2 VPCP[0:15] - VF PCI Express Next Capability Pointer Register

Table 39-104.VPCP[0:15] - VF PCI Express Next Capability Pointer Register

Description: VF PCI Express Next Capability Pointer Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 51h **Offset End:** 51h	
Size: 8 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	MCP	**Next Capability Pointer:** Last capability.		90h	RO

39.13.3.3 VPCR[0:15] - VF PCI Express Capabilities Register

Table 39-105.VPCR[0:15] - VF PCI Express Capabilities Register

Description: VF PCI Express Capabilities Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 52h **Offset End:** 53h	
Size: 16 bit	**Default:** 002h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :14	Reserved	Reserved		00b	RO
13 :9	IMN	**Interrupt Message Number**: This only applies to Root Complex and Switch devices. This register is hardcoded to 0.		00000b	RO
8	SI	**Slot Implemented:** Indicates that the PCI Express Link associated with this port is connected to a slot. Only valid for root complex and switch downstream ports. Hard-wired to 0		0b	RO
7 :4	DPT	**Device/Port Type:** Indicates the type of PCI Express logical device. 0000b - PCI Express Endpoint device		0000b	RO
3 :0	CV	**Capability Version:** Indicates PCI-SIG defined PCI Express capability structure version number EP supports version 2h.		0010b	RO

39.13.3.4 VPDCAP[0:15] - VF PCI Express Device Capabilities Register

Table 39-106.VPDCAP[0:15] - VF PCI Express Device Capabilities Register (Sheet 1 of 2)

Description: VF PCI Express Device Capabilities Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 54h **Offset End:** 57h	
Size: 32 bit	**Default:** 10008061h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :29	Reserved	Reserved		0h	RO
28	FLR	**FLR Cap:** Function Level Reset Capability is required for all VFs and PFs according to the *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1.		1b	RO
27 :26	CSPS	**Captured Slot Power Limit Scale:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 states that this field is undefined for all VFs.		00b	RO
25 :18	CSPV	**Captured Slot Power Limit Value**: The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 states that this field is undefined for all VFs.		0h	RO
17 :16	Reserved1	Reserved1		0h	RO
15	RBEP	**Role-Based Error Reporting**: this bit is set to indicate that this device implements the Role Base Error Reporting defined in *PCI Express Base Specification*, Revision 2.0.		1b	RO
14 :12	Reserved	**Reserved:** Undefined - Treated as Reserved		000b	RO
11 :9	EL1L	**Endpoint L1 Acceptable Latency:** EP does not support L1 active state power management.		000b	RO

Table 39-106.VPDCAP[0:15] - VF PCI Express Device Capabilities Register (Sheet 2 of 2)

Description: VF PCI Express Device Capabilities Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 54h **Offset End:** 57h	
Size: 32 bit	**Default:** 10008061h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
8 :6	EL0L	**Endpoint L0 Acceptable Latency:** Total acceptable latency that the EP can withstand due to a transition from L0s to L0 state.		001b	RO
5	ETFS	**Extended Tag Field Supported:** Indicates the maximum supported size of the Tag field as a Requester. EP does generate 8-bit Tags and supports 8-bit Tags as a completer. Supports 96 outstanding requests.		1b	RO
4 :3	PFS	**Phantom Functions Supported:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field is hardwired to 00b.		00b	RO
2 :0	MPS	**Max_Payload_Size Supported:** This field indicates the maximum payload size that EP can support for TLPs. This value is set to indicate 256B. The defined encodings are: • 000b = 128B max payload size • 001b = 256 bytes max payload size • 010b = 512 bytes max payload size (Max supported) • 011b - 111b = Reserved		010b	RO

39.13.3.5 VPDC[0:15] - VF PCI Express Device Control Register

Table 39-107.VPDC[0:15] - VF PCI Express Device Control Register (Sheet 1 of 2)

Description: VF PCI Express Device Control Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 58h **Offset End:** 59h	
Size: 16 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15	STARTFLR	**Initiate Function Level Reset:** A write of 1b to this bit initiates Function Level Reset to the VF. The value is always read as 0b.		0b	RW
14 :12	MRS	**Max Read Request Size:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field is hardwired to all zeros.		000b	RO
11	ENS	**Enable No Snoop:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field is hardwired to all zeros.		0b	RO
10	APME	**Aux Power PM Enable**: The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field is hardwired to all zeros.		0b	RO
9	PFE	**Phantom Functions Enable:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field is hardwired to all zeros.		0b	RO
8	ETFE	**Extended Tag Field Enable:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field is hardwired to all zeros.		0b	RO
7 :5	MPS	**Max Payload Size:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field is hardwired to all zeros.		000b	RO

Table 39-107.VPDC[0:15] - VF PCI Express Device Control Register (Sheet 2 of 2)

Description: VF PCI Express Device Control Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 58h **Offset End:** 59h	
Size: 16 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
4	ERO	**Enable Relaxed Ordering:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field is hardwired to all zeros.		0b	RO
3	URRO	**Unsupported Request Reporting Enable (URRE):** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field is hardwired to all zeros. In addition the functionality associated with this bit setting is controlled by the corresponding bit in "PPDCNTL - PF PCI Express Device Control Register", and will apply to all VFs.		0b	RO
2	FERE	**Fatal Error Reporting Enable:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field is hardwired to all zeros. In addition the functionality associated with this bit setting is controlled by the corresponding bit in "PPDCNTL - PF PCI Express Device Control Register", and will apply to all VFs.		0h	RO
1	NERE	**Non-Fatal Error Reporting Enable**: The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field is hardwired to all zeros. In addition the functionality associated with this bit setting is controlled by the corresponding bit in "PPDCNTL - PF PCI Express Device Control Register", and will apply to all VFs.		0h	RO
0	CERE	**Correctable Error Reporting Enable**: The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field is hardwired to all zeros. In addition the functionality associated with this bit setting is controlled by the corresponding bit in "PPDCNTL - PF PCI Express Device Control Register", and will apply to all VFs.		0h	RO

39.13.3.6 VPDS[0:15] - VF PCI Express Device Status Register

Table 39-108.VPDS[0:15] - VF PCI Express Device Status Register (Sheet 1 of 2)

Description: VF PCI Express Device Status Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 5Ah **Offset End:** 5Bh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :6	Reserved	Reserved		0h	RO
5	TP	**Transactions Pending:** This bit when set indicates that a device has issued Non-Posted Requests which have not been completed. A device reports this bit cleared only when all Completions for any outstanding Non-Posted Requests have been received.		0h	RW/1C/V
4	APD	**AUX Power Detected:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field is hardwired to all zeros.		0h	RO
3	URD	**Unsupported Request Detected:** This bit indicates that the device received an Unsupported Request. Errors are logged in this register regardless of whether error reporting is enabled or not in the Device Control Register. For a multi-function device, each function indicates status of errors as perceived by the respective function.		0h	RW/1C/V

Table 39-108.VPDS[0:15] - VF PCI Express Device Status Register (Sheet 2 of 2)

Description: VF PCI Express Device Status Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 5Ah **Offset End:** 5Bh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
2	FED	**Fatal Error Detected:** This bit indicates status of Fatal errors detected. A one indicates that an error was detected since the last time this bit was cleared. Errors are updated in this field regardless of whether the error reporting bit is set in the Device Control Register. For a multi-function device, each function indicates status of errors as perceived by the respective function.		0h	RW/1C/V
1	NED	**Non-Fatal Error Detected:** This bit indicates status of Nonfatal errors detected. A one indicates that an error was detected since the last time this bit was cleared. Errors are updated in this field regardless of whether the error reporting bit is set in the Device Control Register. For a multi-function device, each function indicates status of errors as perceived by the respective function.		0h	RW/1C/V
0	CED	**Correctable Error Detected:** This bit indicates status of correctable errors detected. A one indicates that an error was detected since the last time this bit was cleared. Errors are updated in this field regardless of whether the error reporting bit is set in the Device Control Register. For a multi-function device, each function indicates status of errors as perceived by the respective function.		0h	RW/1C/V

39.13.3.7 VLCR[0:15] - VF Link Capabilities Register

Table 39-109.VLCR[0:15] - VF Link Capabilities Register (Sheet 1 of 2)

Description: Link Capabilities Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 5Ch **Offset End:** 5Fh	
Size: 32 bit	**Default:** 00000011h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :24	PORTNUM	**Port Number**: Assigned by EP after link training phase.		0h	RO
23 :22	Reserved1	Reserved1		00b	RO
21 :21	Reserved2	Reserved2		0b	RO
20 :18	Reserved3	Reserved3		000b	RO
17 :15	L1EL	**L1 Exit Latency**- Indicates the exit latency from L1 to L0 state. EP does not support L1 transition 000b - Less than 1 is 001b - 1 is - 2 is 010b - 2 is - 4 is 011b - 4 is - 8 is 100b - 8 is - 16 is 101b - 16 is - 32 is 110b - 32 is - 64 is 111b - L1 transition not supported		000b	RO

Table 39-109.VLCR[0:15] - VF Link Capabilities Register (Sheet 2 of 2)

Description: Link Capabilities Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 5Ch **Offset End:** 5Fh	
Size: 32 bit	**Default:** 00000011h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
14 :12	L0EL	**L0s Exit Latency**- Indicates the exit latency from L0s to L0 state. 000b - Less than 64ns 001b - 64ns - 128ns 010b - 128ns - 256ns 011b - 256ns - 512ns 100b - 512ns - 1 ìs 101b - 1 ìs - 2 ìs 110b - 2 ìs - 4 ìs 111b - Reserved		000b	RO
11 :10	ASLPM	**Active State Link PM Support** - Indicates the level of active state power management supported in EP. Defined encodings are: 00b - Reserved 01b - L0s Entry Supported 10b - Reserved 11b - L0s and L1 Supported		00b	RO
9 :4	LINKW	**Max Link Width**- Indicates the max link width. Relevant encoding: 000000b - Reserved 000001b - x1 000010b - x2 000100b - x4 001000b - x8 001100b - x12 010000b - x16 100000b - x32 EP value depends on SKU. However the max link width is x16.		000001b	RO
3 :0	MAXSPEED	**Max Link Speed** - Indicates Maximum supported Link Speed. Defined encodings are: 0001b - 2.5Gbs Link speed supported (Gen 1) 0010b - 5.0Gbs Link speed supported (Gen 2) EP indicates a max Link Speed of 5.0 Gbs.		0001b	RO

39.13.3.8 VLCNTLR[0:15] - VF Link Control Register

Table 39-110.VLCNTLR[0:15] - VF Link Control Register

Description: Link Control Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 60h **Offset End:** 61h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :12	Reserved1	Reserved1		0000b	RO
11	Reserved	Reserved2		0b	RO
10	Reserved	Reserved3		0b	RO
9	Reserved	Reserved: The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field is hardwired to all zeros.		0b	RO
8	ECLKPM	The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field is hardwired to all zeros.		0b	RO
7	EXTSYNC	**Extended Synch**: The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field is hardwired to all zeros.		0b	RO
6	CCLKCFG	**Common Clock Configuration:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field is hardwired to all zeros.		0b	RO
5	RETRAIN	**Retrain Link**: Not Applicable to endpoints. Hard-wired to 0		0b	RO
4	LINKDIS	**Link Disable:** Not Applicable to endpoints. Hard-wired to 0		0b	RO
3	RCB	**Read Completion Boundary (RCB) Control:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field is hardwired to all zeros.		0b	RO
2	Reserved	Reserved4		0b	RO
1 :0	ASPMC	**Active State PM Control:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field is hardwired to all zeros.		00b	RO

39.13.3.9 VLSR[0:15] - VF Link Status Register

For the VF, all settings in this field are reserved and the PF setting in the Section 39.12.5.10, "PDCAPR2 - PF Device Capabilities 2 Register" applies to all of the VFs.

Table 39-111.VLSR[0:15] - VF Link Status Register

Description: Link Status Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 62h **Offset End:** 63h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	Reserved	Reserved		0H	RO

39.13.3.10 VDCAPR2[0:15] - VF Device Capabilities 2 Register

Table 39-112.VDCAPR2[0:15] - VF Device Capabilities 2 Register

Description: Device Capabilities 2 Register					
View: PCI VF	**BAR:** Configuration	**Bus:Device:Function:** M:8 + Index1		**Offset Start:** 74h **Offset End:** 77h	
Size: 32 bit	**Default:** 00000012h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :5	Reserved	Reserved.		0b	RO
4	CTODS	Completion Timeout Disable Supported. A value of 1b indicates support for the completion timeout disable mechanism. The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 states that this field value must be identical as the PF value.		1b	RO
3 :0	CTORS	Completion Timeout Ranges Supported: This field indicates support for the optional completion timeout programmability mechanism. This mechanism enables system software to modify the completion timeout value. Four time value ranges are defined: • Range A = 50 us to 10 ms • Range B = 10 ms to 250 ms • Range C = 250 ms to 4 s • Range D = 4 s to 64 s Bits are set according to the following table to show the timeout value ranges that are supported. • 0000b = Completion timeout programming not supported. • 0001b = Range A. • 0010b = Range B. • 0011b = Ranges A and B. • 0110b = Ranges B and C. • 0111b = Ranges A, B and C. • 1110b = Ranges B, C and D. • 1111b = Ranges A, B, C and D. • All other values are reserved. It is strongly recommended that the completion timeout mechanism not expire in less than 10 ms. The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 states that this field value must be identical as the PF value.		0010b	RO

39.13.3.11 VLCR2[0:15] - VF Link Capabilities 2 Register

Table 39-113.VLCR2[0:15] - VF Link Capabilities 2 Register

Description: Link Capabilities 2 Register					
View: PCI VF	**BAR:** Configuration	**Bus:Function:** M:8 + Index1		**Offset Start:** 7Ch **Offset End:** 7Fh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :0	Reserved	**Reserved**		00000000h	RO

39.13.3.12 VLCNTLR2 [0:15] - VF Link Control 2 Register

Table 39-114.VLCNTLR2 [0:15] - VF Link Control 2 Register

Description: Link Control Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 80h **Offset End:** 81h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :0	Reserved	Reserved4		0000h	RO

39.13.3.13 VLSR2[0:15] - VF Link Status 2 Register

For the VF, all settings in this field are reserved and the PF setting in the Section 39.12.5.10, "PDCAPR2 - PF Device Capabilities 2 Register" applies to all of the VFs.

Table 39-115.VLSR2[0:15] - VF Link Status Register

Description: Link Status 2 Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 82h **Offset End:** 83h	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	Reserved	Reserved		0H	RO

39.13.4 VF MSI Capability Structure

39.13.4.1 VMSICID[0:15] - Message Signaled Interrupt Capability ID Register

The Message Signaled Interrupt Capability record defines how the device generates PCI MSI messages. It is a 10B PCI SIG-defined capability record and includes the MCID, MCP, MCTL, MADR, and MDATA fields of the configuration header.

Table 39-116.VMSICID[0:15] - Message Signaled Interrupt Capability ID Register

Description: Message Signaled Interrupt Capability ID Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 90h **Offset End:** 90h	
Size: 8 bit	**Default:** 05h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	MCID	**Capability ID:** PCI SIG assigned capability record ID (05h, MSI capability)		05h	RO

39.13.4.2 VMSINCP[0:15] - Message Signaled Interrupt Next Capability Pointer Register

Table 39-117.VMSINCP[0:15] - Message Signaled Interrupt Next Capability Pointer Register

Description: Message Signaled Interrupt Next Capability Pointer Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 91h **Offset End:** 91h	
Size: 8 bit	**Default:** 50h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
07 :00	MCP	**Next Capability Pointer:** Next Capability is PCI Express.		0h	RO

39.13.4.3 VMSICTL[0:15] - Message Signaled Interrupt Control Register

Table 39-118.VMSICTL[0:15] - Message Signaled Interrupt Control Register

Description: Message Signaled Interrupt Control Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 92h **Offset End:** 93h	
Size: 16 bit	**Default:** 0180h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :09	Reserved	Reserved		0h	RO
08	MC	**Per-Vector Masking Capable:** Per-vector masking capable.		1b	RO
07	C64	**64 bit Address Capable:** Hardwired to 1 to indicate the device does not generate 64b message addresses.		1h	RO
06 :04	MME	**Multiple Message Enable:** System software writes to this field to indicate the number of allocated messages (less than or equal to the number of requested messages in MMC). A value of 0 corresponds to one message.		000h	RW
03 :01	MMC	**Multiple Message Capable:** System software reads this field to determine the number of requested messages. Hardwired to 0 to request one message.		000h	RO
00	MSIE	**MSI Enable:** System software sets this bit to enable MSI signaling. A device driver is prohibited from writing this bit to mask a device's service request. If 1, the device can use an MSI to request service. If 0, the device cannot use an MSI to request service.		0h	RW

39.13.4.4 VMSILADDR[0:15] - Message Signaled Interrupt Lower Address Register

Table 39-119.VMSILADDR[0:15] - Message Signaled Interrupt Lower Address Register

Description: Message Signaled Interrupt Lower Address Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 94h **Offset End:** 97h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :02	ADDR	**Message Address:** Written by the system to indicate the lower 30-bits of the address to use for the MSI memory write transaction.		0h	RW
01 :00	Reserved	**Reserved**		00b	RO

39.13.4.5 VMSIUADDR[0:15] - Message Signaled Interrupt Upper Address Register

Table 39-120.VMSIUADDR[0:15] - Message Signaled Interrupt Upper Address Register

Description: Message Signaled Interrupt Upper Address Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 98h **Offset End:** 9Bh	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	ADDR	**Message Address:** Written by the system to indicate the lower 32-bits of the address to use for the MSI memory write transaction. The lower two bits will always be written as 0.		0h	RW

39.13.4.6 VMSIDATA[0:15] - Message Signaled Interrupt Data Register

Table 39-121.VMSIDATA[0:15] - Message Signaled Interrupt Data Register

Description: Message Signaled Interrupt Data Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 9Ch **Offset End:** 9Dh	
Size: 16 bit	**Default:** 0000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	DATA	**Message Data:** Written by the system to indicate the lower 16 bits of the data written in the MSI memory write DWORD transaction. The upper 16 bits of the transaction are written as 0.		0h	RW

39.13.4.7 VMSIMSK[0:15] - VF Message Signaled Interrupt Mask Register

Table 39-122.VMSIMSK[0:15] - VF Message Signaled Interrupt Mask Register

Description: Message Signaled Interrupt Mask Register					
View: PCI VF	**BAR:** Configuration		**Bus:Device:Function:** M:8 + Index1	**Offset Start:** A0h **Offset End:** A3h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :01	Reserved	Reserved.		0h	RO
00 :00	MASK0	**Mask Bits:** Only one bit defined. Refer to the interrupt section.		0b	RW

39.13.4.8 VMSIPND[0:15] - VF Message Signaled Interrupt Pending Register

Table 39-123.VMSIPND[0:15] - VF Message Signaled Interrupt Pending Register

Description: Message Signaled Interrupt Pending Register					
View: PCI VF	**BAR:** Configuration		**Bus:Device:Function:** M:8 + Index1	**Offset Start:** A4h **Offset End:** A7h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :01	Reserved	Reserved.		0h	RO
00 :00	PNDB	**Pending Bits:** Only one bit defined. Refer to the interrupt section.		0b	RO/V

39.13.5 VF Advanced Error Reporting Capability Structure

39.13.5.1 VPCIEAERCAPID[0:15] - VF PCI Express AER Capability ID Register

The PCI Express Capability List register enumerates the PCI Express AER Capability structure in the PCI 3.0 configuration space capability list.

Table 39-124.VPCIEAERCAPID[0:15] - VF PCI Express AER Capability ID Register

Description: VF PCI Express AER Capability ID Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 100h **Offset End:** 103h	
Size: 32 bit	**Default:** 13810001h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :20	PCIEAERNCP	Next PCI Express Extended Capability Pointer: This is hardwired to 138H to point to the Alternative Routing ID extended capability.		138h	RO
19 :16	PCIEAERCVN	Advanced Error Capability Version Number: PCI Express Advanced Error Reporting Extended Capability Version Number.		1h	RO
15 :00	PCIEAERCID	Advanced Error Capability ID: PCI Express Extended Capability ID indicating Advanced Error Reporting Capability.		0001h	RO

39.13.5.2 VPAERUCS[0:15] - VF PCI Express AER Uncorrectable Error Status Register

Table 39-125.VPAERUCS[0:15] - VF PCI Express AER Uncorrectable Error Status Register

Description: VF PCI Express AER Uncorrectable Error Status Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 104h **Offset End:** 107h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :23	Reserved0	Reserved0:		0h	RO
:22	IEUNC	**Uncorrectable Internal Error Status:**		0b	RW/1C/V
:21	Reserved	Reserved		0b	RO
:20	UR	**Unsupported Request Error Status:** As a receiver, Set whenever an unsupported request is detected. The Header is logged.		0b	RW/1C/V
:19	ECRCC	**ECRC Check:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros.		0b	RO
:18	MTLP	**Malformed TLP:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros.		0b	RO
:17	RO	**Receiver Overflow:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros.		0b	RO
:16	EC	**Unexpected Completion**: As a receiver, set whenever a completion is received that does not match the EP requester ID or outstanding Tag. The Header is logged.		0b	RW/1C/V
:15	CA	**Completer Abort**: As a completer, set whenever an internal agent signals a data abort. The header is logged.		0b	RW/1C/V
:14	CT	**Completion Timeout:** As a requester, set whenever an outbound Non Posted Request does not receive a completion within 16-32ms.		0b	RW/1C/V

Table 39-125.VPAERUCS[0:15] - VF PCI Express AER Uncorrectable Error Status Register

Description: VF PCI Express AER Uncorrectable Error Status Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 104h **Offset End:** 107h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
:13	FCPES	**Flow Control Protocol Error Status:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros. Not supported.		0b	RO
:12	PTLPR	**Poisoned TLP Received:** As a receiver, set whenever a poisoned TLP is received from PCI Express. The header is logged. Note that internal queue errors are not covered by this bit, they are logged by the Configuration target of the transaction.		0h	RW/1C/V
11 :6	Reserved1	**Reserved1**		0h	RO
:05	SDES	**Surprise Down Error:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros. Not supported.		0b	RO
:04	DLPE	**Data Link Protocol Error**: As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros.		0b	RO
03 :00	Reserved2	**Reserved2**.		0b	RO

39.13.5.3 VPAERUCM[0:15] - VF PCI Express AER Uncorrectable Error Mask Register

Table 39-126.VPAERUCM[0:15] - VF PCI Express AER Uncorrectable Error Mask Register (Sheet 1 of 2)

Description: VF PCI Express AER Uncorrectable Error Mask Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 108h **Offset End:** 10Bh	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :23	Reserved0	Reserved0		0h	RO
:22	IEUNC	**Uncorrectable Internal Error Status:**		00b	RO
21	ACSVEM	**ACS Violation Error Mask:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros and that the setting of this mask in the Section 39.12.7.3, "PPAERUCM - PF PCI Express AER Uncorrectable Error Mask Register" applies to all of the VFs.		0b	RO
20	UR	**Unsupported Request Error Mask:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros and that the setting of this mask in the Section 39.12.7.3, "PPAERUCM - PF PCI Express AER Uncorrectable Error Mask Register" applies to all of the VFs.		0b	RO

Table 39-126.VPAERUCM[0:15] - VF PCI Express AER Uncorrectable Error Mask Register (Sheet 2 of 2)

Description: VF PCI Express AER Uncorrectable Error Mask Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 108h **Offset End:** 10Bh	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
19	ECRCC	**ECRC Check Error Mask:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros.		0b	RO
18	MTLP	**Malformed TLP Error Mask:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros.		0b	RO
17	RO	**Receiver Overflow Error Mask:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros.		0b	RO
16	EC	**Unexpected Completion Error Mask:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros and that the setting of this mask in the Section 39.12.7.3, "PPAERUCM - PF PCI Express AER Uncorrectable Error Mask Register" applies to all of the VFs.		0b	RO
15	CA	**Completer Abort Error Mask**: The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros and that the setting of this mask in the Section 39.12.7.3, "PPAERUCM - PF PCI Express AER Uncorrectable Error Mask Register" applies to all of the VFs.		0b	RO
14	CT	**Completion Time Out Error Mask:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros and that the setting of this mask in the Section 39.12.7.3, "PPAERUCM - PF PCI Express AER Uncorrectable Error Mask Register" applies to all of the VFs.		0b	RO
13	FCPES	**Flow Control Protocol Error Mask:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros. Not supported.		0b	RO
12	PTLPR	**Poisoned TLP Received Error Mask:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros and that the setting of this mask in the Section 39.12.7.3, "PPAERUCM - PF PCI Express AER Uncorrectable Error Mask Register" applies to all of the VFs.		0h	RO
11 :6	Reserved1	Reserved1.		0000b	RO
05	SDES	**Surprise Down Error Mask**: As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros. Not supported.		0b	RO
04	DLPE	**Data Link Protocol Error Mask:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros.		0b	RO
03 :01	Reserved2	Reserved2		0000b	RO
00	Reserved3	Reserved3		0b	RO

39.13.5.4 VPAERUCSEV[0:15] - VF PCI Express AER Uncorrectable Error Severity Register

Table 39-127.VPAERUCSEV[0:15] - VF PCI Express AER Uncorrectable Error Severity Register (Sheet 1 of 2)

Description: VF PCI Express AER Uncorrectable Error Severity Register					
View: PCI VF	**BAR:** Configuration	**Bus:Function:** M:8 + Index1		**Offset Start:** 10Ch **Offset End:** 10Fh	
Size: 32 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :23	Reserved0	reserved0		0h	RO
22	IEUNC	Uncorrectable Internal Error status severity		0b	RO
21	AVES	**ACS Violation Error Severity:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros and that the setting of this field in the Section 39.12.7.4, "PPAERUCSEV - PF PCI Express AER Uncorrectable Error Severity Register"applies to all of the VFs.		0b	RO
20	UR	**Unsupported Request Error Severity:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros and that the setting of this field in the Section 39.12.7.4, "PPAERUCSEV - PF PCI Express AER Uncorrectable Error Severity Register" applies to all of the VFs.		0b	RO
19	ECRCC	**ECRC Check Error Severity**: As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros.		0b	RO
18	MTLP	**Malformed TLP Error Severity:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros.		0b	RO
17	RO	**Receiver Overflow Error Severity:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros.		0b	RO
16	EC	**Unexpected Completion Error Severity:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros and that the setting of this field in the Section 39.12.7.4, "PPAERUCSEV - PF PCI Express AER Uncorrectable Error Severity Register" applies to all of the VFs.		0b	RO
15	CA	**Completer Abort Error Severity:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros and that the setting of this field in the Section 39.12.7.4, "PPAERUCSEV - PF PCI Express AER Uncorrectable Error Severity Register" applies to all of the VFs.		0b	RO
14	CT	**Completion Time Out Error Severity:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros and that the setting of this field in the Section 39.12.7.4, "PPAERUCSEV - PF PCI Express AER Uncorrectable Error Severity Register" applies to all of the VFs.		0b	RO
13	FCPES	**Flow Control Protocol Error Severity:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros. Not supported.		0b	RO

Table 39-127.VPAERUCSEV[0:15] - VF PCI Express AER Uncorrectable Error Severity Register (Sheet 2 of 2)

Description: VF PCI Express AER Uncorrectable Error Severity Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 10Ch **Offset End:** 10Fh	
Size: 32 bit	**Default:** 00h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
12	PTLPR	**Poisoned TLP Received Error Field:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros and that the setting of this field in the Section 39.12.7.4, "PPAERUCSEV - PF PCI Express AER Uncorrectable Error Severity Register" applies to all of the VFs.		0h	RO
11 :6	Reserved2	Reserved2		0000b	RO
5	SDES	**Surprise Down Error Severity:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros. Not supported.		0000b	RO
04	DLPE	**Data Link Protocol Error Severity:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros.		0b	RO
03 :01	Reserved3	Reserved3		0000b	RO
00	Reserved4	Reserved4		0b	RO

39.13.5.5 VPAERCS[0:15] - VF PCI Express AER Correctable Error Status Register

Table 39-128.VPAERCS[0:15] - VF PCI Express AER Correctable Error Status Register

Description: VF PCI Express AER Correctable Error Status Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 110h **Offset End:** 113h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :14	Reserved	**Reserved**.		00b	RO
13	ANFES	**Advisory Non-Fatal Error Status:**		0b	RW/1C/V
12	RTTS	**Replay Timer Timeout Status:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros.		0b	RO
11 :09	Reserved1	**Reserved**1.		0b	RO
08	RNRS	**REPLAY NUM Rollover Status:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros.		0h	RO
07	BDLLPS	**Bad DLLP Status:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros.		0000b	RO

Table 39-128.VPAERCS[0:15] - VF PCI Express AER Correctable Error Status Register

Description: VF PCI Express AER Correctable Error Status Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 110h **Offset End:** 113h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
06	DLPE	**Bad TLP Status: A**s a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros.		0b	RO
05 :01	Reserved2	**Reserved**2.		00000b	RO
00	RES	**Receiver Error Status:** As a non-function specific error, the *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros.		0b	RO

39.13.5.6 VPAERCM[0:15] - VF PCI Express AER Correctable Error Mask Register

Table 39-129.VPAERCM[0:15] - VF PCI Express AER Correctable Error Mask Register

Description: VF PCI Express AER Correctable Error Mask Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 114h **Offset End:** 117h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :14	Reserved	Reserved		00b	RO
13	ANFES	**Advisory Non-Fatal Error Mask:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be treated as reserved and that the setting of the corresponding bit in the Section 39.12.7.6, "PPAERCM - PF PCI Express AER Correctable Error Mask Register" will apply to the VFs.		0b	RO
12	RTTS	**Replay Timer Timeout Mask:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be treated as Reserved and that the setting of the corresponding bit in the Section 39.12.7.6, "PPAERCM - PF PCI Express AER Correctable Error Mask Register" will apply to the VFs.		0b	RO
11 :09	Reserved1	Reserved1.		000b	RO
08	RNRS	**REPLAY NUM Rollover Mask:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be treated as Reserved and that the setting of the corresponding bit in the Section 39.12.7.6, "PPAERCM - PF PCI Express AER Correctable Error Mask Register" will apply to the VFs.		0b	RO
07	BDLLPS	**Bad DLLP Mask:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be treated as Reserved and that the setting of the corresponding bit in the Section 39.12.7.6, "PPAERCM - PF PCI Express AER Correctable Error Mask Register" will apply to the VFs.		0b	RO

Table 39-129.VPAERCM[0:15] - VF PCI Express AER Correctable Error Mask Register

Description: VF PCI Express AER Correctable Error Mask Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 114h **Offset End:** 117h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
06	DLPE	**Bad TLP Mask:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be treated as Reserved and that the setting of the corresponding bit in the Section 39.12.7.6, "PPAERCM - PF PCI Express AER Correctable Error Mask Register" will apply to the VFs.		0b	RO
05 :01	Reserved2	Reserved2.		00h	RO
00	RES	**Receiver Error Mask:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be treated as Reserved and that the setting of the corresponding bit in the Section 39.12.7.6, "PPAERCM - PF PCI Express AER Correctable Error Mask Register" will apply to the VFs.		0b	RO

39.13.5.7 VPAERCTLCAP[0:15] - VF PCI Express AER Control and Capability Register

Table 39-130.VPAERCTLCAP[0:15] - VF PCI Express AER Control and Capability Register

Description: VF PCI Express AER Control and Capability Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 118h **Offset End:** 11Bh	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :09	Reserved	Reserved		0h	RO
08	ECRCCE	**ECRC Check Enable:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros and that the setting of this field in the Section 39.12.7.1, "PPCIEAERCAPID - PF PCI Express AER Capability ID Register" applies to all of the VFs.		0b	RO
07	ECRCCC	**ECRC Check Capable:** Indicates the EP is **not** capable of checking ECRC.		0b	RO
06	ECRCGE	**ECRC Generation Enable:** The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 requires that this field be hardwired to all zeros and that the setting of this field in the Section 39.12.7.1, "PPCIEAERCAPID - PF PCI Express AER Capability ID Register" applies to all of the VFs.		0b	RO
05	ECRCGC	**ECRC Generation Capable:** Indicates the EP is **not** capable of generating ECRC.		00000b	RO
04 :00	TFEP	**The First Error Pointer:** Identifies the bit position of the first error reported in the Section 39.13.5.2, "VPAERUCS[0:15] - VF PCI Express AER Uncorrectable Error Status Register" register. This register will not update until all bits in the ERRUNC STS register are cleared.		0h	RO/V

39.13.5.8 VPAERHDRLOG0[0:15] - VF PCI Express AER Header Log 0 Register

Table 39-131.VPAERHDRLOG0[0:15] - VF PCI Express AER Header Log 0 Register

Description: VF PCI Express AER Header Log 0 Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 11Ch **Offset End:** 11Fh	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HDRLOGDW0	1st DWord of the Header for the PCI Express packet in error (HDRLOGDW0): Once an error is logged in this register, it remains locked for further error logging until the time the software clears the status bit that cause the header log i.e., the error pointer is rearmed to log again.		0h	RO/V

39.13.5.9 VPAERHDRLOG1[0:15] - VF PCI Express AER Header Log 1 Register

Table 39-132.VPAERHDRLOG1[0:15] - VF PCI Express AER Header Log 1 Register

Description: VF PCI Express AER Header Log 1 Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 120h **Offset End:** 123h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HDRLOGDW1	2nd DWord of the Header for the PCI Express packet in error (HDRLOGDW1): Once an error is logged in this register, it remains locked for further error logging until the time the software clears the status bit that cause the header log i.e., the error pointer is rearmed to log again.		0h	RO/V

39.13.5.10 VPAERHDRLOG2[0:15] - VF PCI Express AER Header Log 2 Register

Table 39-133.VPAERHDRLOG2[0:15] - VF PCI Express AER Header Log 2 Register

Description: VF PCI Express AER Header Log 2 Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 124h **Offset End:** 127h	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HDRLOGDW2	3rd DWord of the Header for the PCI Express packet in error (HDRLOGDW2): Once an error is logged in this register, it remains locked for further error logging until the time the software clears the status bit that cause the header log i.e., the error pointer is rearmed to log again.		0h	RO/V

39.13.5.11 VPAERHDRLOG3[0:15] - VF PCI Express AER Header Log 3 Register

Table 39-134.VPAERHDRLOG3[0:15] - VF PCI Express AER Header Log 3 Register

Description: VF PCI Express AER Header Log 3 Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 128h **Offset End:** 12Bh	
Size: 32 bit	**Default:** 0h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :00	HDRDWLOG3	4th DWord of the Header for the PCI Express packet in error (HDRDWLOG3): Once an error is logged in this register, it remains locked for further error logging until the time the software clears the status bit that cause the header log i.e., the error pointer is rearmed to log again.		0h	RO/V

39.13.6 VF Alternative Routing ID Extended Capability Structure

This section describes the PCI Express Extended Configuration Space registers that make up the Alternative Routing ID Extended Capability Structure.

Some information from the specification is repeated here as an aid to the reader or to describe implementation choice. Refer to the *PCI Express Base Specification 2.0* for the full register descriptions and additional information regarding their operation.

39.13.6.1 VARIDHDR[0:15]– VF Alternative Routing ID Capability Header

This register contains information associated with the Alternative Routing ID capability. This is compliant with the *PCI-SIG ECN: Alternatifve Routing-ID Interpretation (ARI)*, Updated June4,2007.

Table 39-135.VARIDHDR[0:15] - VF Alternative Routing ID Capability Header

Description: VF Alternative Routing ID Capability Header					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 138h **Offset End:** 13Bh	
Size: 32 bit	**Default:** 0001000Eh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :20	ARINCO	**Next Capability Offset**: This field contains pointer to next Capability.		1B0h	RO
19 :16	ARICV	**Capability Version**: This is set to 1h for the most current version of the specification.		1h	RO
15 :0	ARIC	**PCI Express Extended Capability ID**: The PCI SIG has assigned 000Eh to the ARI extended capability.		000EH	RO

39.13.6.2 VFARICAP[0:15] - VF ARI Capabilities Register

This register contains information associated with the Alternative Routing ID capability.

Table 39-136.VFARICAP[0:15] - VF ARI Capabilities Register

Description: VF ARI Capabilities Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 13Ch **Offset End:** 13Dh	
Size: 16 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :8	NFN	**Next Function Number**: The *Single Root I/O Virtualization and Sharing Specification*, Revision 1.1 states that this field is undefined for VFs.		0h	RO
7 :2	Reserved	Reserved.		0h	RO
1	ACS	**ACS Functional Groups Capability**: not supported.		0b	RO
0	MFVC	**MFVC Functional Groups Capability**: not supported.		0b	RO

39.13.6.3 VARIDCTL[0:15] - VF Alternative Routing ID Control Register

This register contains information associated with the Alternative Routing ID capability.

Table 39-137.VARIDCTL[0:15] - VF Alternative Routing ID Control Register

Description: VF Alternative Routing ID Control Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 13Eh **Offset End:** 13Fh	
Size: 16 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :7	Reserved	Reserved		0h	RO
6 :4	FG	**Function Group**: Hardwired to Zero EP does not support Function Groups.		0b	RO
3 :2	Reserved1	Reserved1		0b	RO
1	ACS	**ACS Functional Groups Enable**: Hardwired to Zero as EP does not support.		0b	RO
0	MFVC	**MFVC Functional Groups Enable**: Hardwired to Zero as EP does not support.		0b	RO

39.13.7 VF Access Control Services (ACS) Capability

The PCIe ACS defines a set of control points within a PCIe topology to determine whether a TLP should be routed normally, blocked, or redirected. ACS is applicable to RCs, switches, and multifunction devices. The ACS Capability structure is shared and exposed to all VFs.

This section describes the registers for ACS Capability.

39.13.7.1 VACSCAPPID[0:15] - VF - ACS Capability ID Register

Table 39-138.VACSCAPID[0:15] - VF ACS Capability ID Register

Description: ACS capability					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 1B0h **Offset End:** 1B1h	
Size: 16 bit	**Default:** 000Dh			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :00	ACSCID	**Capability ID:** PCI SIG assigned a capability record ID for CAS per the 1.0 revision		000Dh	RO

39.13.7.2 VACSCVNC[0:15] - VF ACS Capability Version and Next Capability Pointer Register

Table 39-139.VACSCVNC[0:15] - VF ACS Capability Version and Next Capability Pointer Register

Description: ACS Capability Version					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 1B2h **Offset End:** 1B3h	
Size: 16 bit	**Default:** 1h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
15 :4	ACSCO	Next Capability Offset - last extended capability.		00h	RO
3 :0	ACSCV	Capability Version - This is set to 1h for the Single Root I/O Virtualization and Sharing Specification, Revision 1.1.		1h	RO

39.13.7.3 VACSCAP[0:15] - VF ACS Capabilities Register

Table 39-140.VACSCAP[0:15] - VF ACS Capabilities Register (Sheet 1 of 2)

Description: ACS Capabilities Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 1B4h **Offset End:** 1B7h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
31 :23	Reserved	Reserved0		0h	RO
:22	ACSDTP2PE	ACS Direct Translated P2P Enable (T) – Hard-wired to zero, not supported.		0h	RO
:21	ACSP2PECE	ACS P2P Egress Control Enable (E) – Hard-wired to zero, not supported.		0h	RO
:20	ACSUFE	ACS Upstream Forwarding Enable (U) – Hard-wired to zero, not supported.		0h	RO
:19	ACSP2PCRE	ACS P2P Completion Redirect Enable (C) – Hard-wired to zero, not supported.		0h	RO

Table 39-140.VACSCAP[0:15] - VF ACS Capabilities Register (Sheet 2 of 2)

Description: ACS Capabilities Register					
View: PCI VF	**BAR:** Configuration		**Bus:Function:** M:8 + Index1	**Offset Start:** 1B4h **Offset End:** 1B7h	
Size: 32 bit	**Default:** 00000000h			**Power Well:** Core	
Bit Range	**Bit Acronym**	**Bit Description**	**Sticky**	**Bit Reset Value**	**Bit Access**
:18	ACSP2PRRE	ACS P2P Request Redirect Enable (R) – Hard-wired to zero, not supported.		0h	RO
:17	ACSTBE	ACS Translation Blocking Enable (B) – Hard-wired to zero, not supported.		0h	RO
:16	ACSSVE	ACS Source Validation Enable (V) – Hard-wired to zero, not supported.		0h	RO
15 :8	ACSECVS	Egress Control Vector Size – Hard-wired to zero, not supported.		0h	RO
:7	Reserved1	Reserved.1		0h	RO
:6	ACSDTP2P	ACS Direct Translated P2P (T) – Hard-wired to zero, not supported.		0h	RO
:5	ACSP2PEC	ACS P2P Egress Control (E) – Hard-wired to zero, not supported.		0h	RO
:4	ACSUF	ACS Upstream Forwarding (U) – Hard-wired to zero, not supported.		0h	RO
:3	ACSP2PCR	ACS P2P Completion Redirect (C) – Hard-wired to zero, not supported.		0h	RO
:2	ACSP2PRR	ACS P2P Request Redirect (R) – Hard-wired to zero, not supported.		0h	RO
:1	ACSTB	ACS Translation Blocking (B) – Hard-wired to zero, not supported.		0h	RO
:0	ACSSV	ACS Source Validation (V) – Hard-wired to zero, not supported.		0b	RO

§

40 PCH EndPoint Only (EPO) Mode

40.1 Acronyms

Acronyms	Description
EPO	EndPoint Only Mode - special PCH configuration for standalone Intel® QuickAssist Technology and 10 GbE functionality
PCH EPO	When the PCH is configured to for EndPoint Only Mode
Intel® QAT	Intel® QuickAssist Technology
SPS FW	Server Platform Services is FW required by the PCH and is part of a shared NVM image stored on SPI NVM device.
SPS EPO FW	A version of SPS FW used specifically to support PCH EPO Mode
VSP	Virtual Switch Port

40.2 References

None.

40.3 PCH EndPoint Only (EPO) Mode Overview

The PCH supports an Endpoint Only Mode. In this mode, a non-legacy (non-boot) PCH is configured as a PCIe Endpoint device without support for the legacy PCH I/O capabilities. Up to three endpoints associated with Intel QuickAssist Technology are supported and can be enumerated as well as the integrated 10GbE endpoint.

PCH EndPoint Only Mode is configured via a combination of PCH functions, PCH soft straps, and SPS EPO FW.

Note: EPO / EPO PCH **is not** a unique PCH SKU. Rather it is a configuration where a non-boot PCH gets set to specific mode as described in this section.

Note: Intel QuickAssist Technology may not be supported on all PCH SKUs.

40.4 PCH Endpoint Only (EPO) Mode Features

- Intel QuickAssist Technology
  - — Up to three Intel QAT endpoints
- 10 GbE Controller
  - — Up to four 10/1 GbE ports
- PCH PCI Express Uplink
  - — Up to 24 PCI Express Gen 3 Lanes
    - Primary x16 max. uplink
    - Secondary x8 max. uplink
  - — Two Uplink configuration mapping options
- PCH SPI NVM Flash Interface
- PCH SMLink Interface

- PCH Thermal Sensor Support
- Multiple EPO PCHs supported on a platform
  - — Down on the motherboard
  - — Via PCIe Add-in Card

Note: For more details on the supported features described above, refer to their respective section within this document.

40.5 PCH EPO Mode Considerations

The section contains details and considerations for PCH EPO Mode.

40.5.1 EPO PCH - SPI

In Endpoint Only Mode, the PCH SPI controller is enabled and accessible to support FW requirements for Intel QuickAssist Technology and the integrated 10 GbE endpoint. In this mode, since DMI is unavailable, SPI related cycles that require PCIe Root Port interaction are transacted via the PCIe x16 PCH Upstream Port. This capability is primarily available in order to support in-system updating of the SPI flash device.

40.5.2 EPO PCH - Thermal Sensor and P2SB

The PCH thermal sensor and P2SB are available in EPO mode. Since DMI is not available in EPO mode, transactions are handled via the primary PCIe x16 Uplink and VSP[5:4].

40.5.3 EPO PCH - Additional Considerations

The following are other PCH EPO considerations:

- Endpoint Only Mode requires the use of External clocking mode to support it's input clock requirements. Refer to Section 8.5 for more details on external clocking.
- Intel® C620 Series Chipset SKUs that support Intel QuickAssist Technology require an Adaptive Voltage Identification (AVID) compatible VR solution (Vccprim_Avid of 0.85V to 1.0V) to be used.

40.5.4 EP0 PCH - FW

The EPO PCH utilizes a shared flash image that contains a subset of regions. The supported regions are the Flash Descriptor Region (FDR), the Intel ME Region, and the 10 GbE Region. While the FDR and Intel ME Regions are required the 10 GbE region is only required if the 10 GbE controller will be supported in the EPO Mode.

The EPO PCH uses SPS EPO Intel ME FW. This version of FW is specific for EPO mode. Neither SPS Si Enabling or SPS with Node Manager FW are compatible with EPO mode.

If the 10GbE controller will be supported as part of a EPO PCH implementation it's FW must also be used and integrated into the shared SPI NVM flash image.

The EPO PCH shared FW image does not require a BIOS image or any other additional FW aside from what has previously been identified in this section.

40.5.5 EPO PCH - Soft Straps

Setting the PCH into EPO mode requires the use of specific soft strap settings. Refer to the *Intel*® C620 Series Chipset *PCH-Server SPI Programmings Guide* (RDC document number 559021) for soft strap setting details.

Note: The SPS EPO FW package must be used as it works in conjunction with soft strap settings to ensure proper configuration for EPO mode.

40.5.6 EPO PCH - Unused PCH Functions / Interfaces

In PCH EPO mode the PCH Upstream Port(s) are connected to a PCIe root port; however DMI is not connected nor utilized. Since DMI is not available, the following tables lists PCH capabilities that **are not supported**, not functional, and cannot be enumerated or used in PCH EPO Mode. Many of these functions will either be power gated or held in reset in PCH EPO mode.

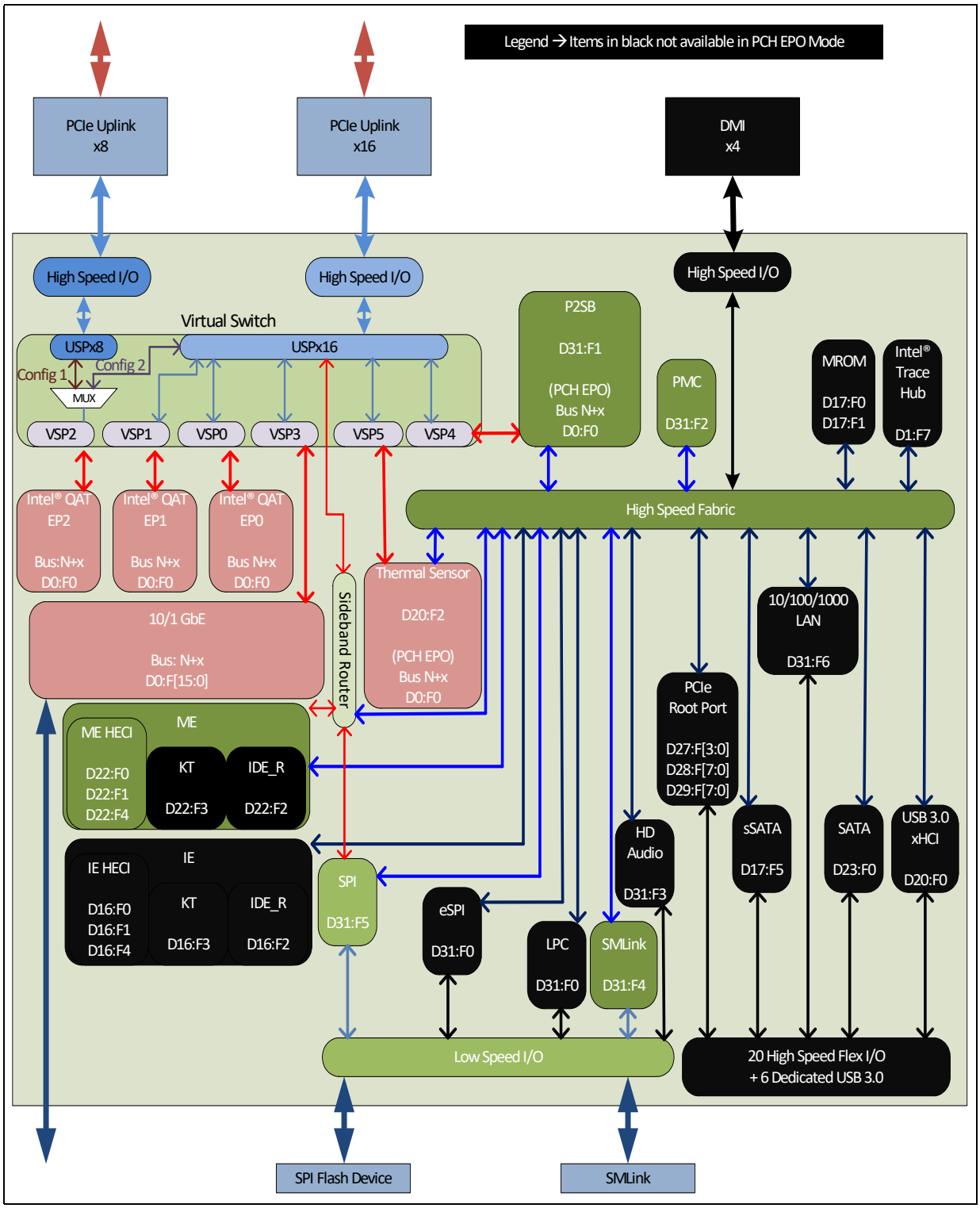
Table 40-1. EPO PCH - Functions Not Supported

DMI	PCI Express Root Ports	xHCI USB
SATA	sSATA	eSPI
LPC	Intel HD Audio	Host SMBus
10/100/1000 GbE	Fan Speed Controller	Pulse Width Modulation
Innovation Engine	Manageability: KT or IDE-R	MROM
Any function that requires DMI Access		Intel Trace Hub

40.5.7 PCH EPO - Block Diagram

The following is a block diagram reflecting a PCH in EPO mode. The blocks represented in black are not available in PCH EPO mode.

Figure 40-1. PCH in EndPoint Only (EPO) Mode Block Diagram



40.6 EPO PCH - PCIe Uplink

40.6.1 PCIe Uplink Overview

The EPO PCH has up to two PCIe Uplinks. The primary PCIe Gen 3 Uplink has a max width of 16 lanes and is required to be connected. The primary upstream port controller is used by both the Intel QuickAssist Technology and the Integrated 10GbE Controller. The max width of the primary Uplink is PCH SKU dependent.

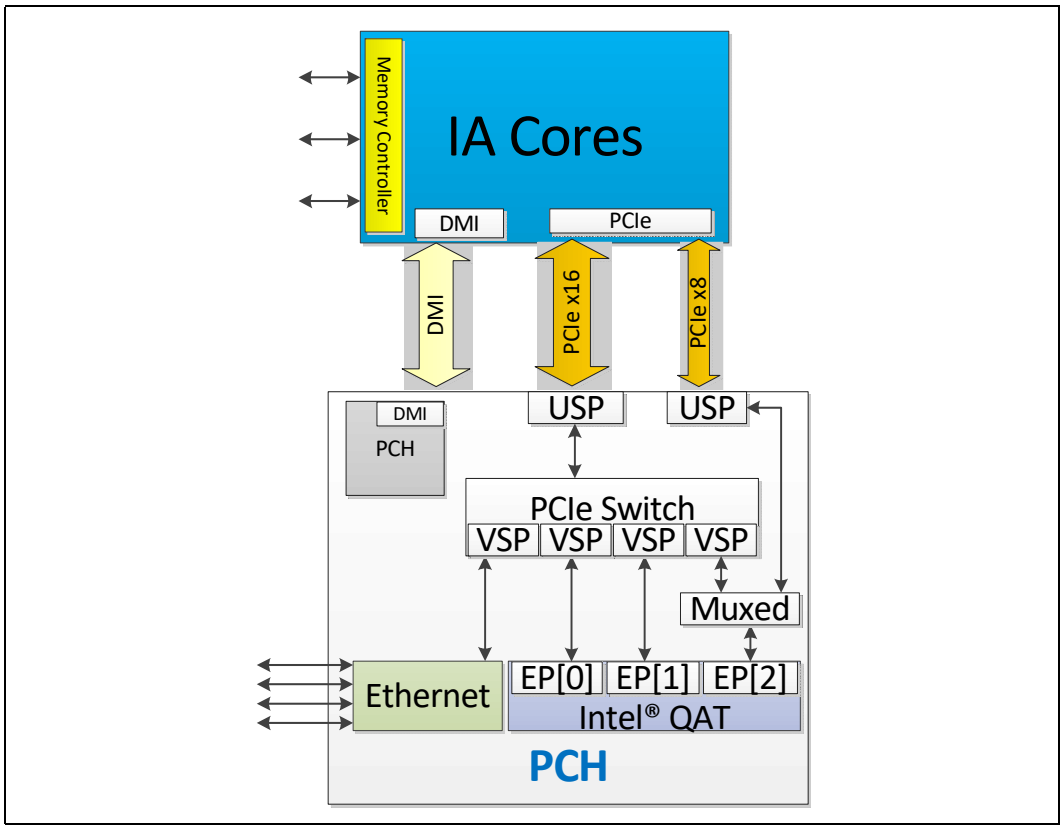
For PCH SKUs that support a second Upstream Port controller, an additional x8 lanes are available that are dedicated to Intel QuickAssist Technology. The secondary PCIe Gen 3 Uplink has a max width of 8 lanes, is optional, and requires the use of Configuration 2 described later in this section.

All three Intel QuickAssist Technology endpoints are accessible via an integrated PCIe Switch and the switch's Virtual Switch Ports (VSP). On PCH SKUs that support the secondary x8 Upstream Port controller, the secondary upstream port controller is muxed with the Intel QuickAssist Technology endpoint 2 (EP[2]) and will have dedicated access to that endpoint (EP[2]) as shown in the following figure.

The PCH has two PCIe Uplink Configurations: Configuration 1 and Configuration 2. These configuration are controlled via softstraps.

Note: Not all PCH SKUs support the secondary PCIe x8 Uplink. On those SKUs that do not support the secondary Uplink, the Uplink has been fused to Configuration 2.

Figure 40-2. Intel QuickAssist Technology and PCH Upstream Ports

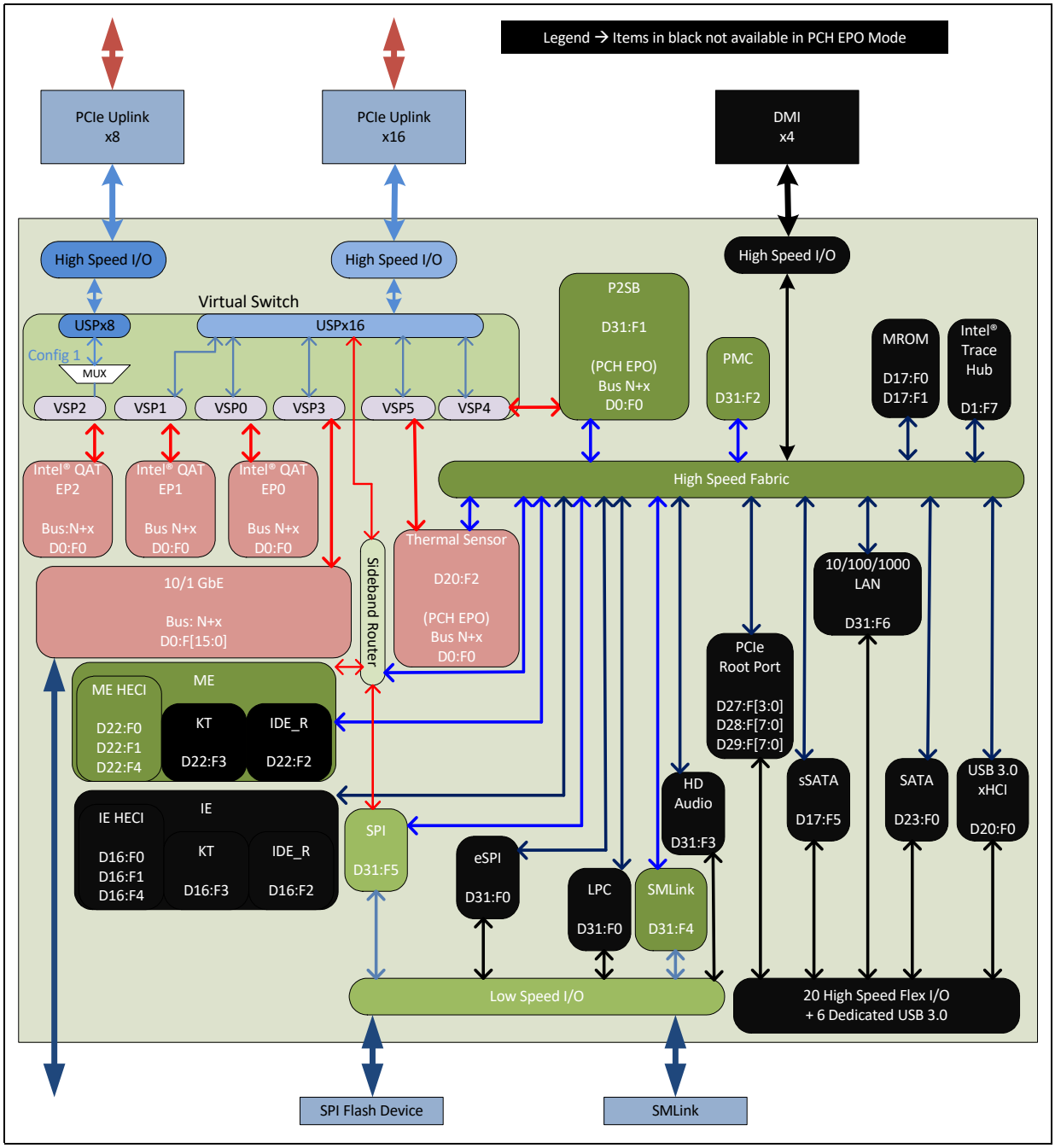


40.6.2 EPO PCH - PCI Uplink Configuration 1

In PCIe Uplink Configuration 1, the primary PCIe x16 Uplink is used for transactions to the 10 GbE controller, Intel QAT Endpoints [1:0], P2SB, and the PCH thermal sensor via the Virtual Switch Ports [5:3,1:0]. The secondary PCIe x8 Uplink is connected and dedicated to Intel QAT Endpoint 2 via Virtual Switch Port 2.

The following is a block diagram of PCH EPO Mode Uplink configuration #1.

Figure 40-3. EPO PCH with Uplink Configuration 1

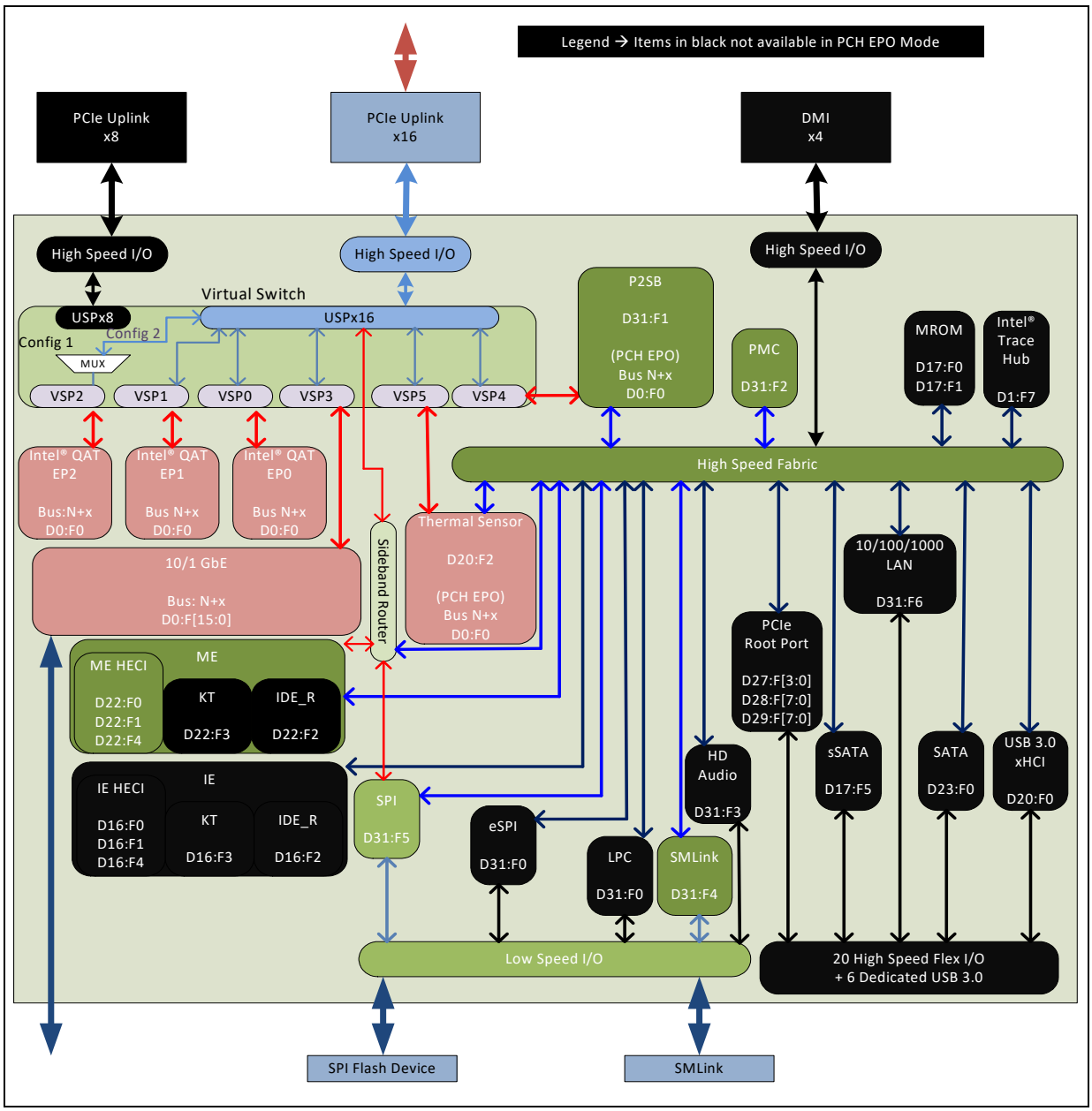


40.6.3 EPO PCH - PCI Uplink Configuration 2

In PCIe Uplink Configuration 2, the primary PCIe x16 Uplink is used for transactions to the 10 GbE controller, Intel QAT Endpoints [2:0], P2SB, and the PCH thermal sensor via the Virtual Switch Ports [5:0]. The secondary PCIe x8 Uplink is unused and is power gated.

The following is a block diagram of PCH EPO Mode Uplink configuration #2.

Figure 40-4. EPO PCH with Uplink Configuration 2



§